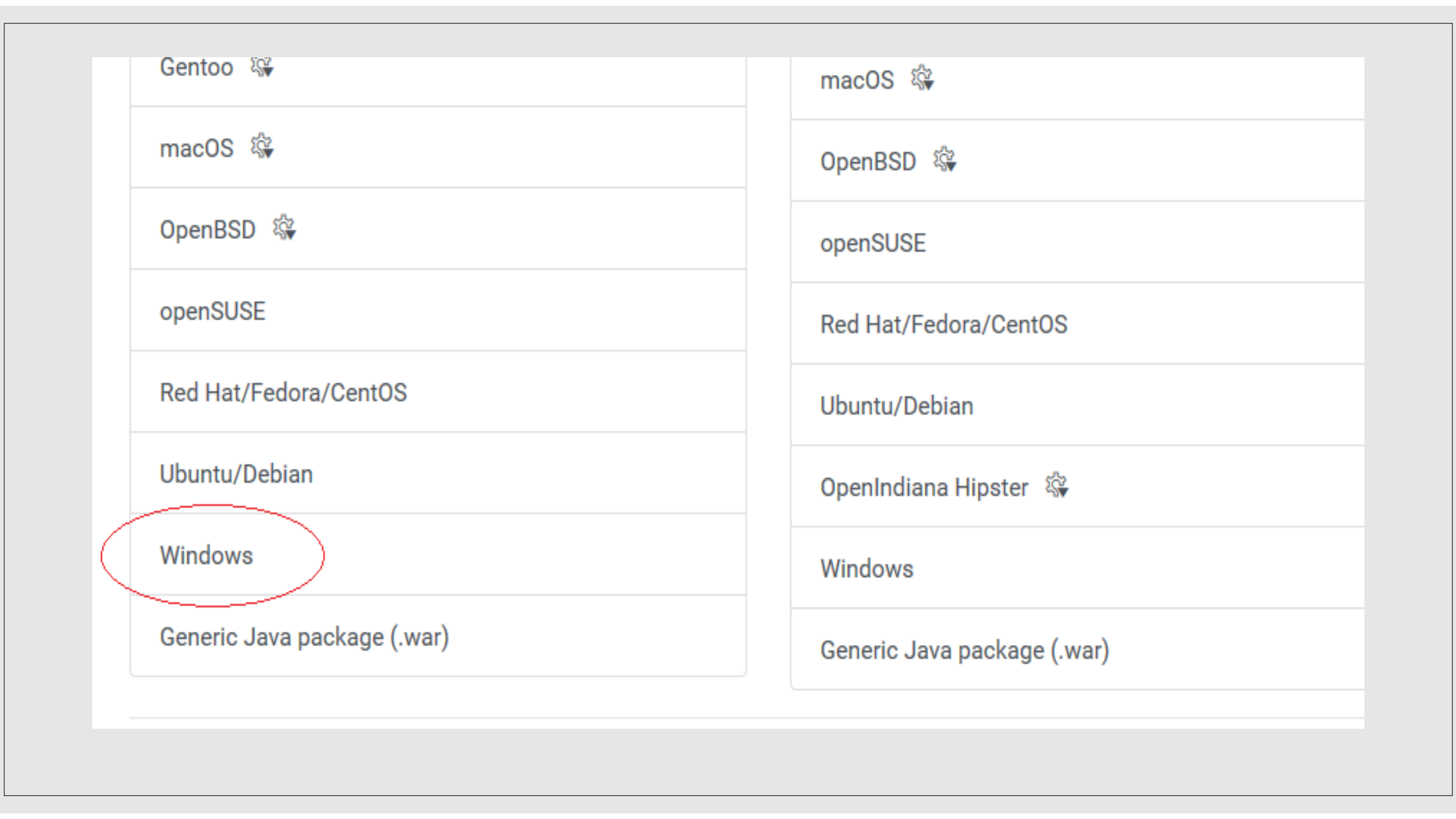
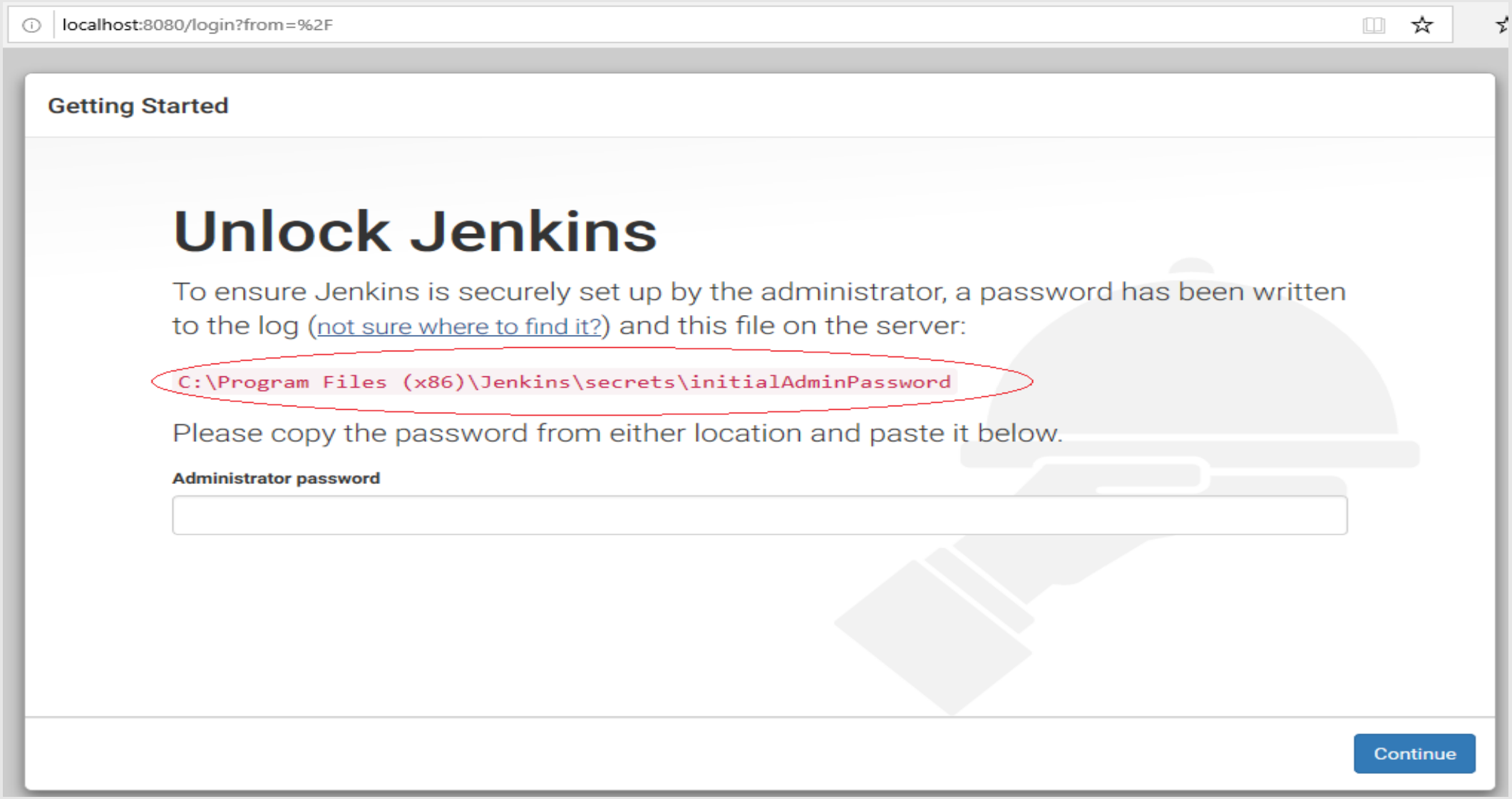
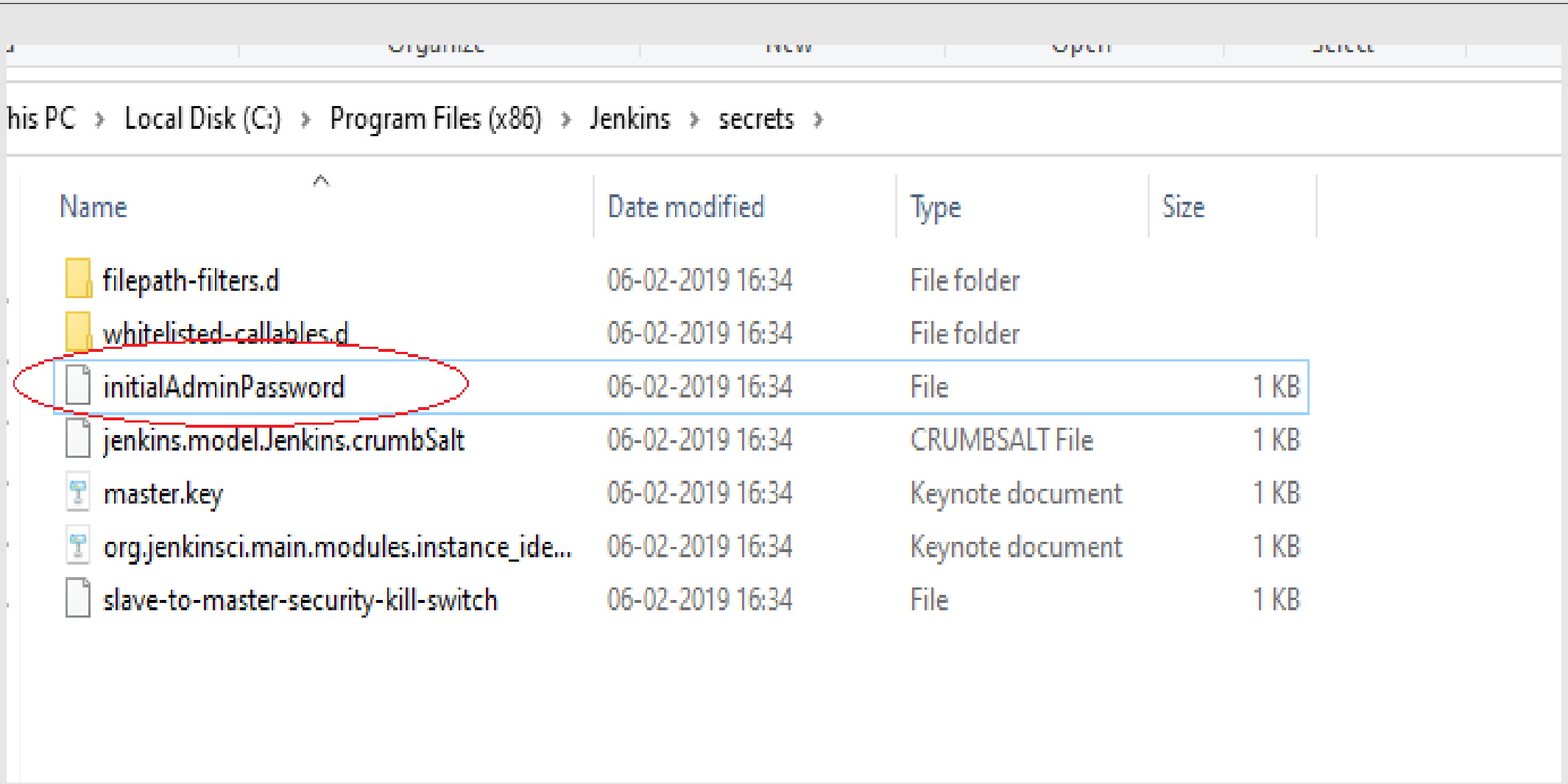
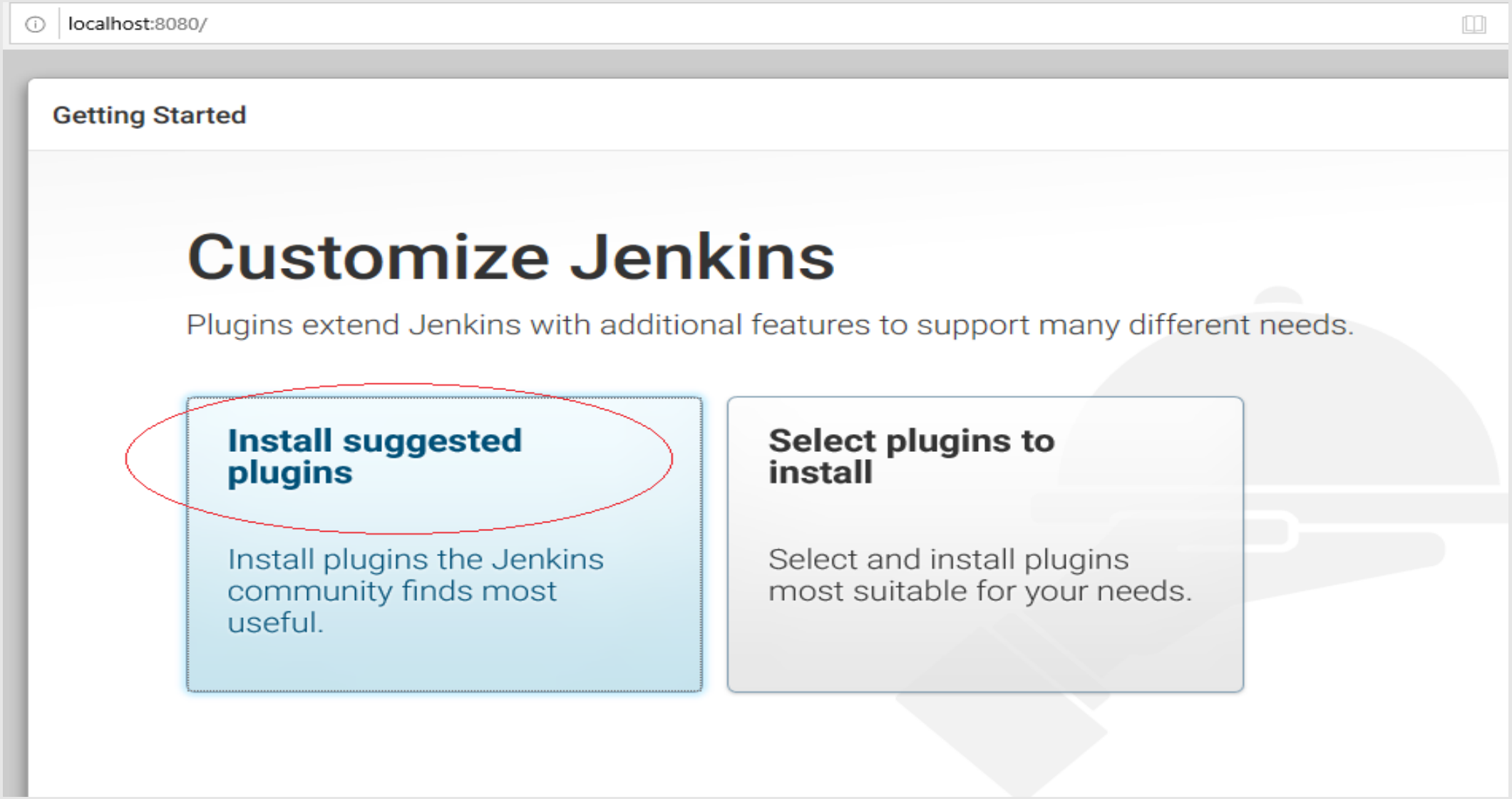
JENKINS_LAB

