**Prerequisites:**

* Jenkins master (controller) node.
* At least one remote machine to act as a slave (agent).
* SSH access between master and agent.
* Java installed on both master and slave nodes.

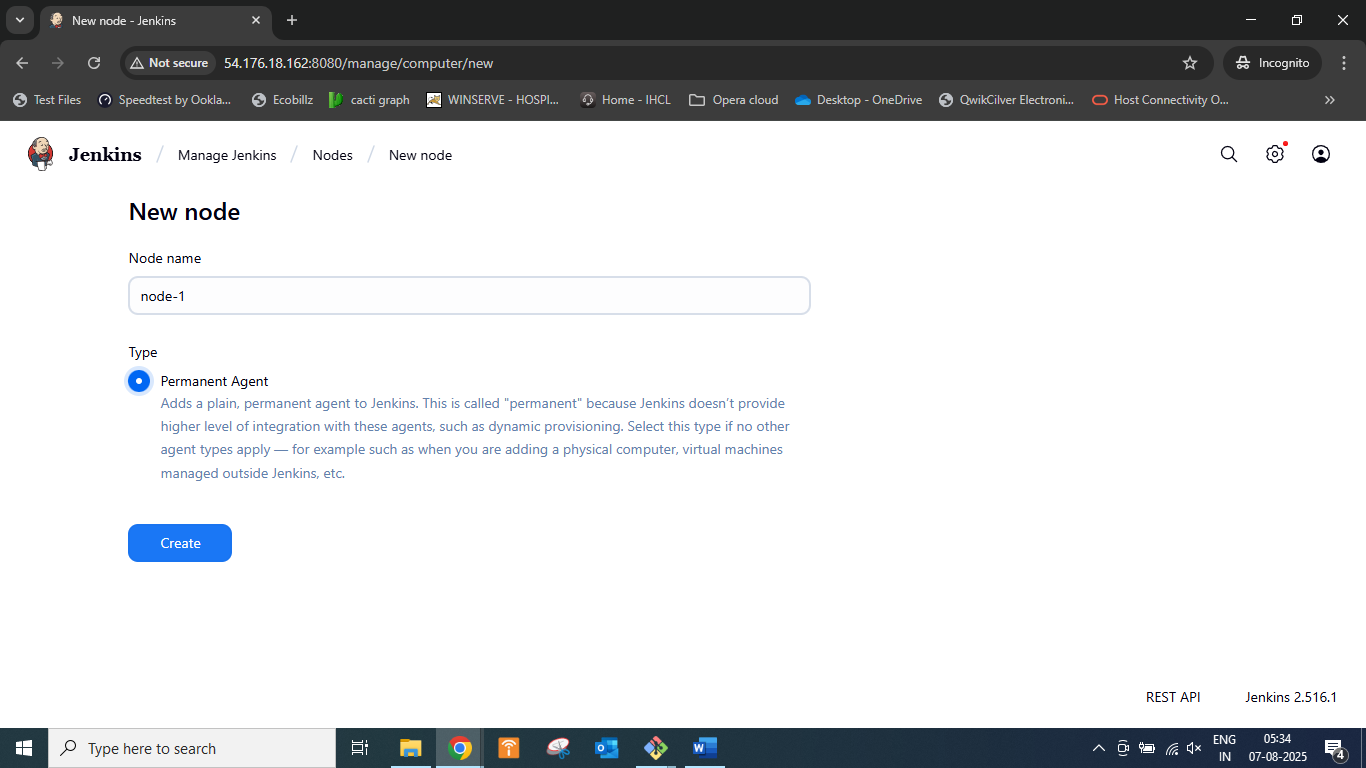
**Step 1: Install Jenkins on the Master Node:**

