**Experiment No. 6 — Jenkins Master-Slave Architecture with Scaling**

**Aim:**

To implement Jenkins Master–Slave architecture with scaling.

**Steps to Perform**

**STEP 1: Open Jenkins Dashboard**

1. Go to your browser → http://localhost:8080
2. Log in to your Jenkins dashboard.
3. Click on:  
   **Manage Jenkins → Manage Nodes and Clouds → New Node**

**STEP 2: Create a New Node (Slave/Agent)**

1. In the **Node Name** field, enter a name like agent2.
2. Select **Permanent Agent** → Click **OK**.
3. This opens the configuration page for the new node.

**STEP 3: Configure the Node**

Fill in the following details:

* **Name:** agent2
* **Description:** Secondary build node
* **# of Executors:** 1
* **Remote root directory:** /home/jenkins (or any directory on the agent machine)
* **Labels:** agent2 (this label will be used in jobs later)
* **Usage:** "Use this node as much as possible"
* **Launch method:** Select  
  👉 **Launch agent by connecting it to the master** (Java Web Start / JNLP)

Under **Node Properties**, if required, specify:

* **JDK path** → /usr/lib/jvm/java-11-openjdk-amd64

Then click **Save**.

**STEP 4: Configure JNLP Port (Master Side)**

After clicking **Save**, Jenkins will show an error message — because the master cannot yet communicate with the agent.

To fix it:

1. Go to **Manage Jenkins → Security → Configure Global Security**.
2. Find **TCP port for inbound agents**.
3. Set a fixed port, e.g. 50000.
4. Click **Save**.

Now, from the terminal (on master system), allow this port through the firewall:

sudo ufw allow 50000/tcp

✅ This allows Jenkins Master to accept agent connections.

**STEP 5: Connect the Slave (Agent) Machine**

Now go back to Jenkins:

1. Go to **Manage Jenkins → Nodes → agent2**
2. You’ll see **“Launch agent via Java Web Start”** instructions.
3. Copy the command shown there — something like:
4. java -jar agent.jar -jnlpUrl http://<master-ip>:8080/computer/agent2/slave-agent.jnlp -secret <secret-key> -workDir "/home/jenkins"
5. Run this command on your **agent machine** (or same machine if simulating).

✅ Once executed, you’ll see a message:

INFO: Connected to Jenkins master

and in Jenkins → agent2 will show **Connected** (green dot).

**STEP 6: Create a Job to Test the Agent**

1. Go to **Dashboard → New Item → Freestyle Project**.
2. Enter a name like test-agent-job.
3. In **Restrict where this project can be run**, type:
4. agent2
5. Add a **Build Step → Execute Shell**, and enter:
6. echo "Hello from Agent Node!"
7. hostname
8. Click **Save** and then **Build Now**.

**STEP 7: Check the Output**

1. Click on the build number → **Console Output**
2. You’ll see:
3. Hello from Agent Node!
4. <agent-system-name>

✅ This confirms your Jenkins Slave node successfully executed the job.

**STEP 8: Verify Connected Nodes**

1. Go to **Dashboard → Manage Jenkins → Nodes**
2. You’ll see both:
   * **Master (built-in node)**
   * **agent2 (connected)**

✅ You can now add more slave nodes with different environments for scaling.