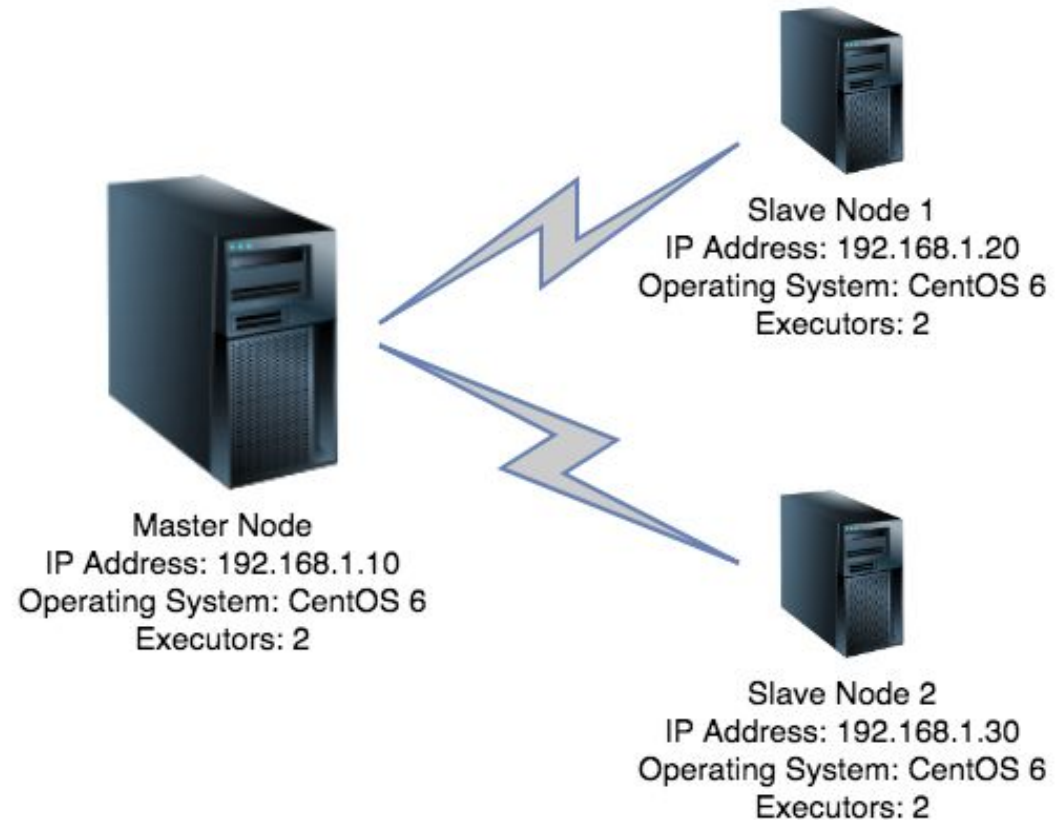# Distributed Builds on Jenkins

Agenda

- Connect agents to your Jenkins master and access them securely

- Configure freestyle projects to run on your agents

- Instruct pipelines to run on different agents in your build environment

Distributed Builds

- Refers to the allocation of different nodes (slaves) to run your build tasks

The Master-Slave Model

- Allows to configure distributed builds on Jenkins
- The master node is basically the node on which Jenkins is installed
- This is the administrative node that carries out most of the tasks

Slave Node 1
IP Address: 192.168.1.20
Operating System: CentOS 6
Executors: 2

Master Node
IP Address: 192.168.1.10
Operating System: CentOS 6
Executors: 2

Slave Node 2
IP Address: 192.168.1.30
Operating System: CentOS 6
Executors: 2

Setting up Slave Nodes

- Refer:
  - **Add slave nodes to Jenkins in "3. Continuous Integration setup.md"**

Configuring Tasks to Run on our Slaves

| General | Source Code Management | Build Triggers | Build Environment | Build | Post-build Actions |

Description

This project will only run on node 2

[Plain text] Preview

☐ Discard old builds

☐ GitHub project

☐ This project is parameterized

☐ Throttle builds

☐ Disable this project

☐ Execute concurrent builds if necessary

☑ Restrict where this project can be run

Label Expression | node2 |

Label node2 is serviced by 1 node. Permissions or other restrictions provided by plugins may prevent this job from running on those nodes.

Advanced...

Configuring Tasks to Run on our Slaves

Build Project

- We are going to run the project from the main Jenkins dashboard as opposed to the project dashboard.

- Click on the drop-down on the project name and select Build Now, as shown:

Build Project

- If we watch the executors at the bottom left, we can see that our project is running on node2, as we defined in the configuration

- Review the Console Output

Build Executor Status

🖥 **master**

  1 Idle

  2 Idle

🖥 **node1**

  1 Idle

  2 Idle

🖥 **node2**

  1 Idle

  2 node-2-project   #1

Running Pipelines on Different Nodes

- node {

-   stage("Might Run On Node 1 Or Master") {

-     printMessage("Running")

-     sh 'sleep 10'

-     sh 'hostname'

-    }

- }

- node('node2') {

-   stage("On Node 2") {

-     printMessage("Running On Node 2")

-     sh 'sleep 10'

-     sh 'hostname'

-    }

- }

- def printMessage(message) {   echo "${message}" }