NAME : PARENITA A.SHIRSATH PRN : 221101062 ROLL.NO : 57 T.E.A.I.&.D.S.



**SEPM EXPERIMENT 6**

**Aim:**  To understand Jenkins Master Slave Architecture and scale your Jenkins standalone implementation by implementing slave nodes.

**Theory:**

A Jenkins slave node is simply a device configured to act as an automation executor on behalf of the master. The Jenkins master simply represents the base installation of Jenkins. The master will continue to perform basic operations and serve the user interface, while the slaves do the heavy lifting.

**Jenkins Master**

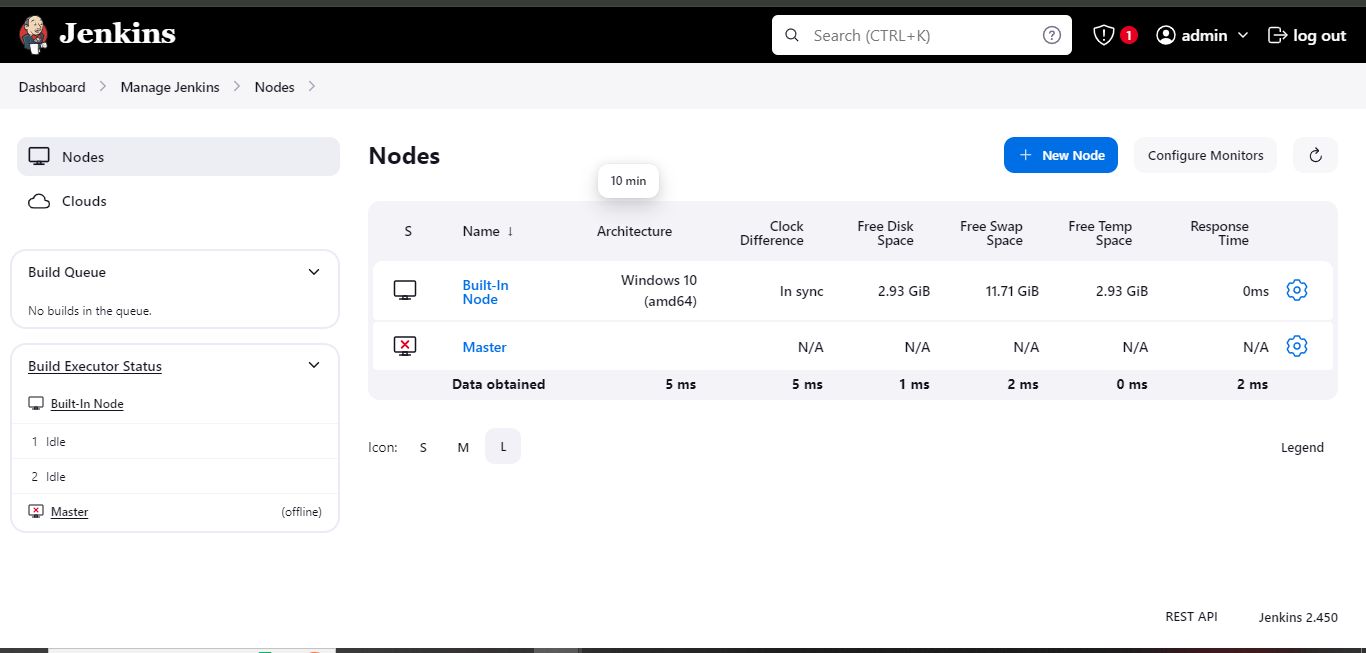
Your main Jenkins server is the Master. The Master’s job is to handle:

* Scheduling build jobs.
* Dispatching builds to the slaves for the actual execution.
* Monitor the slaves (possibly taking them online and offline as required).
* Recording and presenting the build results.
* A Master instance of Jenkins can also execute build jobs directly.