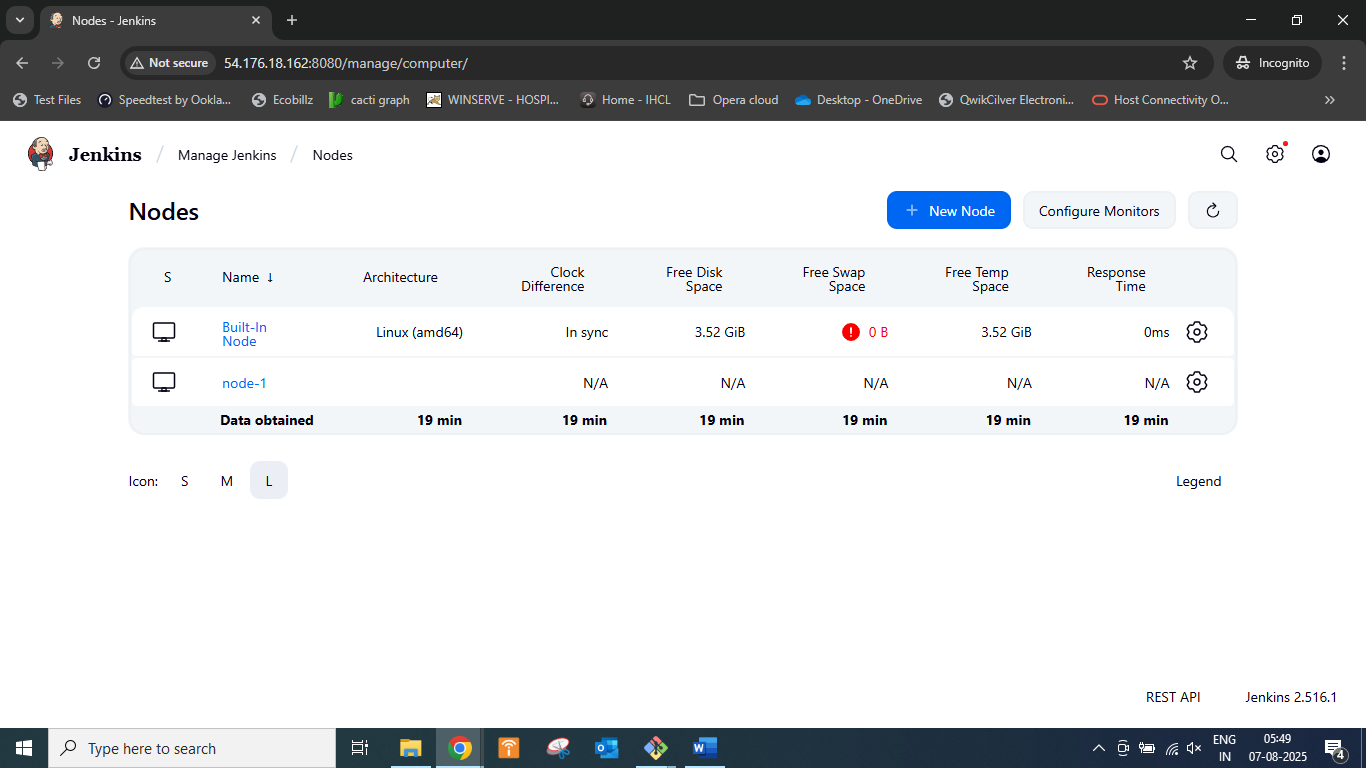
* Make sure Jenkins is running on the master machine.
* Access Jenkins at http://<master-ip>:8080.

**Step 2: Install Java on the Master Node:**

**Step 3: Add New Node (Slave):**

1. Go to: Manage Jenkins → Manage Nodes and Clouds
2. Click on "New Node"
3. Enter a node name.
4. Select Permanent Agent → Click OK