Download Jenkins for windows https://jenkins.io/download/


Gentoo
macOS
OpenBSD
openSUSE
Red Hat/Fedora/CentOS
Ubuntu/Debian
Windows
Generic Java package (.war)
macOS
OpenBSD
openSUSE
Red Hat/Fedora/CentOS
Ubuntu/Debian
OpenIndiana Hipster
Windows
Generic Java package (.war)


localhost:8080/login?from=%2F
Getting Started
Unlock Jenkins
To ensure Jenkins is securely set up by the administrator, a password has been written to the log (not sure where to find it?) and this file on the server:
C:\Program Files (x86)\Jenkins\secrets\initialAdminPassword
Please copy the password from either location and paste it below.
Administrator password
Continue

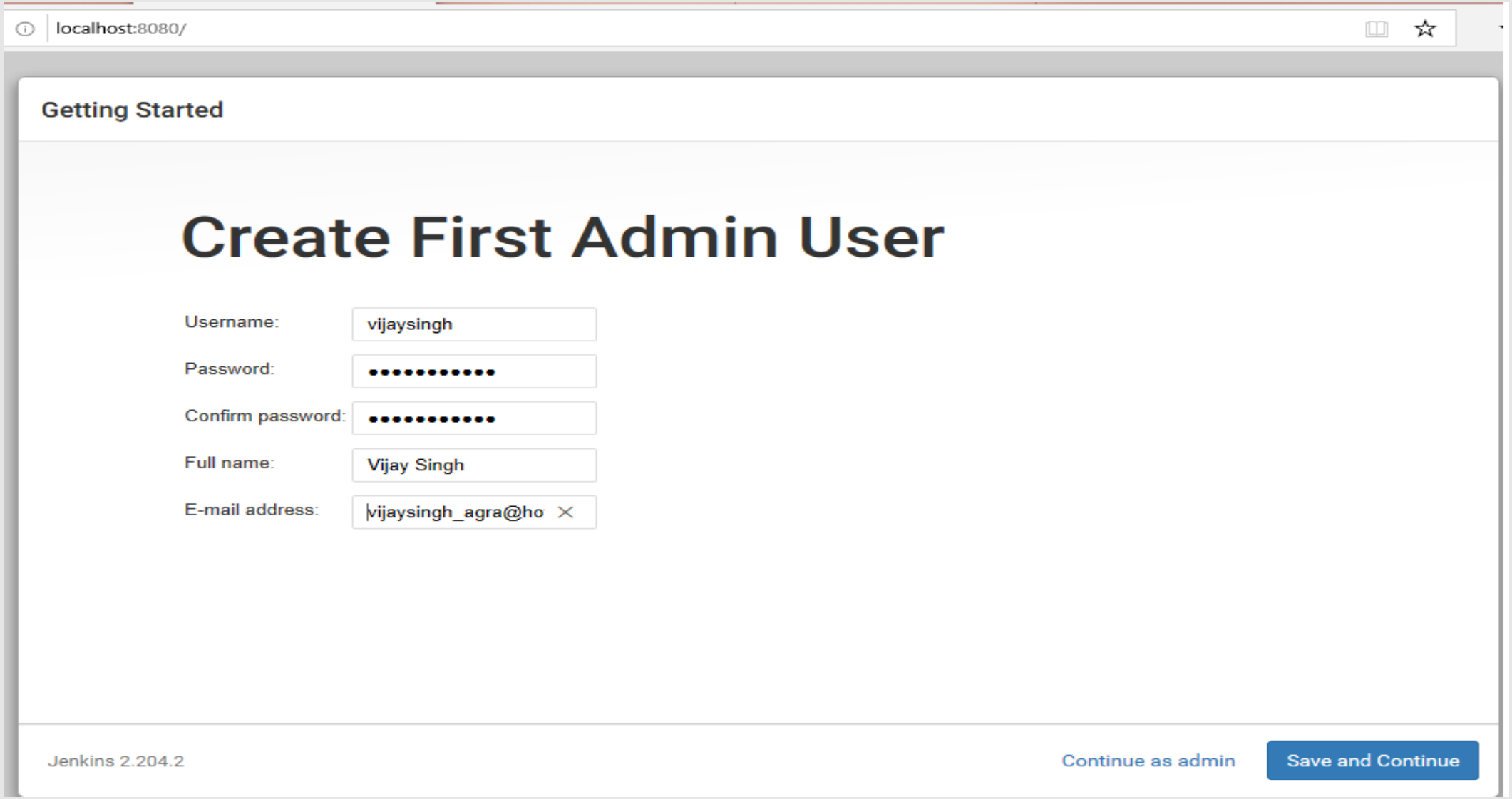

his PC › Local Disk (C:) › Program Files (x86) › Jenkins › secrets ›
Name
Date modified
Type
Size
filepath-filters.d
06-02-2019 16:34
File folder
whitelisted-callables.d
06-02-2019 16:34
File folder
initialAdminPassword
06-02-2019 16:34
File
1 KB
jenkins.model.Jenkins.crumbSalt
06-02-2019 16:34
CRUMBSALT File
1 KB
master.key
06-02-2019 16:34
Keynote document
1 KB
org.jenkinsci.main.modules.instance_ide...
06-02-2019 16:34
Keynote document
1 KB
slave-to-master-security-kill-switch
06-02-2019 16:34
File
1 KB

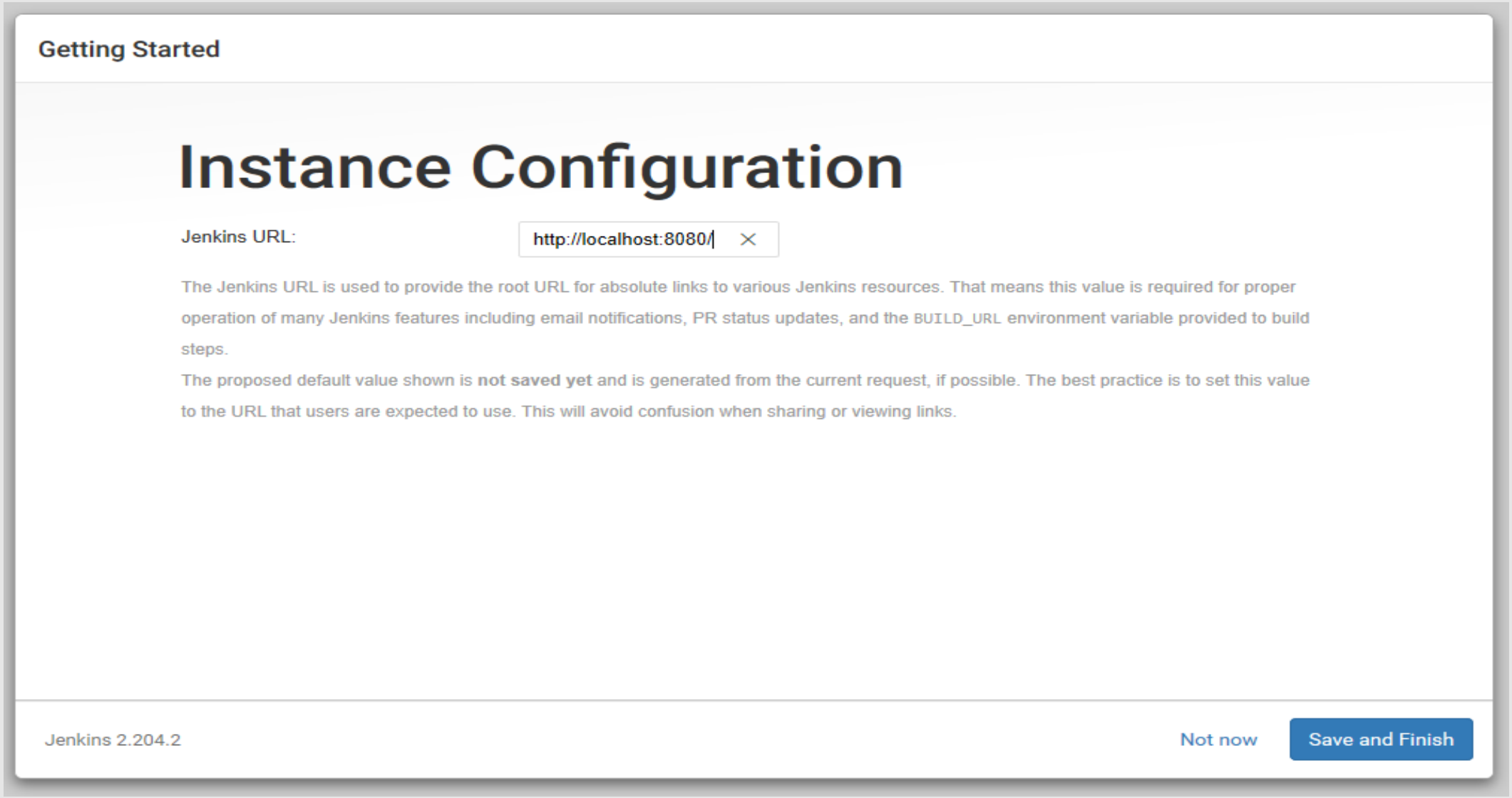

localhost:8080/
Getting Started
Customize Jenkins
Plugins extend Jenkins with additional features to support many different needs.
Install suggested plugins
Install plugins the Jenkins community finds most useful.
Select plugins to install
Select and install plugins most suitable for your needs.

