

Jenkins – Basic Documentation

What is Jenkins?

Jenkins is an open-source automation server used to automate building, testing, and deploying applications.

It helps implement Continuous Integration (CI) and Continuous Deployment (CD).

Key Features of Jenkins

1. Open Source & Free
2. Cross-Platform
3. Extensible with Plugins
4. Easy Integration with Git, Docker, AWS, etc.
5. Pipeline Support
6. Distributed Builds

Jenkins Architecture

1. Jenkins Master – Controls build process and manages jobs.
2. Jenkins Agent (Slave) – Executes build tasks assigned by the master.
3. Jenkins Console – Web interface for monitoring jobs.

Jenkins Installation (Ubuntu)

1. `sudo apt update`
2. `sudo apt install openjdk-17-jdk -y`
3. Add Jenkins repo and key
4. `sudo apt install jenkins -y`
5. `sudo systemctl start jenkins`
6. Access Jenkins at `http://localhost:8080`

Unlocking Jenkins (First Time Setup)

1. Visit `http://localhost:8080`
2. Run: `sudo cat /var/lib/jenkins/secrets/initialAdminPassword`
3. Paste password, install plugins, and create admin user.

Jenkins Workflow (CI/CD)

1. Developer pushes code to GitHub.
2. Jenkins triggers build automatically.
3. Code is built and tested.
4. Application is deployed to server/cloud.

Basic Jenkins Pipeline Example

```
pipeline {  
  agent any  
  stages {  
    stage('Build') { steps { echo 'Building...' } }  
    stage('Test') { steps { echo 'Testing...' } }  
    stage('Deploy') { steps { echo 'Deploying...' } }  
  }  
}
```

Common Jenkins Commands

```
sudo systemctl start jenkins  
sudo systemctl stop jenkins  
sudo systemctl restart jenkins  
sudo systemctl status jenkins  
cat /var/lib/jenkins/secrets/initialAdminPassword
```

Common Plugins

- Git Plugin
- Pipeline Plugin
- Docker Plugin
- Slack Plugin
- AWS Plugin

Advantages

- Easy to set up
- Open-source
- Integrates with many DevOps tools
- Supports distributed builds
- Large community

Disadvantages

- Outdated UI
- Requires manual maintenance
- Plugin compatibility issues