Jenkins Controller + 2 Docker Agents (RHEL 10)

Prepared by: Narendra Geddam
Date: November 2025

Objective

Set up a Jenkins controller in Docker with two inbound agents for distributed CI/CD builds on RHEL 10.

System Requirements

- RHEL 10 / Ubuntu 22.04+
- Docker & Docker Compose installed
- Internet connection
- 4 GB RAM minimum

Step 1: Prepare Jenkins Home Directory

Run the following commands to create the Jenkins home directory and set permissions:

```
mkdir -p /home/narendra/Jenkins/jenkins_home
sudo chown -R 1000:1000 /home/narendra/Jenkins/jenkins_home
```

Step 2: Docker Compose Setup (WebSocket-based Agents)

Create `/home/narendra/Jenkins/docker-compose.yml` with the following content:

```
services:
  jenkins:
    container_name: jenkins
    image: jenkins/jenkins:lts
    ports:
      - "8080:8080"
      - "50000:50000"
    volumes:
      - /home/narendra/Jenkins/jenkins_home:/var/jenkins_home
    networks:
      - jenkins_net

  agent1:
    image: jenkins/inbound-agent
    container_name: jenkins-agent-1
    environment:
      - JENKINS_URL=http://jenkins:8080
      - JENKINS_AGENT_NAME=agent1
      - JENKINS_SECRET=<REPLACE_WITH_YOUR_KEY>
      - JENKINS_WEB_SOCKET=true
    depends_on:
      - jenkins
    networks:
      - jenkins_net

  agent2:
    image: jenkins/inbound-agent
    container_name: jenkins-agent-2
    environment:
      - JENKINS_URL=http://jenkins:8080
      - JENKINS_AGENT_NAME=agent2
      - JENKINS_SECRET=<REPLACE_WITH_YOUR_KEY>
      - JENKINS_WEB_SOCKET=true
    depends_on:
      - jenkins
    networks:
      - jenkins_net

networks:
  jenkins_net:
```

Step 3: Access Jenkins

Visit: http://localhost:8080

Get the admin password:
docker exec -it jenkins cat /var/jenkins_home/secrets/initialAdminPassword

Step 4: Configure Agents in Jenkins UI

1. Go to Manage Jenkins → Nodes → New Node
2. Create agent1 and agent2
3. Launch method → Launch agent by connecting it to the controller
4. Enable WebSocket
5. Copy agent secrets and replace <REPLACE_WITH_YOUR_KEY> in compose file.

Step 5: Verify Connection

```
docker logs jenkins-agent-1 | grep Connected
docker logs jenkins-agent-2 | grep Connected
```

Expected Output: INFO: WebSocket connection established INFO: Connected
INFO: Jenkins agent is ready to receive tasks

Step 6: Test Pipeline

```
pipeline {
    agent { label 'agent1' }
    stages {
        stage('Build') {
            steps {
                echo 'Running on agent1'
                sh 'hostname'
            }
        }
        stage('Test') {
            agent { label 'agent2' }
            steps {
                echo 'Running on agent2'
                sh 'echo Hello from agent2'
            }
        }
    }
}
```

Step 7: Parallel Build Example

```
pipeline {
    agent none
    stages {
        stage('Parallel Test') {
            parallel {
                stage('Run on agent1') {
                    agent { label 'agent1' }
                    steps {
                        echo "Hello from agent1"
                        sh 'hostname'
                    }
                }
                stage('Run on agent2') {
                    agent { label 'agent2' }
                    steps {
                        echo "Hello from agent2"
                        sh 'hostname'
                    }
                }
            }
        }
    }
}
```

Troubleshooting

Problem	Fix
Agents offline	Check WebSocket and secrets
TCP error 404	Enable 'TCP port for inbound agents' = 50000
Jenkins not reachable	Verify jenkins_net network
Permission denied	Run sudo chown -R 1000:1000 /home/narendra/Jenkins/jenkins_home

Author

Prepared by: Narendra Geddam
Date: November 2025
Setup Type: Jenkins + 2 Docker Agents (RHEL 10)