REPORT

Mar 25, 2025 P. Kurella

.JENKINS 101.

ABOUT JENKINS:

Jenkins is an open-source automation server made by HUDSON that is used for continuous integration (CI) and continuous delivery (CD) in software development. It helps automate building, testing, and deploying applications, making development faster and more efficient. Jenkins is highly extensible, with thousands of plugins available for integration with different tools and services.

→ **Automation Tool:** Used to automate the projects without any interference from humans.
→ **CI/CD Tool (Continous Integration/Continous Deployment):**

- **CI:** Process where you integrate a set of tools or a set of processes that are followed before delivering the application to the client.
- **CD:** A process where you deploy/deliver your application to your end user.
- **CI/CD:** An approach to software development that combines the processes of CI and CD to make developing applications faster, safer, and more efficient.

USES OF JENKINS:

- Jenkins overcomes the drawbacks of both agile and waterfall models.
- The port number of Jenkins is 8080.
- Jenkins is also known as the Heart of DevOps.
- We integrate all the DevOps tools with Jenkins and deploy the applications onto the web server (tomcat).
- We can integrate DevOps tools like Git, Maven, Docker, Kubernetes, Ansible, Teraform, Sonarqube, and Nexus for deployment. We use another separate instance i.e., Tomcat.
- In Jenkins in order to CI/CD, we need to create jobs or projects first.
- Running a job/project is known as “build”.
- In Jenkins, there are 2 types of Jobs
    - Free Style Jobs: by clicks in dashboard of Jenkins.
    - Pipeline jobs: Jenkins dashboard → groovy scripts & github →jenkins file in github → executing them.
- Jenkins will create parameter instances.

Perks of using Jenkins:

- **Automation of Software Development:** Jenkins automates repetitive tasks such as building and testing applications, significantly reducing human errors and improving efficiency.
- **Extensive Plugin Ecosystem:** Jenkins provides over 1,800+ plugins, allowing seamless integration with various tools and platforms like Git, Docker, Kubernetes, AWS, and more.
- **Faster Development and Deployment:** By providing instant feedback on code changes, Jenkins ensures that bugs are identified and fixed early, speeding up the software development lifecycle.
- **Cross-Platform Compatibility:** Jenkins supports multiple operating systems, including Windows, macOS, and Linux, making it highly versatile for different development environments.
- **Open-Source and Cost-Effective:** Jenkins is completely free to use, backed by a large community that continuously enhances its features and security.
- **Scalability and Distributed Builds:** Jenkins supports parallel execution of jobs and can distribute workloads across multiple machines, improving efficiency and resource utilization.
- **Pipeline as Code (Jenkinsfile):** Jenkins allows developers to define build and deployment workflows using a Jenkinsfile, enabling better version control and collaboration.
- **Security and Customization:** Jenkins offers role-based access control, authentication, and security plugins, ensuring a secure CI/CD pipeline.

Important Jenkins configurations:

- **Jenkins home directory:** /var/lib/Jenkins
- **Install plugins:** /var/lib/Jenkins/plugins
- **Create jobs:** /var/lib/Jenkins/workspace
- **Nodes info:** /var/lib/Jenkins/nodes
- **Jenkins logs:** /var/lib/Jenkins/log

Install Jenkins in your AWS instance:

Step 1: Install Java:

Jenkins requires Java to run. Install OpenJDK 11:

```
$ sudo amazon-linux-extras enable corretto8
$ sudo yum install java-11-amazon-corretto -y
```

Verify the installation:

```
$ version --java
```

Step 2: Add the Jenkins Repository:

Jenkins is not included by default in Amazon Linux 2, so you need to add the repository.

```
$ sudo wget -O /etc/yum.repos.d/jenkins.repo \
https://pkg.jenkins.io/redhat-stable/jenkins.repo
```

Import the Jenkins key:

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

Step 3: Install Jenkins:

Now, install Jenkins using the following command:

```
$ sudo yum install jenkins -y
```

Step 4: Start and Enable Jenkins:

Once installed, start Jenkins and enable it to launch on system boot:

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

Check the status to ensure Jenkins is running:

```
$ sudo systemctl status jenkins
```

Step 5: Configure Firewall for Jenkins:

By default, Jenkins runs on port 8080. Allow inbound traffic on this port:

```
$ sudo firewall-cmd --permanent --add-port=8080/tcp
$ sudo firewall-cmd --reload
```

If using AWS Security Groups, ensure port 8080 is open in the EC2 instance's inbound rules.

Step 6: Access Jenkins Web Interface:

1. Open a browser and navigate to:
   http://your-ec2-public-ip:8080
2. Retrieve the initial admin password:
   `$ sudo cat /var/lib/jenkins/secrets/initialAdminPassword`
3. Copy the password, paste it into the Jenkins web UI, and complete the setup.