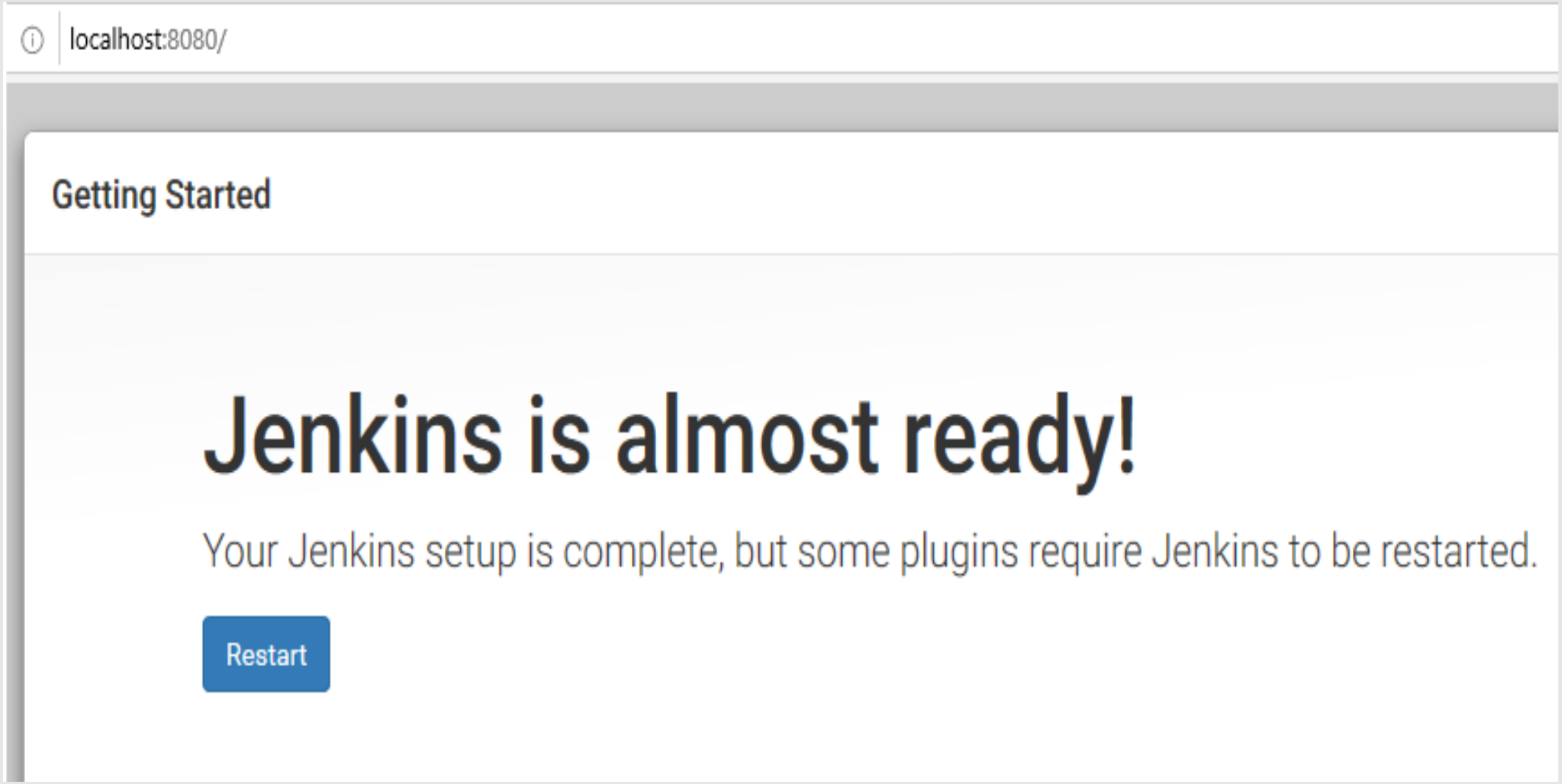
Getting Started

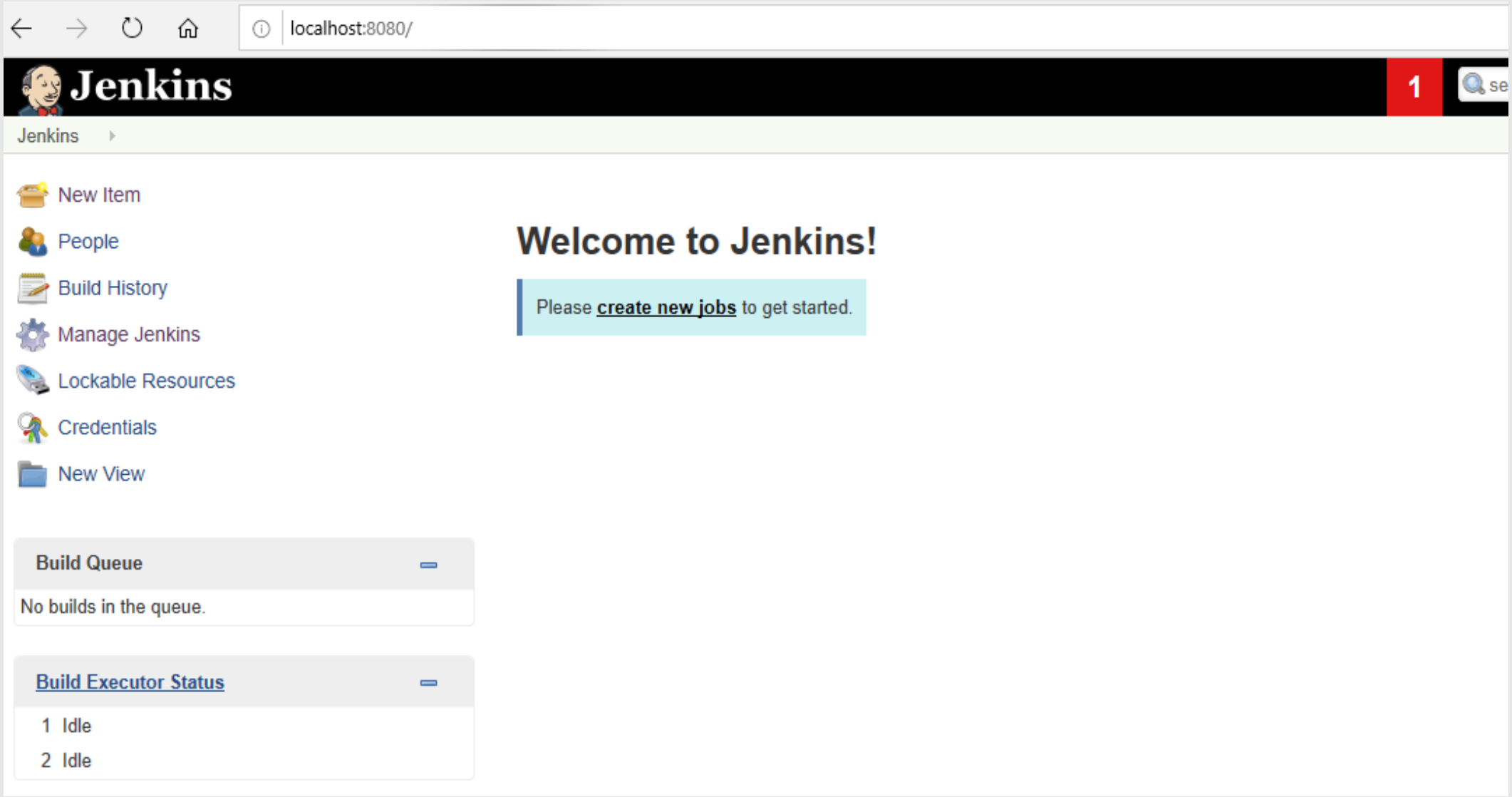
✘ Folders	✔ OWASP Markup Formatter	✔ Build Timeout	✘ Credentials Binding
✔ Timestamper	✘ Workspace Cleanup	✔ Ant	✔ Gradle
Pipeline	GitHub Branch Source	Pipeline: GitHub Groovy Libraries	✔ Pipeline: Stage View
Git	Subversion	SSH Slaves	Matrix Authorization Strategy
PAM Authentication	LDAP	Email Extension	✔ Mailer

```
4.x API
** Display URL API
Mailer
** Pipeline: Basic Ste
Gradle
** Pipeline: Milestone
** Jackson 2 API
** Pipeline: Input Ste
** Pipeline: Stage Ste
** Pipeline Graph Anal
** Pipeline: REST API
** JavaScript GUI Lib:
bundle
** JavaScript GUI Lib:
bundle
Pipeline: Stage View
** Pipeline: Build Ste
** Pipeline: Model API
** Pipeline: Declarati
```


localhost:8080/
Getting Started
Create First Admin User
Username:
vijaysingh
Password:
Confirm password:
Full name:
Vijay Singh
E-mail address:
vijaysingh_agra@ho
Jenkins 2.204.2
Continue as admin
Save and Continue


Getting Started
Instance Configuration
Jenkins URL:
http://localhost:8080/
The Jenkins URL is used to provide the root URL for absolute links to various Jenkins resources. That means this value is required for proper operation of many Jenkins features including email notifications, PR status updates, and the BUILD_URL environment variable provided to build steps.
The proposed default value shown is not saved yet and is generated from the current request, if possible. The best practice is to set this value to the URL that users are expected to use. This will avoid confusion when sharing or viewing links.
Jenkins 2.204.2
Not now
Save and Finish


localhost:8080/
Getting Started
Jenkins is almost ready!
Your Jenkins setup is complete, but some plugins require Jenkins to be restarted.
Restart


localhost:8080/
Jenkins
1
Jenkins
New Item
People
Build History
Manage Jenkins
Lockable Resources
Credentials
New View
Welcome to Jenkins!
Please create new jobs to get started.
Build Queue
No builds in the queue.
Build Executor Status
1 Idle
2 Idle

Enter an item name

» This field cannot be empty, please enter a valid name

Freestyle project

This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.

Pipeline

Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.

Multi-configuration project

Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.

Folder

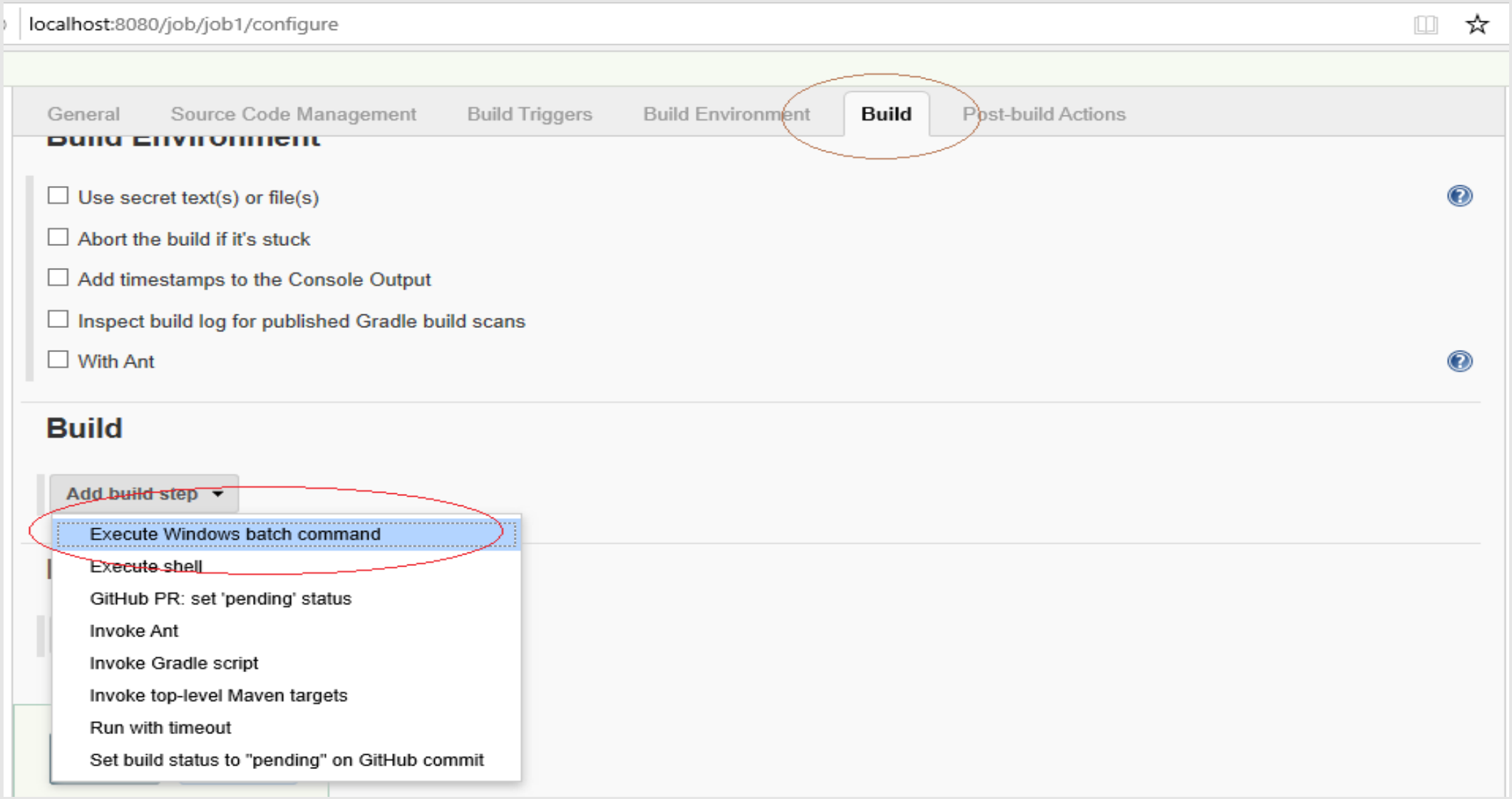
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.

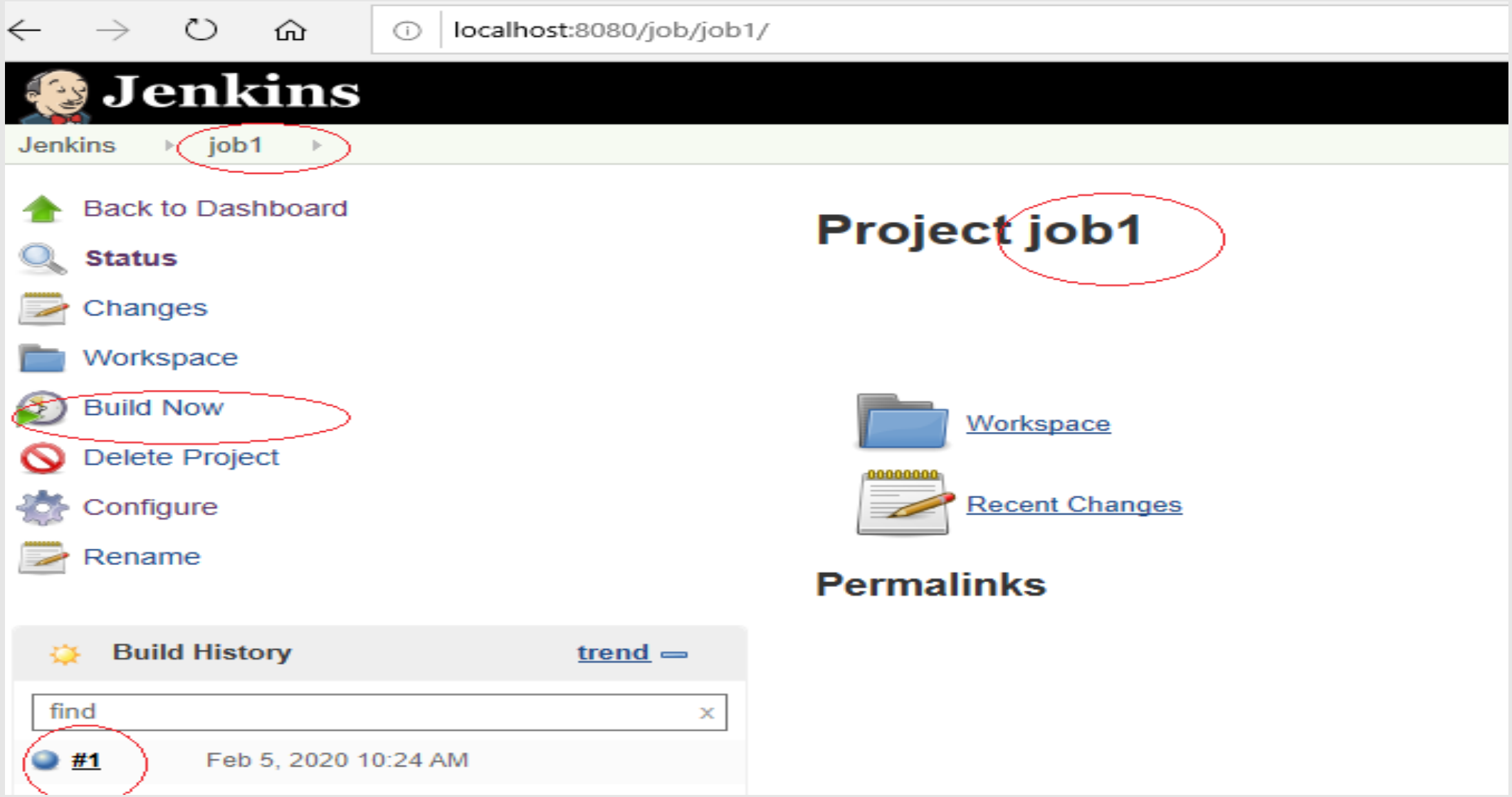
GitHub Organization

Scans a GitHub organization (or user account) for all repositories matching some defined markers.

