


1- Install role based authorization plugin.



# Jenkins

abdelrahman
log out

Dashboard

Plugin Manager

[Back to Dashboard](#)
[Manage Jenkins](#)

Updates


Available


Installed

Advanced

Enabled	Name	Version	Previously installed version	Uninstall
<input checked="" type="checkbox"/>	<b>Caffeine API Plugin</b> <a href="#">Caffeine</a> api plugin for use by other Jenkins plugins.	2.9.2-29.v717aac953ff3		<button>Uninstall</button>
<input checked="" type="checkbox"/>	<b>Matrix Authorization Strategy Plugin</b> Offers matrix-based security authorization strategies (global and per-project).	2.6.8		<button>Uninstall</button>
<input checked="" type="checkbox"/>	<b>Role-based Authorization Strategy</b> Enables user authorization using a Role-Based strategy. Roles can be defined globally or for particular jobs or nodes selected by regular expressions.	3.2.0		<button>Uninstall</button>

2 - Create new user & create read role and assign it to that user.

Jenkins

 [abdelrahman](#) [log out](#)

Dashboard

New Item

People

Build History

My Views

Build Queue

No builds in the queue.

Build Executor Status

1 Idle

2 Idle

Welcome to Jenkins!

This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.

Start building your software project

Create a job

Jenkins


 ?
 admin
 log out

---

Dashboard ▾ > Manage and Assign Roles

- New Item
- People
- Build History
- Manage Jenkins
- My Views
- Lockable Resources
- New View

**Build Queue** ^

No builds in the queue.

**Build Executor Status** ^

- 1 Idle
- 2 Idle

## Manage Roles

### Global roles

Role	Overall					Credentials					Agent						
	Administer	Read	Create	Delete	ManageDomains	Update	View	Build	Configure	Connect	Create	Delete	Disconnect	Provision	Build	Cancel	Config
admin	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
developer	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Role to add

### Item roles

Role	Pattern	Credentials					Job					Run					
		Create	Delete	ManageDomains	Update	View	Build	Cancel	Configure	Create	Delete	Discover	Move	Read	Workspace	Delete	Replay

Jenkins

search

admin

log out

Dashboard

Manage and Assign Roles

New Item

People

Build History

Manage Jenkins

My Views

Lockable Resources

New View

Build Queue

No builds in the queue.

Build Executor Status

1 Idle

2 Idle

Assign Roles

Global roles

User/group	admin	developer
abdelrahman	<input type="checkbox"/>	<input checked="" type="checkbox"/>
admin	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Anonymous	<input type="checkbox"/>	<input type="checkbox"/>

User/group to add

Add

Item roles

User/group
Anonymous

3- Create a free-style pipeline and link it to private git repository (Create directory and create file with "hello world" inside)

Dashboard

freestyle-demo

General

Source Code Management

Build Triggers

Build Environment

Build

Post-build Actions

Advanced...

Source Code Management

None

Git

Repositories

Repository URL

https://github.com/abdelrahman179/demo-jenkins.git

Credentials

abdelrahman179/\*\*\*\*\*

Add

Advanced...

Add Repository

Branches to build

Save

Apply

Dashboard freestyle-demo

General

Source Code Management

Build Triggers

Build Environment

Build

Post-build Actions

☐ Use secret text(s) or file(s)
 ☐ Abort the build if it's stuck
 ☐ Add timestamps to the Console Output
 ☐ Inspect build log for published Gradle build scans
 ☐ With Ant

Build

Execute shell

Command

```
ls
mkdir sp_bootcamp
cd sp_bootcamp
touch demo
echo "Hello World !!" >> demo
```

See [the list of available environment variables](#)

Advanced...

Add build step

Post-build Actions

Save

Apply

Dashboard freestyle-demo #8

Back to Project

Status

Changes

Console Output

View as plain text

Edit Build Information

Delete build '#8'