**Step 4: Install Java on the Agent Node:**

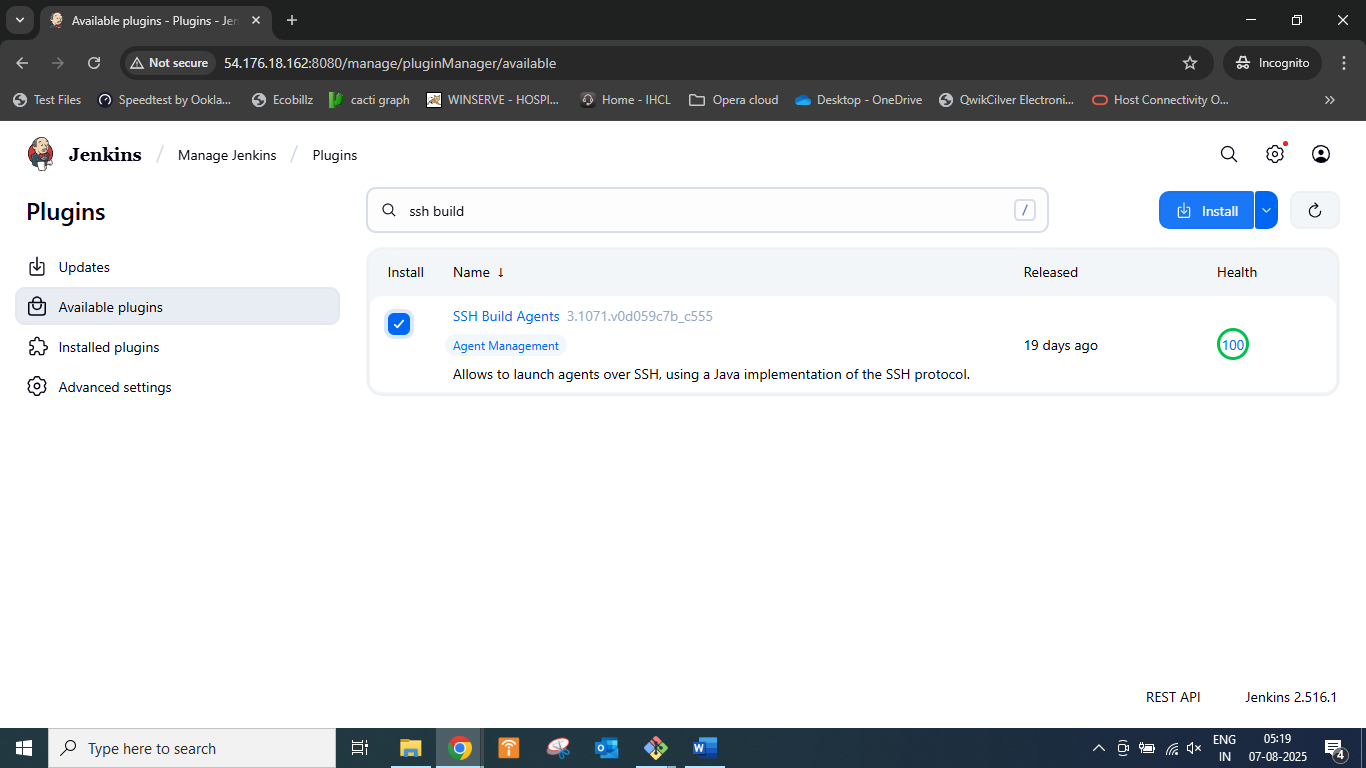
Make sure Java (compatible version) is installed on the slave machine.

java –version if not available

sudo apt install openjdk-17-jdk -y to install java

**Step 5: Install ssh-build agent on the Master Node:**

Install ssh-build Agent on the Master Node which allows to launch agents over SSH using a java implementation of the SSH protocol and restart Jenkins.



A screenshot of a computer

AI-generated content may be incorrect.

**Step 6: Configure Slave Node:**

Fill in the following:

* Remote root directory: A folder path on the agent machine (e.g., /home/jenkins)
* Labels: Optional, for grouping slaves
* Usage: Select how Jenkins schedules builds
* Launch method:
  + Launch agent via SSH (for Linux agents)
  + Launch agent via Java Web Start (for Windows agents)
  + Launch agent by connecting it to the controller (manual mode)

Provide:

* Host (IP address or hostname of agent)
* Credentials (username/password or SSH key)

**Step 7: Verify Agent is Connected**

* Go to Manage Jenkins → Manage Nodes
* The agent node should show a green checkmark and “Online” status.

**Step 8: Use Agent in Jobs**

* When creating or configuring a Jenkins job:
  + Under **"Restrict where this project can be run"**, set the **Label Expression** to match your agent's label.
* This will cause the job to run on that specific agent.

A screenshot of a computer

AI-generated content may be incorrect.

**Summary: Master-Slave Architecture in Jenkins**

The Master-Slave Architecture in Jenkins (now often called Controller-Agent Architecture) is designed to distribute build workloads across multiple machines, improving performance, scalability, and flexibility.