Multibranch Pipeline

Creates a set of Pipeline projects according to detected branches in one SCM repository.


localhost:8080/job/job1/configure
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
Add build step
Execute Windows batch command
Execute shell
GitHub PR: set 'pending' status
Invoke Ant
Invoke Gradle script
Invoke top-level Maven targets
Run with timeout
Set build status to "pending" on GitHub commit


localhost:8080/job/job1/
Jenkins
Jenkins
job1
Back to Dashboard
Status
Changes
Workspace
Build Now
Delete Project
Configure
Rename
Project job1
Workspace
Recent Changes
Permalinks
Build History
trend
find
#1
Feb 5, 2020 10:24 AM

################################## **Task 1** ##################################

1. Create new job or item
2. Enter an item name :Job1
3. Choose freestyle project
4. OK
5. configure the Job1
6. Choose Build section
7. Add build step
8. Execute shell or Execute windows batch command(on case of windows)
9. In command field type: echo "My first task in Jenkins"
10. Save
11. For executing the Job1, go to Job1 and click on "Build Now"

//Here Build the Job means executing the Job

12. Check the Out using Build History.

Task 2##################

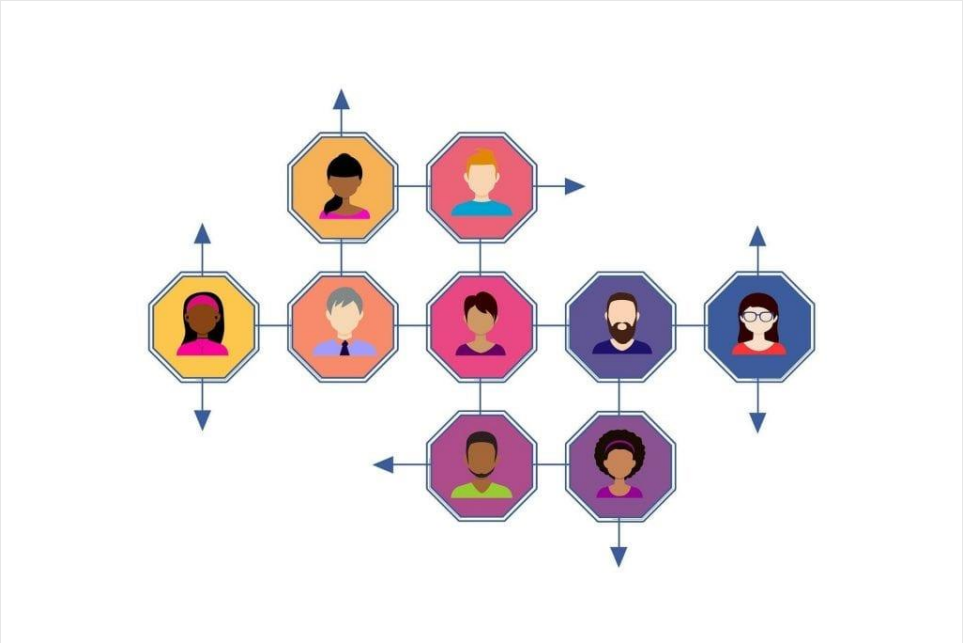
1. Create Job2
2. Freestyle Project
3. OK
4. Configure
5. Build
6. Add build step
7. Execute shell
8. Command: date>> /tmp/date.txt
9. Save
10. Build Now