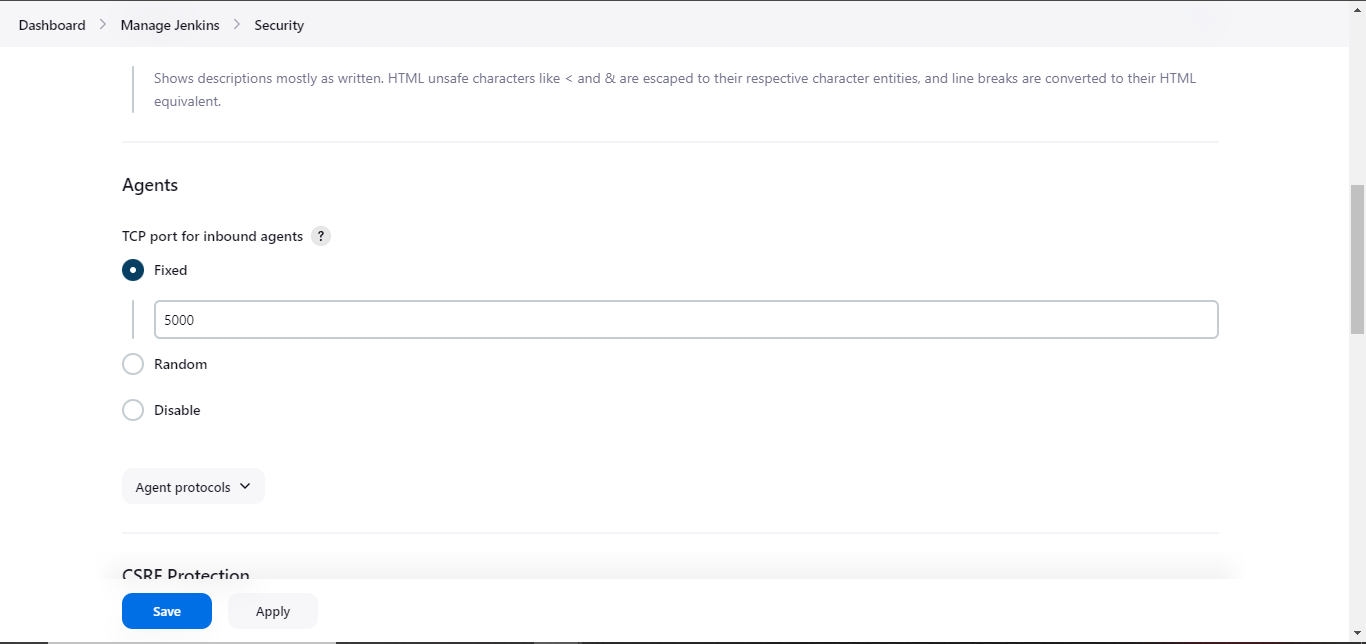
**Jenkins Slave**

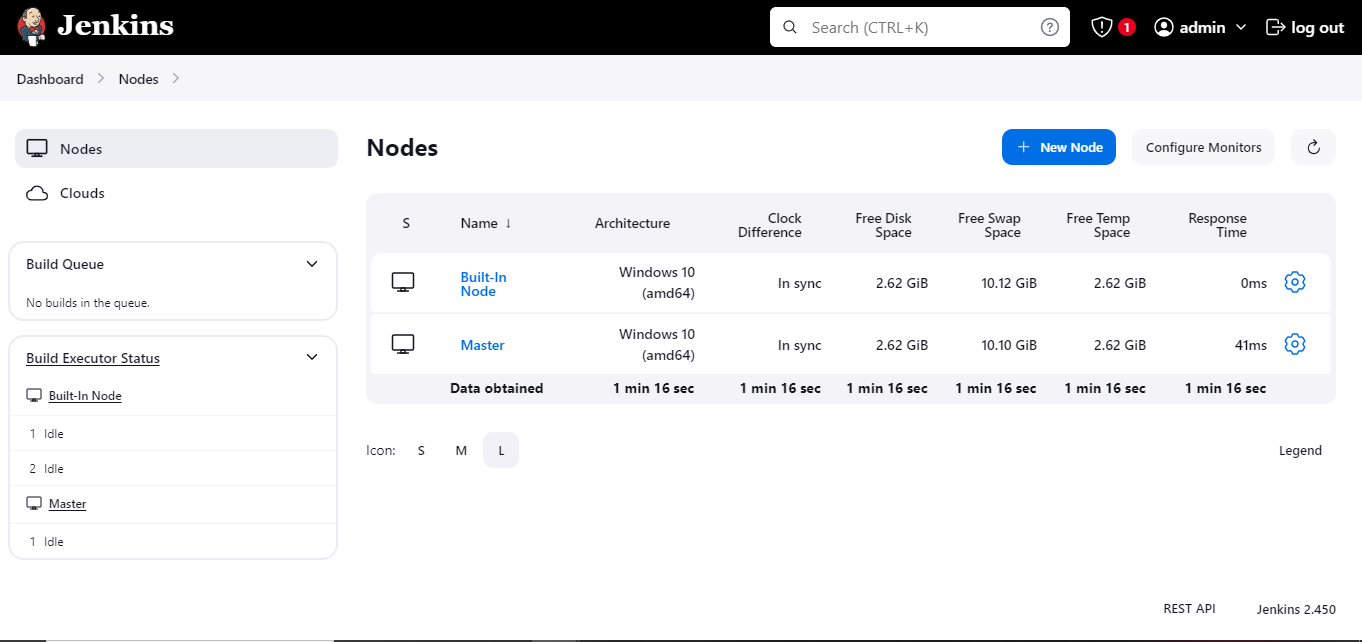
A Slave is a Java executable that runs on a remote machine. The following are the characteristics of Jenkins's Slaves:

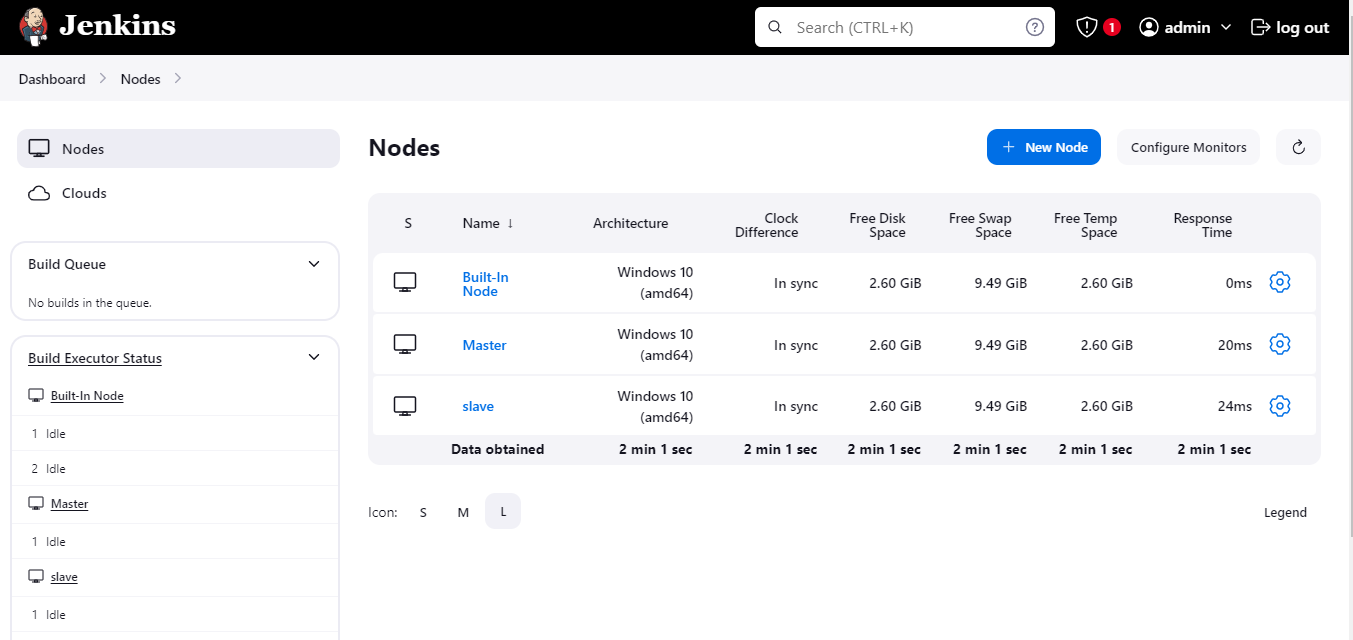
* It hears requests from the Jenkins Master instance.
* Slaves can run on a variety of operating systems.
* The job of a Slave is to do as they are told to, which involves executing build jobs dispatched by the Master.
* You can configure a project to always run on a particular Slave machine or a particular type of Slave machine, or simply let Jenkins pick the next available Slave.











Conclusion:

We have studied and understood Jenkins Master Slave Architecture and its benefits. We also studied and installed Jenkins, along with its prerequisites. Lastly, we created our first freestyle build job and scaled the standalone implementation by implementing slave nodes.