Git Build Data

Previous Build

Console Output

Started by user [admin](#)  
 Running as SYSTEM  
 Building in workspace /var/jenkins\_home/workspace/freestyle-demo  
 [WS-CLEANUP] Deleting project workspace...  
 [WS-CLEANUP] Deferred wipeout is used...  
 [WS-CLEANUP] Done  
 The recommended git tool is: NONE  
 using credential github  
 Cloning the remote Git repository  
 Cloning repository <https://github.com/abdelrahman179/demo-jenkins.git>  
 > git init /var/jenkins\_home/workspace/freestyle-demo # timeout=10  
 Fetching upstream changes from <https://github.com/abdelrahman179/demo-jenkins.git>  
 > git --version # timeout=10  
 > git --version # 'git version 2.30.2'  
 using GIT\_ASKPASS to set credentials  
 > git fetch --tags --force --progress -- <https://github.com/abdelrahman179/demo-jenkins.git>  
 +refs/heads/\*:refs/remotes/origin/\* # timeout=10  
 > git config remote.origin.url <https://github.com/abdelrahman179/demo-jenkins.git> # timeout=10  
 > git config --add remote.origin.fetch +refs/heads/\*:refs/remotes/origin/\* # timeout=10  
 Avoid second fetch  
 > git rev-parse refs/remotes/origin/main^{commit} # timeout=10  
 Checking out Revision 333f73cb204bb839291ba647ec926393451686fa (refs/remotes/origin/main)  
 > git config core.sparsecheckout # timeout=10  
 > git checkout -f 333f73cb204bb839291ba647ec926393451686fa # timeout=10  
 Commit message: "Initial commit"  
 > git rev-list --no-walk 333f73cb204bb839291ba647ec926393451686fa # timeout=10  
 [freestyle-demo] \$ /bin/sh -xe /tmp/jenkins8827553752418027954.sh  
 + ls  
 README.md  
 + mkdir sp\_bootcamp

Jenkins

search

admin log out

Dashboard

New Item

People

Build History

Project Relationship

Check File Fingerprint

Manage Jenkins

My Views

Lockable Resources

New View

Build Queue

No builds in the queue.

Build Executor Status

All

freestyle-demo

5 min 5 sec - #8

15 min - #6

1.1 sec

Icon: S M L

Legend

Atom feed for all

Atom feed for failures

Atom feed for just latest builds

Dashboard
freestyle-demo
#8

View as plain text

Edit Build Information

Delete build '#8'

Git Build Data

Previous Build

```

The recommended git tool is: NONE
using credential github
Cloning the remote Git repository
Cloning repository https://github.com/abdelrahman179/demo-jenkins.git
> git init /var/jenkins_home/workspace/freestyle-demo # timeout=10
Fetching upstream changes from https://github.com/abdelrahman179/demo-jenkins.git
> git --version # timeout=10
> git --version # 'git version 2.30.2'
using GIT_ASKPASS to set credentials
> git fetch --tags --force --progress -- https://github.com/abdelrahman179/demo-jenkins.git
+refs/heads/*:refs/remotes/origin/* # timeout=10
> git config remote.origin.url https://github.com/abdelrahman179/demo-jenkins.git # timeout=10
> git config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
Avoid second fetch
> git rev-parse refs/remotes/origin/main^{commit} # timeout=10
Checking out Revision 333f73cb204bb839291ba647ec926393451686fa (refs/remotes/origin/main)
> git config core.sparsecheckout # timeout=10
> git checkout -f 333f73cb204bb839291ba647ec926393451686fa # timeout=10
Commit message: "Initial commit"
> git rev-list --no-walk 333f73cb204bb839291ba647ec926393451686fa # timeout=10
[jenkins@47472a1a1b2a:~/workspace/freestyle-demo]$ /bin/sh -xe /tmp/jenkins8827553752418027954.sh
+ ls
README.md
+ mkdir sp_bootcamp
+ cd sp_bootcamp
+ touch demo
+ echo Hello World !!
Finished: SUCCESS

```

```

logs
jenkins@47472a1a1b2a:~$ cd workspace/
jenkins@47472a1a1b2a:~/workspace$ ls
freestyle-demo  freestyle-demo@tmp
jenkins@47472a1a1b2a:~/workspace$ cd freestyle-demo
jenkins@47472a1a1b2a:~/workspace/freestyle-demo$ ls
README.md  sp_bootcamp
jenkins@47472a1a1b2a:~/workspace/freestyle-demo$ cd sp_bootcamp/
jenkins@47472a1a1b2a:~/workspace/freestyle-demo/sp_bootcamp$ ls
demo
jenkins@47472a1a1b2a:~/workspace/freestyle-demo/sp_bootcamp$ cat demo
Hello World !!
jenkins@47472a1a1b2a:~/workspace/freestyle-demo/sp_bootcamp$

```

#### 4- Create declarative in Jenkins GUI pipeline for your own repository to do "ls"

Dashboard
pipeline-demo

General
Build Triggers
Advanced Project Options
Pipeline

SCM

Git

Repositories

Repository URL
https://github.com/abdelrahman179/demo-jenkins.git

Credentials
abdelrahman179/\*\*\*\*\*
Add

Advanced...
Add Repository

Branches to build

Branch Specifier (blank for 'any')
\*/main
X

Add Branch

Save
Apply

Dashboard > pipeline-demo >

GeneralBuild TriggersAdvanced Project OptionsPipeline

Add Branch

Repository browser  
(Auto) ?

Additional Behaviours  
Add ?

Script Path  
Jenkinsfile ?