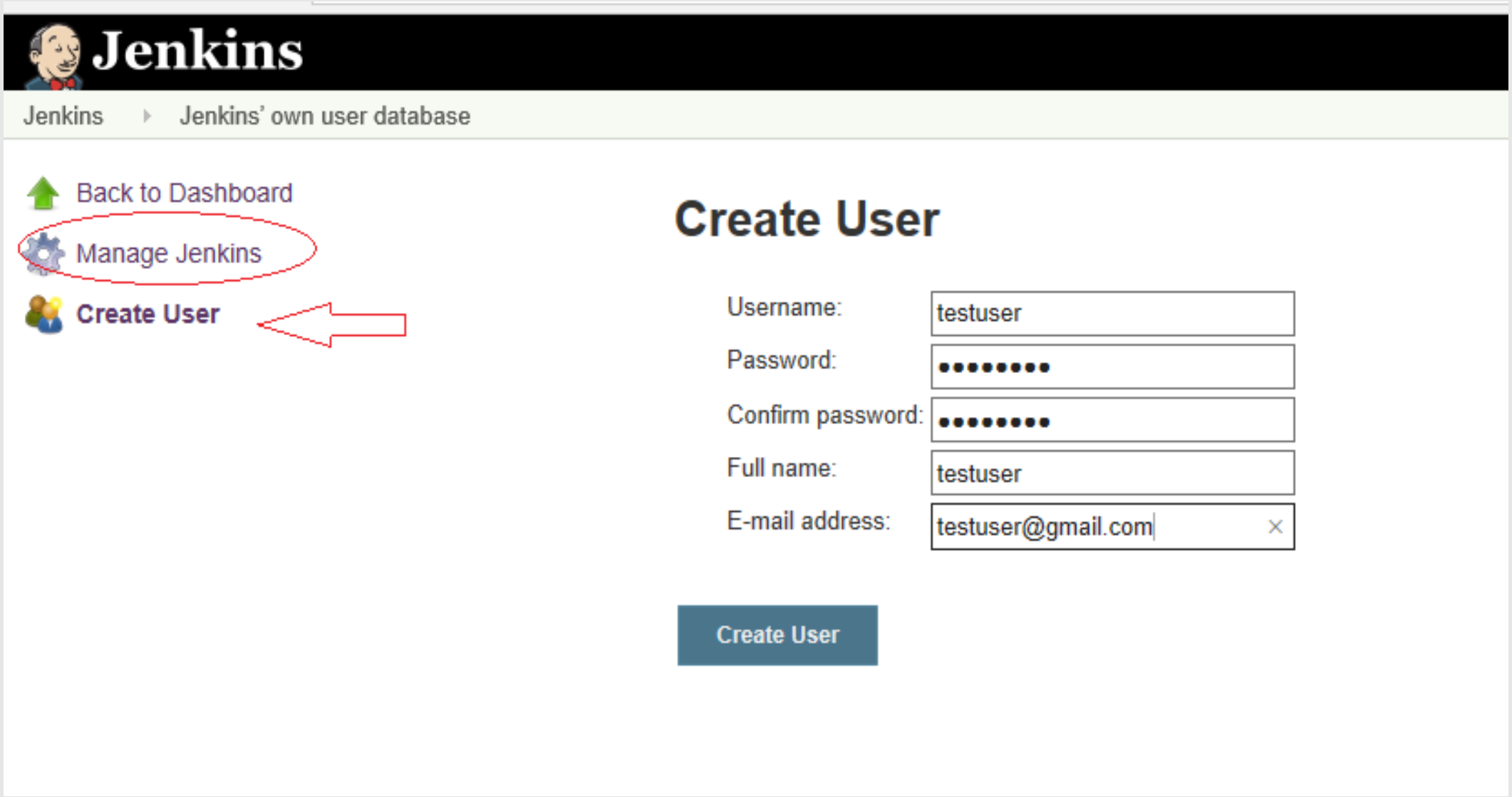
// check the date.txt in tmp

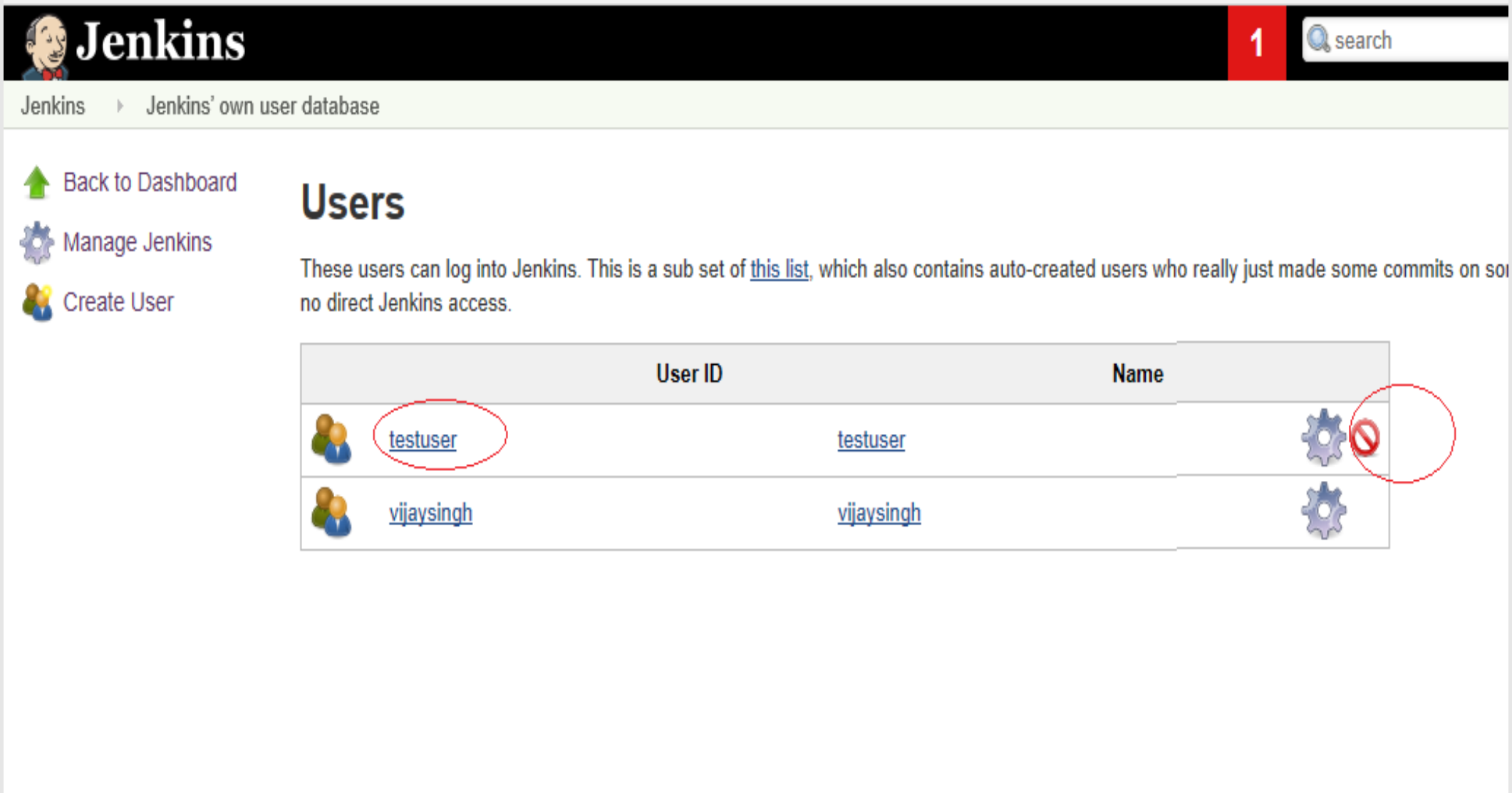
######### Task 3 ########

//Create User

1. Manage Jenkins
2. Manage Users
3. Create User (create some users)



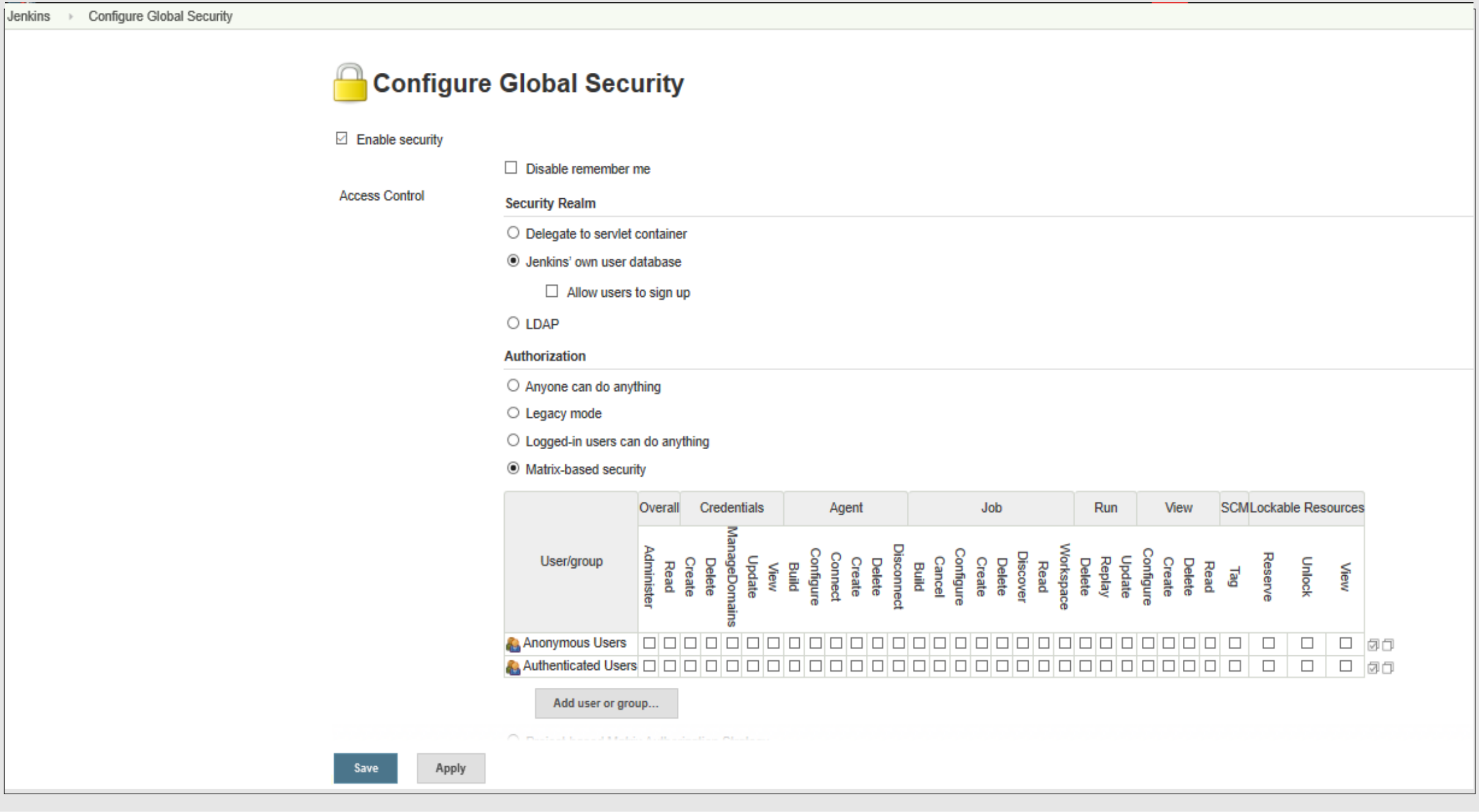

Jenkins
Jenkins
Jenkins' own user database
Back to Dashboard
Manage Jenkins
Create User
Create User
Username:
testuser
Password:
Confirm password:
Full name:
testuser
E-mail address:
testuser@gmail.com
Create User

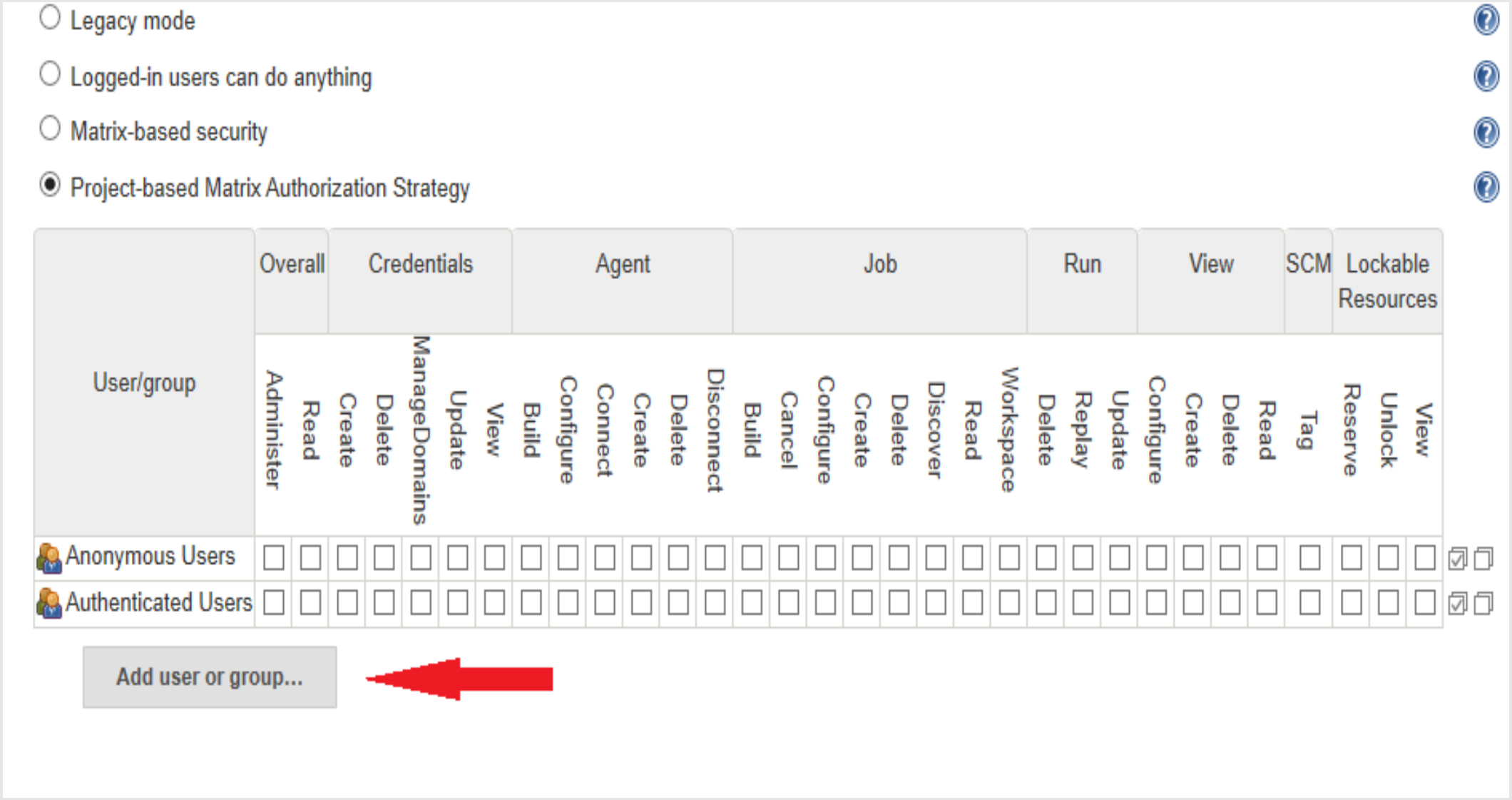

Jenkins
1
search
Jenkins
Jenkins' own user database
Back to Dashboard
Manage Jenkins
Create User
Users
These users can log into Jenkins. This is a sub set of this list, which also contains auto-created users who really just made some commits on so
no direct Jenkins access.
User ID
Name
testuser
testuser
vijaysingh
vijaysingh

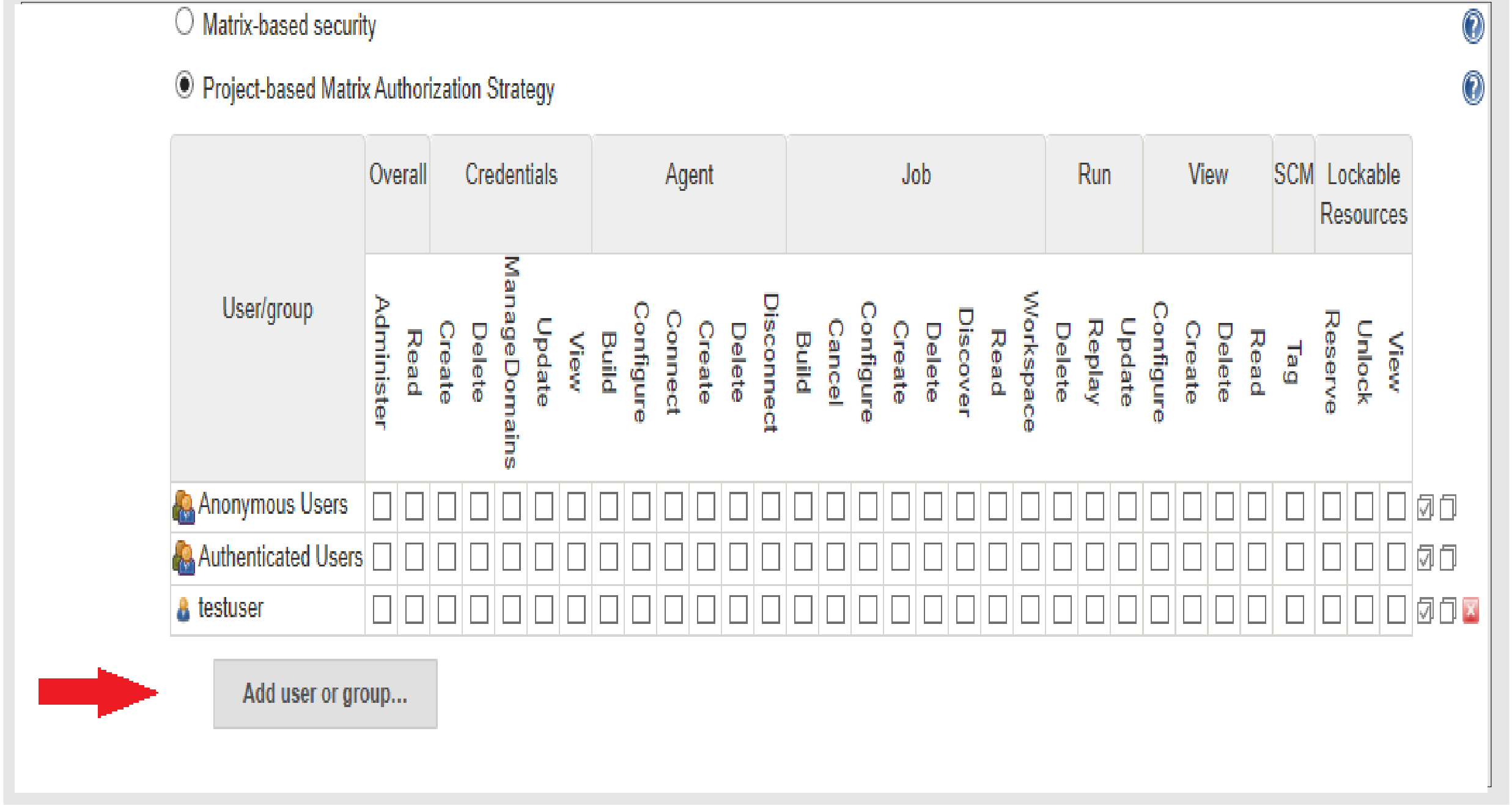
Task 4

//security in Jenkins

1. Manage Jenkins
2. Configure Global Security
3. Authorization
4. Matrix-based Security or Project-based security
5. Add user if you required
6. Assign the previlages
7. Logout and log-in again with different user and explore.


Jenkins
Configure Global Security
Configure Global Security
Enable security
Disable remember me
Access Control
Security Realm
Delegate to servlet container
Jenkins' own user database
Allow users to sign up
LDAP
Authorization
Anyone can do anything
Legacy mode
Logged-in users can do anything
Matrix-based security
User/group
Overall
Credentials
Agent
Job
Run
View
SCM
Lockable Resources
Administer
Read
Create
Delete
ManageDomains
Update
View
Build
Configure
Connect
Create
Delete
Disconnect
Build
Cancel
Configure
Create
Delete
Discover
Read
Workspace
Delete
Replay
Update
Configure
Create
Delete
Read
Tag
Reserve
Unlock
View
Anonymous Users
Authenticated Users
Add user or group...
Save
Apply


Legacy mode
Logged-in users can do anything
Matrix-based security
Project-based Matrix Authorization Strategy
User/group
Overall
Credentials
Agent
Job
Run
View
SCM
Lockable Resources
Administer
Read
Create
Delete
ManageDomains
Update
View
Build
Configure
Connect
Create
Delete
Disconnect
Build
Cancel
Configure
Create
Delete
Discover
Read
Workspace
Delete
Replay
Update
Configure
Create
Delete
Read
Tag
Reserve
Unlock
View
Anonymous Users
Authenticated Users
Add user or group...

