

# JENKINS TASK 02

## 1) Configure 2 slave machines in jenkins master.

Dashboard > Nodes > slave-02 > Log

Status  
Delete Agent  
Configure  
Build History  
Load Statistics  
Script Console  
Log  
System Information  
Disconnect

Inbound agent connected from 52.55.121.40  
Remoting version: 3261.v9c670a\_4748a\_9  
Launcher: JNLPLauncher  
Communication Protocol: WebSocket  
This is a Unix agent  
Agent successfully connected and online

Dashboard > Nodes > slave-01

Status  
Delete Agent  
Configure  
Build History  
Load Statistics  
Script Console  
Log  
System Information  
Disconnect

Agent slave-01  
slave for java

Monitoring Data

Architecture	Linux (amd64)
Clock Difference	In sync
Free Disk Space	17.87 GiB
Free Swap Space	0 B
Free Temp Space	17.87 GiB
Response Time	61ms

Build Executor Status 0/2

Labels  
java

Projects tied to slave-01  
None

Nodes

S	Name	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
	Built-In Node	Linux (amd64)	In sync	17.51 GiB	0 B	17.51 GiB	0ms
	slave-01	Linux (amd64)	In sync	17.87 GiB	0 B	17.87 GiB	61ms
	slave-02	Linux (amd64)	In sync	17.87 GiB	0 B	17.87 GiB	109ms

Data obtained 3 min 17 sec 3 min 17 sec 3 min 17 sec 3 min 17 sec 3 min 17 sec 3 min 17 sec

## 2) Configure webhooks to jenkins job.

General Webhooks

Access

Collaborators

Moderation options

Code and automation

Webhooks allow external services to be notified when certain events happen. When the specified events happen, we'll send a POST request to each of the URLs you provide. Learn more in our [Webhooks Guide](#).

✓ http://100.24.51.45:8080/github-we... (all events)  
Last delivery was successful.

```
root@ip-10-0-0-159 ec2-user]# cd /var/lib/jenkins/workspace/
root@ip-10-0-0-159 workspace]# ls
webhook
root@ip-10-0-0-159 workspace]# cd webhook/
root@ip-10-0-0-159 webhook]# ll
total 24
-rwxr-xr-x 1 jenkins jenkins 715 Nov 28 11:12 calculator.sh
-rwxr-xr-x 1 jenkins jenkins 385 Nov 28 11:12 new_dir.sh
-rwxr-xr-x 1 jenkins jenkins 432 Nov 28 11:12 nginx-service.sh
-rwxr-xr-x 1 jenkins jenkins 506 Nov 28 11:12 nginx.sh
-rwxr-xr-x 1 jenkins jenkins 250 Nov 28 11:12 testline.sh
-rwxr-xr-x 1 jenkins jenkins 671 Nov 28 11:12 tomcat.sh
```

### 3) Configure poll scm and build periodical options in jenkins job.

Dashboard > webhook > Configuration

#### Configure

- General
- Source Code Management
- Build Triggers**
- Build Environment
- Build Steps
- Post-build Actions

☐ Trigger builds remotely (e.g., from scripts) ?

☐ Build after other projects are built ?

☐ Build periodically ?

☐ GitHub hook trigger for GITScm polling ?

☒ Poll SCM ?

Schedule ?

\*\*\*\*|

No schedules so will only run due to SCM changes if triggered by a post-commit hook

#### Build Executor Status

Executor	Build	Status
Built-In Node		
webhook	#4	1/2
slave-01		0/2
slave-02		0/2

```
root@ip-10-0-0-159 webhook]# ll
total 28
-rwxr-xr-x 1 jenkins jenkins 715 Nov 28 11:12 calculator.sh
-rwxr-xr-x 1 jenkins jenkins 385 Nov 28 11:12 new_dir.sh
-rwxr-xr-x 1 jenkins jenkins 432 Nov 28 11:12 nginx-service.sh
-rwxr-xr-x 1 jenkins jenkins 506 Nov 28 11:12 nginx.sh
-rw-r--r-- 1 jenkins jenkins  3 Nov 28 11:18 README.md
-rwxr-xr-x 1 jenkins jenkins 250 Nov 28 11:12 testline.sh
-rwxr-xr-x 1 jenkins jenkins 671 Nov 28 11:12 tomcat.sh
```

### 4) Take backup of jenkins server by using bash script.

```
#!/bin/bash
```

```
# Define the path to JENKINS_HOME and the backup directory
```

```
JENKINS_HOME="/var/lib/jenkins"
```

```
BACKUP_DIR="/var/lib/jenkins-backup"
```

```
BACKUP_NAME="jenkins-backup.tar"
```

```
# Create a backup directory if it doesn't exist
```

```
mkdir -p $BACKUP_DIR
```

```
# Create a timestamped archive of JENKINS_HOME
```

```
tar -cvf $BACKUP_DIR/$BACKUP_NAME $JENKINS_HOME
```

```
# Print a message indicating the backup is complete
```

```
echo "Backup completed: $BACKUP_DIR/$BACKUP_NAME"
```

```
# Add a cron job to schedule the backup script
```

```
(crontab -l 2>/dev/null; echo "0 0 * * * /path/to/jenkins-backup.sh") | crontab -
```

```
# Print a message indicating the cron job has been added
```

```
echo "Cron job added to run the backup script daily at midnight"
```

```
Backup completed: /var/lib/jenkins-backup/jenkins-backup.tar
Cron job added to run the backup script daily at midnight
[root@ip-10-0-0-159 ec2-user]# cd /var/lib/jenkins-backup
[root@ip-10-0-0-159 jenkins-backup]# ll
total 321112
-rw-r--r-- 1 root root 328816640 Nov 28 11:51 jenkins-backup.tar
[root@ip-10-0-0-159 jenkins-backup]#
```

## 5) Take backup of jenkins using rethin backup plugin.

Dashboard > Manage Jenkins > System > ←

### ThinBackup Configuration

Backup directory ?  
/var/lib/jenkins

Backup schedule for full backups ?

Specifies the execution schedule in a cron notation when a full backup should be done. Look at [Wikipedia](#) for more information.

**Example:**

CRON Expression	Description
0 12 * * 1-5	Executes on 12:00 every weekday (Mo-Fr)

(from ThinBackup)

Backup schedule for differential backups ?

Specifies the execution schedule in a cron notation when a differential backup should be done. Look at [Wikipedia](#) for more information.

A differential backup stores only complete files whose modification is done after the last full backup.

**NOTE:**  
For a differential backup at least one full backup is needed. If no full backup is available it will be done instead of the first differential backup.

(from ThinBackup)

>>RESTART JENKINS AFTER CONFIGURING THE TINBACKUP CONFIGURATIONS

The Jenkins Controller is preparing for shutdown. No new builds can be started.

### ThinBackup

Settings are now integrated in global configuration.

>>CHECKING THE RESTORE IF BACKUP IS AVAILABLE

### Restore Configuration

#### Restore options

Restore backup from: ?

2024-11-28 12:46

2024-11-28 12:46

☐ Restore next build number file (if found in backup) ?