☒ Lightweight checkout ?

Pipeline Syntax

SaveApply

REST APIJenkins 2.

<> CodeIssuesPull requestsActionsProjectsWikiSecurityInsightsSettings

demo-jenkins / Jenkinsfile in mainCancel changes

<> Edit filePreview changesSpaces4No wrap

```
1 pipeline {
2   agent any
3
4
5   stages {
6     stage('Build') {
7       steps {
8         // Get some code from a GitHub repository
9         git 'https://github.com/abdelrahman179/demo-jenkins.git'
10
11         // Run Maven on a Unix agent.
12         sh """ls
13           mkdir declarative_demo
14           touch demo_file
15         """
16         // To run Maven on a Windows agent, use
17         // bat "mvn -Dmaven.test.failure.ignore=true clean package"
18       }
19     }
20   }
21 }
22
```

5- create scripted in Jenkins GUI pipeline for your own repository to do "ls"

The screenshot shows the Jenkins Pipeline configuration page with the 'Scripted Pipeline' tab selected. The 'Definition' section is set to 'Pipeline script'. The script is as follows:

```
1 git https://github.com/jglick/simple-maven-project-with-tests.git
2 // Get the Maven tool.
3 // ** NOTE: This 'M3' Maven tool must be configured
4 // ** in the global configuration.
5 mvnHome = tool 'M3'
6
7
8
9 }
10
11 stage('Build') {
12 // Run the maven build
13 withEnv(["MVN_HOME=${mvnHome}"]) {
14 if (isUnix()) {
15 sh """
16 ls
17 mkdir demo_scripted
18 touch demo_scripted
19 """
20 } else {
21 bat("${MVN_HOME}\bin\mvn" -Dmaven.test.failure.ignore clean package/)
22 }
23 }
24
25 stage('Results') {
```

Below the script, the 'Use Groovy Sandbox' checkbox is checked. At the bottom, there are 'Save' and 'Apply' buttons.

6- create the same with jenkinsfile in your branches as multi-branch pipeline.










The screenshot shows the Jenkins 'Scan Multibranch Pipeline Log' page. The left sidebar contains a navigation menu with options: Up, Status, Configure, Scan Multibranch Pipeline Now, Scan Multibranch Pipeline Log (selected), View as plain text, Multibranch Pipeline Events, Delete Multibranch Pipeline, People, Build History, Project Relationship, Check File Fingerprint, Rename, and Pipeline Syntax.

The main content area displays the log for the 'Scan Multibranch Pipeline' job. The log starts with 'Started' and shows the following steps:

- [Wed Nov 10 23:05:22 UTC 2021] Starting branch indexing...
- > git --version # timeout=10
- > git --version # 'git version 2.30.2'
- using GIT\_ASKPASS to set credentials
- > git ls-remote --symref -- https://github.com/abdelrahman179/demo-jenkins.git # timeout=10
- > git rev-parse --resolve-git-dir /var/jenkins\_home/caches/git-b2513134f4ad8c1499366fb246a92152/.git # timeout=10
- Setting origin to https://github.com/abdelrahman179/demo-jenkins.git
- > git config remote.origin.url https://github.com/abdelrahman179/demo-jenkins.git # timeout=10
- Fetching & pruning origin...
- Listing remote references...
- > git config --get remote.origin.url # timeout=10
- > git --version # timeout=10
- > git --version # 'git version 2.30.2'
- using GIT\_ASKPASS to set credentials
- > git ls-remote -h -- https://github.com/abdelrahman179/demo-jenkins.git # timeout=10
- Fetching upstream changes from origin
- > git config --get remote.origin.url # timeout=10
- using GIT\_ASKPASS to set credentials
- > git fetch --tags --force --progress --prune -- origin +refs/heads/\*:refs/remotes/origin/\* # timeout=10
- Checking branches...
- Checking branch demo
- 'Jenkinsfile' found
- Met criteria
- Did not schedule build for branch: demo
- Checking branch main
- 'Jenkinsfile' found
- Met criteria
- Did not schedule build for branch: main
- Processed 2 branches

A progress bar is visible at the top right of the log area.

Dashboard







-  New Item
-  People
-  Build History
-  Project Relationship
-  Check File Fingerprint
-  Manage Jenkins
-  My Views
-  Lockable Resources
-  New View

Build Queue

No builds in the queue.

Build Executor Status

 add description

All						
S	W	Name	Last Success	Last Failure	Last Duration	
		freestyle-demo	53 min - #8	1 hr 3 min - #6	1.1 sec	
		multi-branch-demo	32 sec - log	N/A	2.7 sec	

Icon: S M L

Legend

 Atom feed for all

 Atom feed for failures

 Atom feed for just latest builds