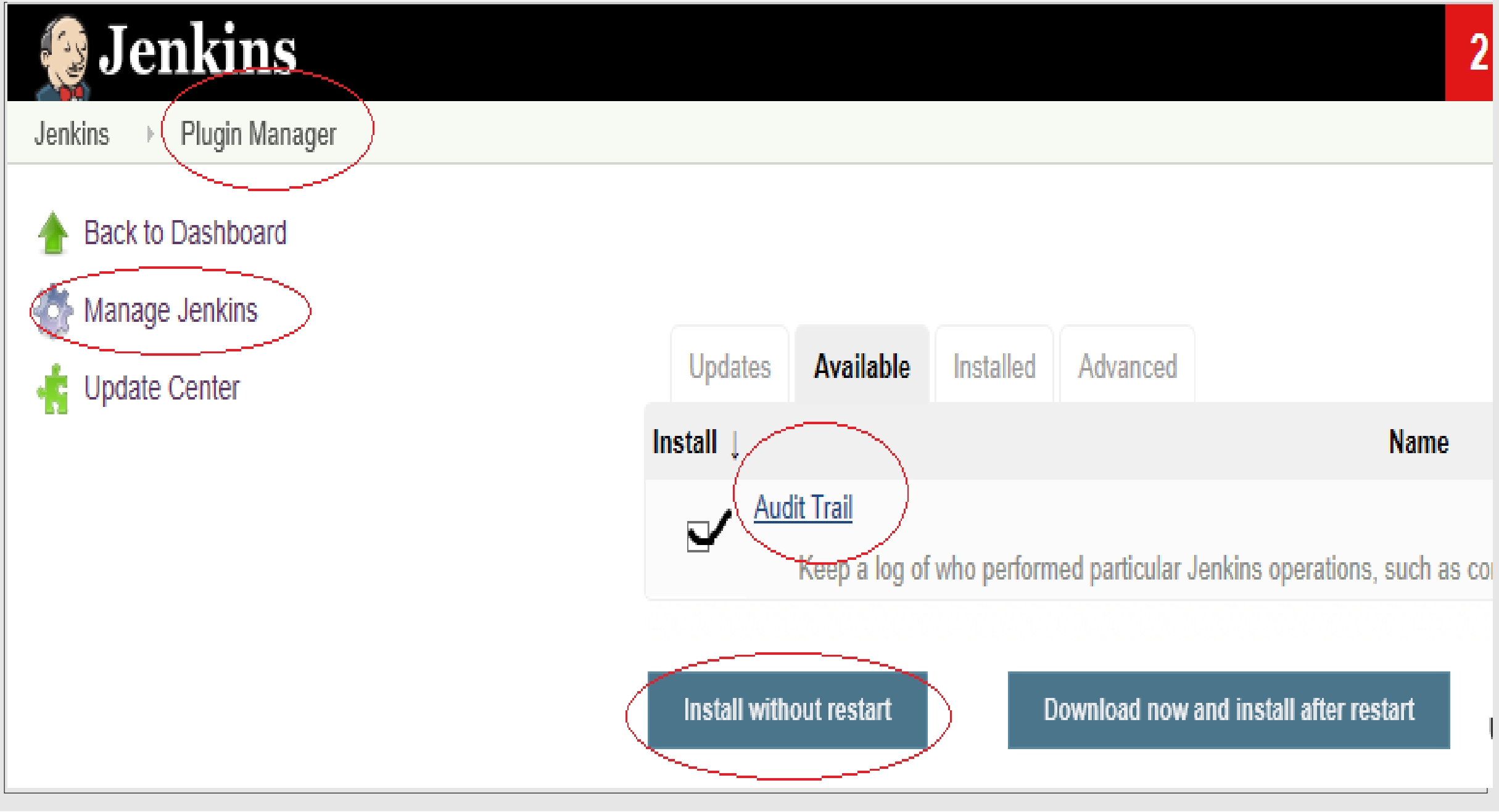

Matrix-based security
Project-based Matrix Authorization Strategy
User/group
Overall
Credentials
Agent
Job
Run
View
SCM
Lockable Resources
Administer
Read
Create
Delete
ManageDomains
Update
View
Build
Configure
Connect
Create
Delete
Disconnect
Build
Cancel
Configure
Create
Delete
Discover
Read
Workspace
Delete
Replay
Update
Configure
Create
Delete
Read
Tag
Reserve
Unlock
View
Anonymous Users
Authenticated Users
testuser
Add user or group...

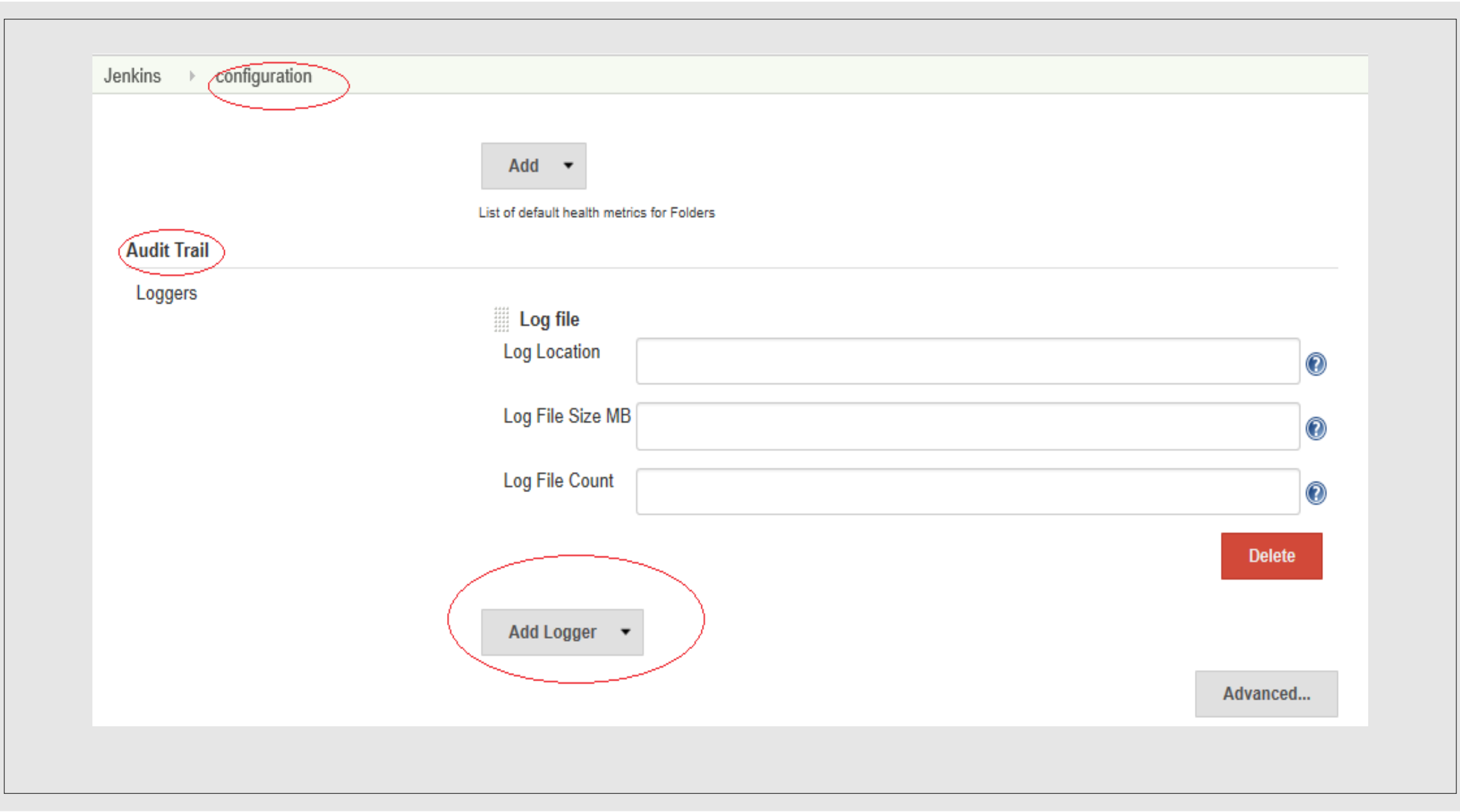
Task 5

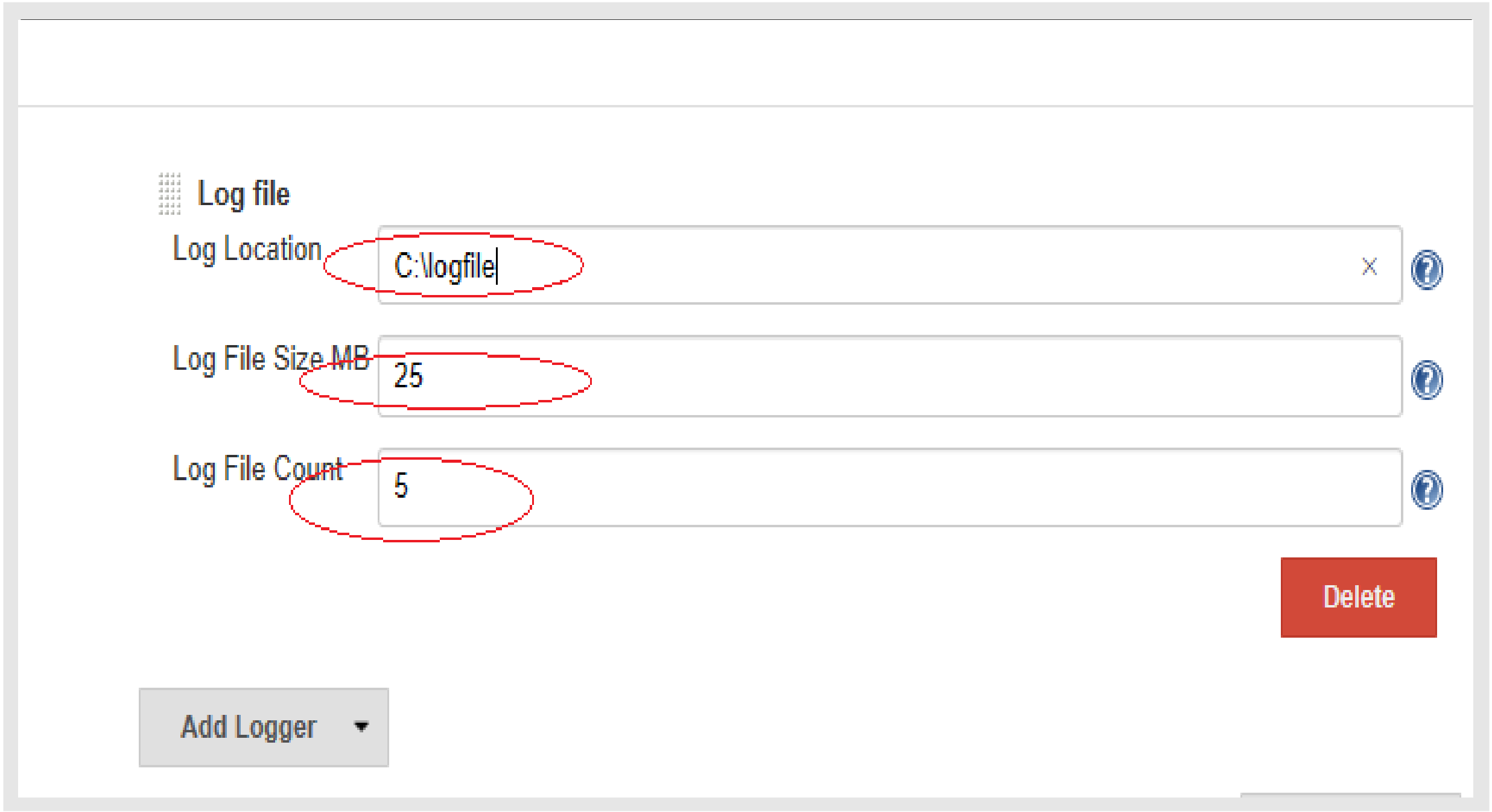
//for maintaining the log file

1. Manage Jenkins
2. Manage Plugins
3. Search Audit Trail plugin
4. Choose Audit Trail and install without restart
5. once it is installed go to Manage Jenkins
6. Configure Systems
7. Global properties->Audit trail
8. Add Logger->log file
9. Add Log Location: /tmp/jenkinsAudir.log
10. Log file sizeMB: 25
11. Log file count: 5
12. Save

// After executing any job you can check the log file /tmp/jenkinsAudit.log


Jenkins
2
Jenkins
Plugin Manager
Back to Dashboard
Manage Jenkins
Update Center
Updates
Available
Installed
Advanced
Install
Name
Audit Trail
Keep a log of who performed particular Jenkins operations, such as co
Install without restart
Download now and install after restart

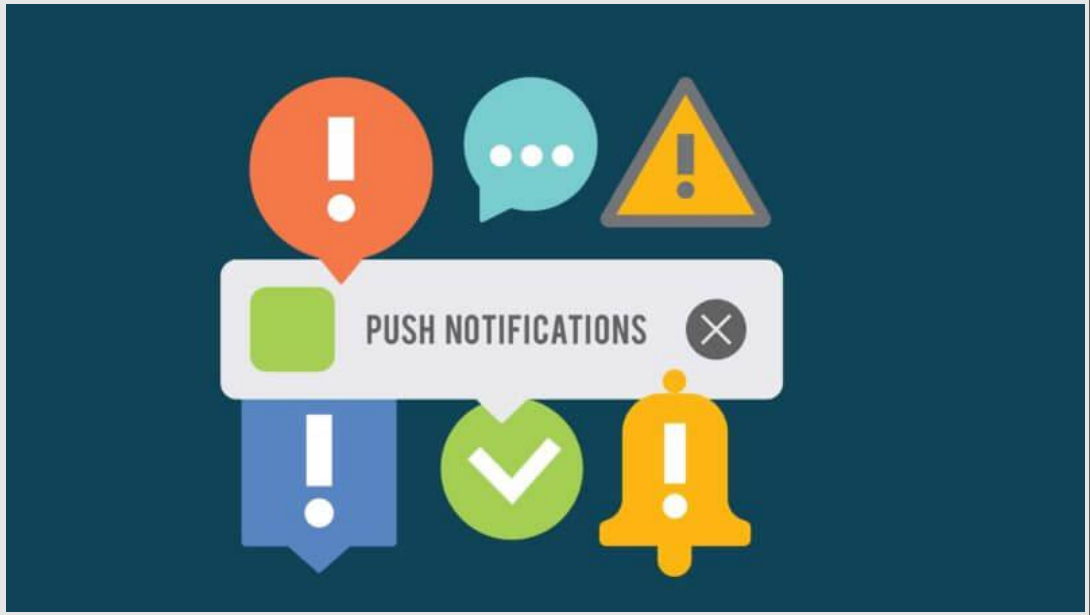

Jenkins
configuration
Add
List of default health metrics for Folders
Audit Trail
Loggers
Log file
Log Location
Log File Size MB
Log File Count
Delete
Add Logger
Advanced...


Log file
Log Location
C:\logfile
Log File Size MB
25
Log File Count
5
Delete
Add Logger

Task 6

//Notifications

1. Manage Jenkins
2. Configure Systems
3. Extended E-mail Notifications
4. SMTP server: smtp.gmail.com
5. Advanced
6. Check Use SMTP Authentication
7. Username : abc@gmail.com
8. Password : *******
9. Use SSI check
10. SMTP port: 465
11. Default recipient :abc@gmail.com
12. Save



Extended E-mail Notification

SMTP server: smtp.gmail.com

Default user E-mail suffix:

Use SMTP Authentication

Advanced Email Properties:

Use SSL: ☑

SMTP port: 465

Charset: UTF-8

Additional accounts: Add

Default Content Type: Plain Text (text/plain)

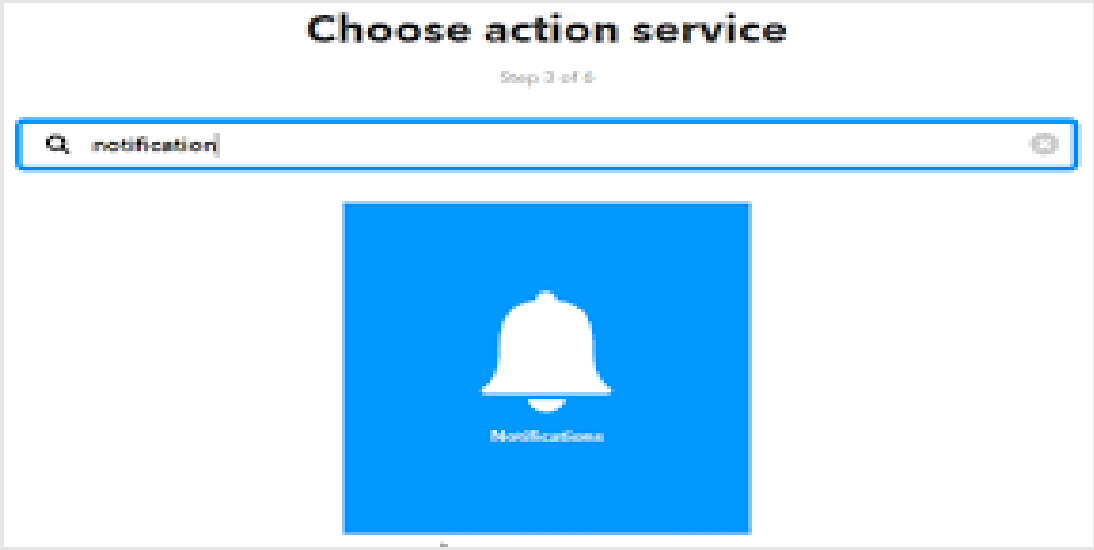
Use List-ID E-mail Header

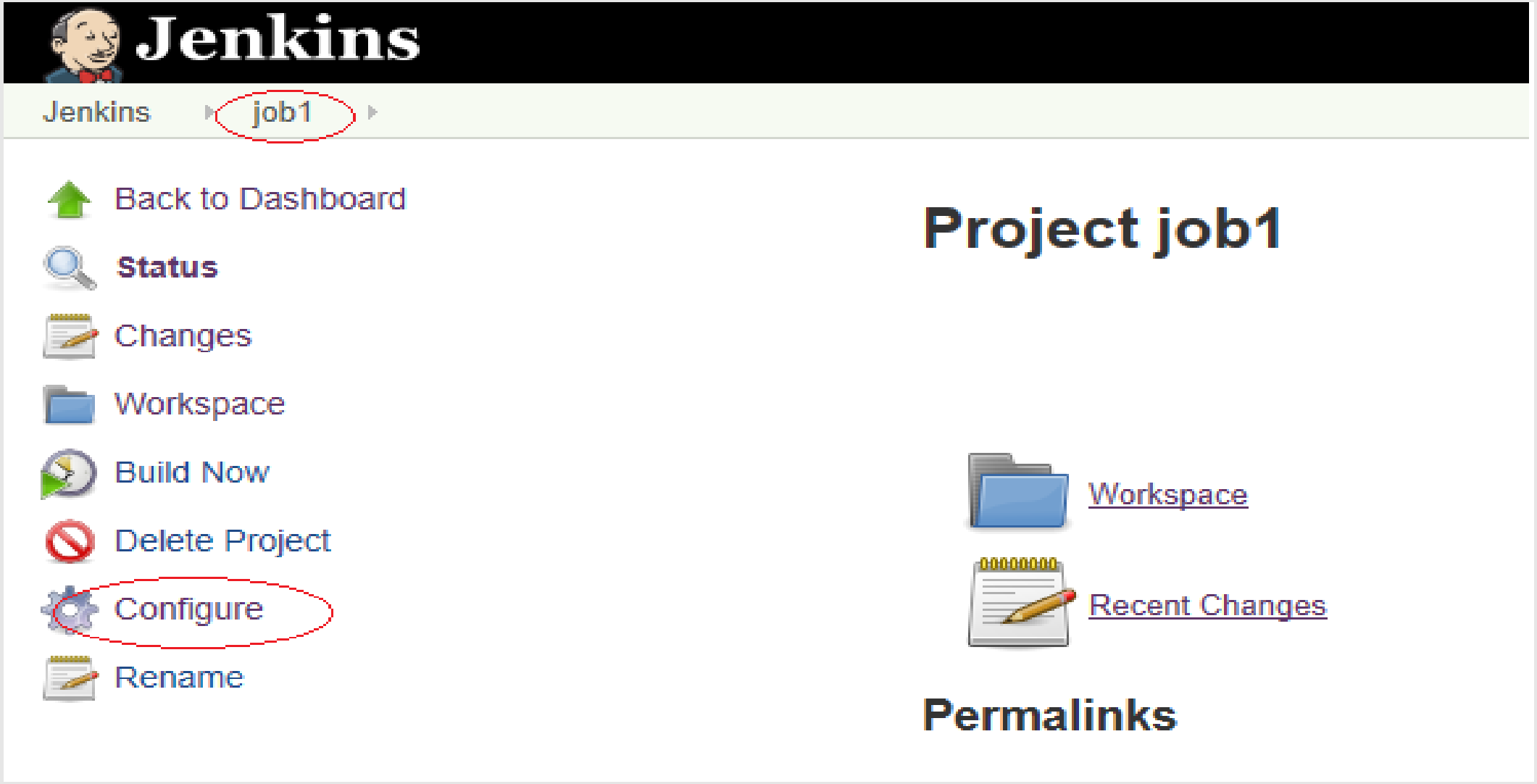
Add 'Precedence: bulk' E-mail Header

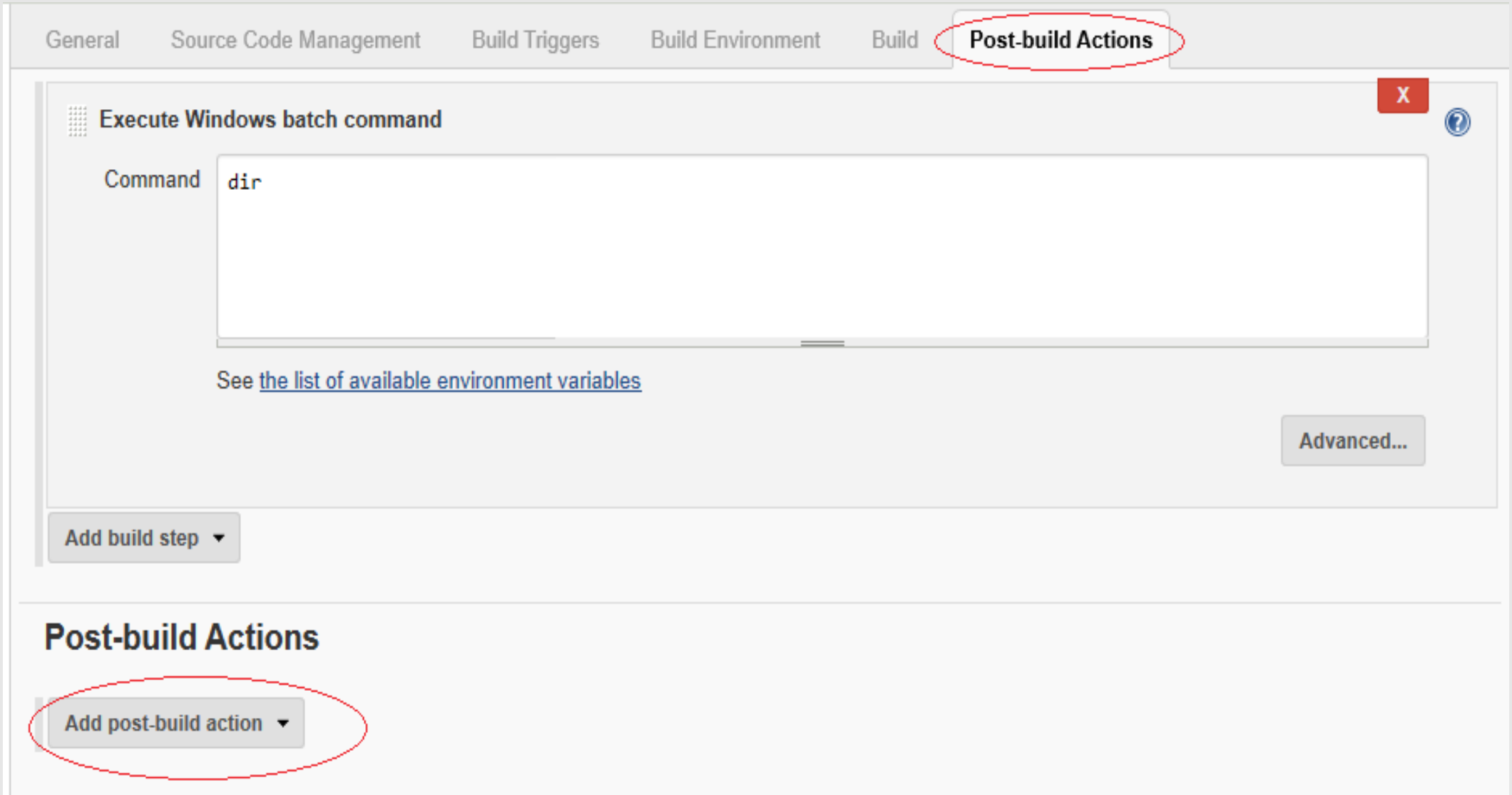
Default Recipients: abc@gmail.com

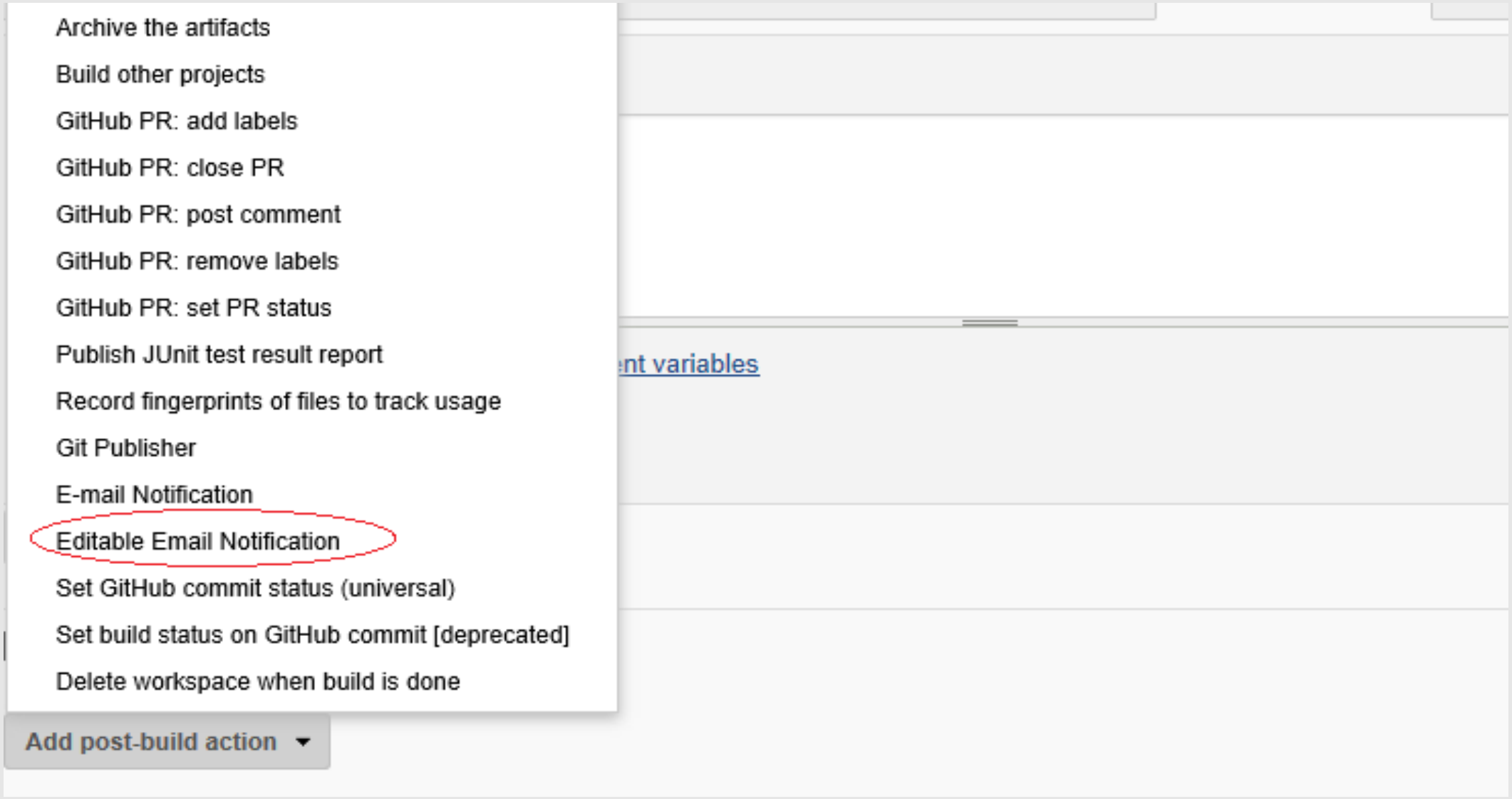
Save | Apply

- ########## Task 7 ##########
- // Notification for a Job
- 1. Select a Job
- 2. Configure
- 3. Post-build Actions
- 4. Editable Email Notification
- 5. Project recipient list: abc@gmail.com
- 6. Advanced
- 7. Trigger
- 8. Add a trigger(recipient list)
- 9. Save




Jenkins
Jenkins
job1
Back to Dashboard
Status
Changes
Workspace
Build Now
Delete Project
Configure
Rename
Project job1
Workspace
Recent Changes
Permalinks


General
Source Code Management
Build Triggers
Build Environment
Build
Post-build Actions
Execute Windows batch command
X
Command
dir
See the list of available environment variables
Advanced...
Add build step
Post-build Actions
Add post-build action


Archive the artifacts
Build other projects
GitHub PR: add labels
GitHub PR: close PR
GitHub PR: post comment
GitHub PR: remove labels
GitHub PR: set PR status
Publish JUnit test result report
Record fingerprints of files to track usage
Git Publisher
E-mail Notification
Editable Email Notification
Set GitHub commit status (universal)
Set build status on GitHub commit [deprecated]
Delete workspace when build is done
nt variables
Add post-build action

//Scheduling the Jobs(Three ways)

a. Timer
b. Poll SCM
c. Pipeline

Task 8

//Timer

1. Choose any Job
2. Configure
3. Build Trigger
4. Build periodically

// little bit Learn about, what is cron job?

5. Type in schedule: */2 * * * *

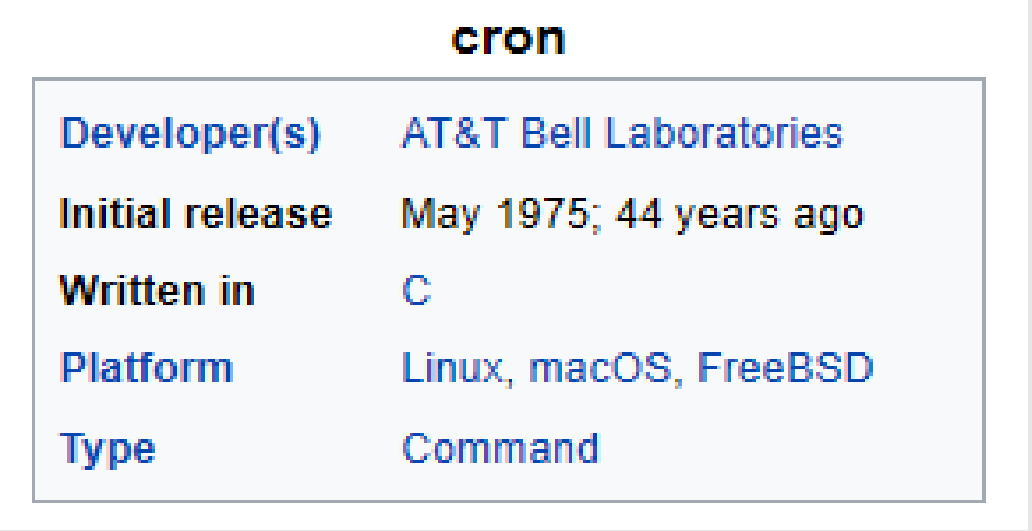
// every two minutes it will execute the job

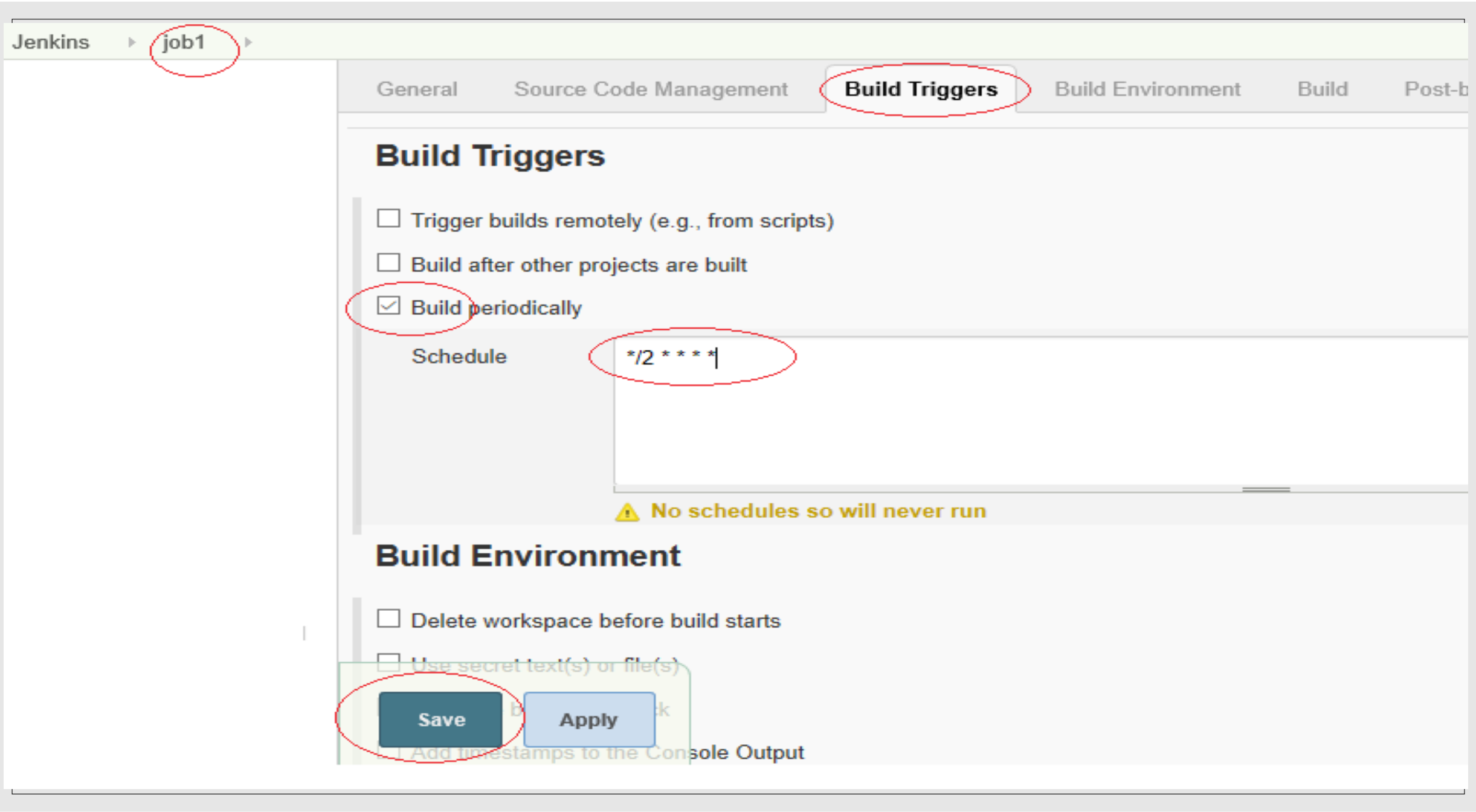
6. Save
7. Check the build history, in output console check that the job is started by timer, not by the user.

cron

The software utility cron is a time-based job scheduler in Unix-like computer operating systems. Users that set up and maintain software environments use cron to schedule jobs (commands or shell scripts) to run periodically at fixed times, dates, or intervals. It typically automates system maintenance or administration—though its general-purpose nature makes it useful for things like downloading files from the Internet and downloading email at regular intervals.

https://en.wikipedia.org/wiki/Cron




Jenkins
job1
General
Source Code Management
Build Triggers
Build Environment
Build
Post-b
Build Triggers
Trigger builds remotely (e.g., from scripts)
Build after other projects are built
Build periodically
Schedule
*/2 * * * *
No schedules so will never run
Build Environment
Delete workspace before build starts
Save
Apply
sole Output

Task 9

//Poll SCM, pull the code from github and compile every two minutes or any changes occur then only it will compile.

1. Create new Job (job_pollSCM_Demo)
2. Freestyle project
3. OK
4. configure
5. Source code management
6. git
7. Repositories
8. Repository URL: copy and paste Git Repository URL

// any commits happens, then only it will build

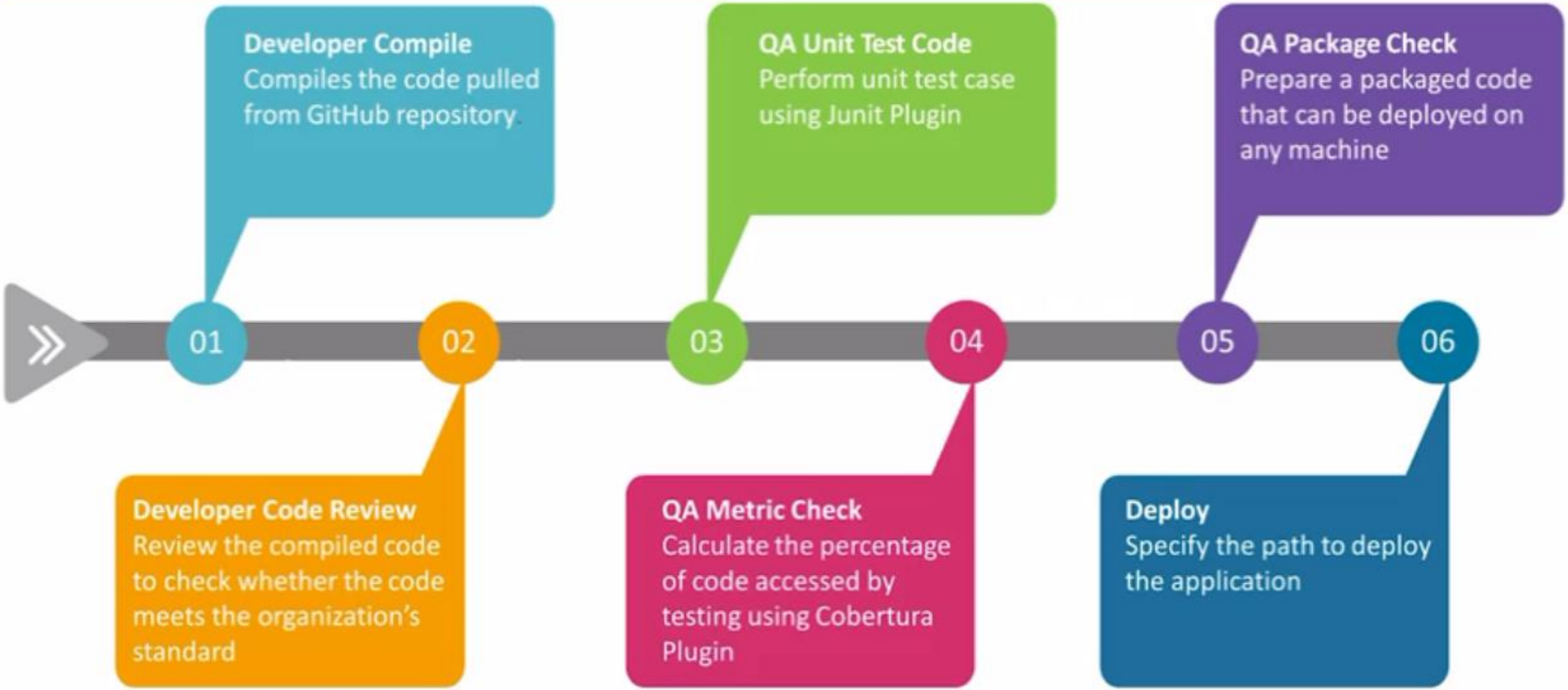
9. Save
10.Configure
11. Build Trigger
12. check Poll SCM
13. Schedule: */2****
14. Save

//perform some commit on Github and check the reflection on jenkins

//check the workspace at /var/lib/jenkins/workspace/

cd /var/lib/jenkins/workspace/

Delivery Pipeline



Task 10

Delivery Pipeline for the project

##

01(Compile)->02(Code Review)->03(Unit Test Code)->04(Metric Check)->05 (Package Check)->06(Deploy)

##

01=>Compile: Compiles the code pulled from github repository.

02=>Code Review: Review the compiled code to check whether the code meets the organization's standard

03=> Unit Test Code: Perform unit test case using

Junit Plugin

04=> Metric Check: Calculate the percentage of code

accessed by testing using Cobertura Plugin

05=>Package Check: Prepare a packaged code that

can be deployed on any machine.

06=>Deploy: Specify the path to deploy the application.