

## **Dvs Technologies Aws & Devops**

**Compiled and Scrutinized by  
Mr. Shaan Shaik  
(Senior DevOps Lead)**

### **Words To The Students**

Though we have taken utmost efforts to present you this book error free, but still it may contain some errors or mistakes. Students are encouraged to bring, if there are any mistakes or errors in this document to our notice. So that it may be rectified in the next edition of this document.

**“Suppressing your doubts is Hindering your growth”.**

We urge you to work hard and make use of the facilities we are providing to you, because there is no substitute for hard work. We wish you all the best for your future.

**“The grass isn’t greener on the other side; the grass is greener where you water it.”**

You and your suggestions are valuable to us; Help us to serve you better. In case of any suggestions, grievance, or complaints, please feel free to write us your suggestions, grievance and feedback on the following

**Dvs.training@gmail.com**

## **Jenkins Class Notes**

### **Introduction to Jenkins:**

#### **What is Jenkins:**

**Jenkins is a leading open source automation server, Jenkins provides hundreds of plugins to support building, deploying, and automating any project.**

**Jenkins is an application that monitors executions of repeated jobs, such as building a software project or jobs run by cron.**

**Traditionally, development make s/w available in a repository and then operations builds and deploys the software to one or more environments.**

**A QA(Quality Assurance) team may then execute load and performance test against that build and release it for production use**

**Jenkins can automate large portions of that repeatable process.**

**What's so great about it:**

**Its a open source!**

**it works for most people**

**Its easy to setup and configure**

**Its hightly flexible with a ton of plugin support.**

**Works in just about any environment.**

**It doesn't use a lot of resources**

**What's Covered in this course:**

**CI & CD**

**Installing Jenkins and Prerequisites**

**Configuring and securing jenkins**

**BasicProjects - Configuring and concepts**

**Folders and views**

**Working with plugins**

**and etc ...**

## **Continuous Integration and Continuous Delivery:**

### **CI:**

**It's a s/w development practice where contributors are integrating their work frequency.**  
**This results in**  
**multiple daily integrations**  
**to a mainline. Automated testing(post-commit promotion) is commonly used.**

### **Basic Workflow:**

- 1. Checkout from source code management (like git)**
- 2. Branch and make local changes**
- 3. Add or change tests as necessary**
- 4. Trigger automated build locally**
- 5. If successful, consider committing**
- 6. Update with latest from mainline**
- 7. Push changes, build and test on integration**

### **Continuous Delivery:**

**A s/w development discipline where software is built so that it can be released to production at anytime**

### **Keys and Assumptions:**

**S/w is always deployable throughout s/w development life cycle (SDLC)**  
**Not breaking the build is prioritized over adding features.**  
**Feedback is fast and production readiness is known.**  
**Push-button deployments are possible of any version of the s/w**  
**Close/Collaborative working relationship**

### **Best practices:**

- 1. Maintain a single source repository**
- 2. Have a common mainline branch (usually master)**
- 3. Automate the build**
- 4. Minimize potential for user error, automate everything possible**
- 5. Make the build self-testing - self-testing code**
- 6. Everyone commits frequently (at least daily preferably)**
- 7. Communication is a key!**
- 8. Frequent mergers will help**

9. The working branch should be updated as frequently as possible to help avoid very large diffs while merging.
10. Build every commit
11. Prioritize fixing broken builds
12. Keep your builds fast !!!
13. Testing environment should be as close to prod as possible
14. Make it easy for anyone to get the latest
15. Keep it open, everyone should see what's happening
16. Automate the deployment

#### Must Do:

- \* Backup Jenkins(at least Jenkins home for job security)
- \* use file fingerprinting to manage dependencies
- \* Build from source control whenever possible (will generate clean builds)
- \* Integrate Jenkins with an issue management or help desk system
- \* Take advantages of automated testing(with java builds,) generate and look at the reports
- \* Layout your jenkins install on the disk with the most storage (both jenkins jobs and build process will take up space)
  
- \* Before deleting a job have an archive copy. better: never delete, move to an archive group or folder and disable the jobs
- \* Resist the temptation to have one build job for multiple environments (dev,test,pord) - Consider creating one job to specialize in each environment to retain flexibility to make changes
- \* Email the results to all developers and operations staff for every job, particularly if jenkins is not integrated into and issue management system
- \* User Jenkins for common maintenance or clean up tasks that are run regularly.
- \* Tag merge or baseline your code in source control after a successful build
- \* Keep your jenkins up to date- atleast be on the latest LTS version
- \* keep your plugins up to date- review the bug reports and see what they address
- \*DON'T BUILD ON MASTER except on very small deployments. If you have more than a dozen jobs and they are used by more than two or three people, set up build slaves to do the work.

All the above are the best practices which helps you to be in safe side.

#### Build Pipeline:

It is a process by which the software build is broken down in sections:

1. Unit test
2. Acceptance test
3. Packaging
4. Reporting
5. Deployment

## **6. Notification**

these can then be executed in series, or in parallel, and depending on the Success or failure of any phase,  
it can automatically be moved to the next phase.

Where is "Devops" in that, through automation, the tools and skills needed are very "cross domain"

### **Best Practices:**

jenkins and devops movement have caused companies to think of common tasks differently:

- \* **Build Management**
- \* **Release Management**
- \* **Deployment Automation**
- \* **Test Orchestration**

### **Difference b/w CI & CD (continuous delivery):**

**Continuous delivery is the ability to release at any time.**

**Continuous integration is just the practice of integrating code continuously.**

### **Continuous Delivery Versus Continuous Deployment:**

**Continuous Delivery means the code can be delivered at anytime**

**Continuous deployment is that code is released continuously as part of an automated pipeline**

### **Questions & Answers:**

#### **1) What's the difference between Continuous Delivery and Continuous Deployment?**

**Continuous Delivery means the code CAN be released at any time, while Continuous Deployment means it is released continuously.**

#### **2) Which answer best describes Continuous Integration?**

**A software development practice where contributors are integrating their work very frequently.**

#### **3) Which answer best describes Continuous Deployment?**

**A software development discipline where software is released continuously as part of an automated pipeline.**

**4) Which is NOT a Continuous Integration best practice?**

**Do everything manually with great care.**

**5) Which answer best describes Continuous Delivery?**

**A software development discipline where software is built so that it can be released to production at any time.**

## Jenkins Installation

ries

Prerequisites:

1. Make sure that java is installed

```
yum install java-1.8.0-openjdk-devel -y  
java -version  
update-alternatives --config java  
update-alternatives --config javac  
java -version
```

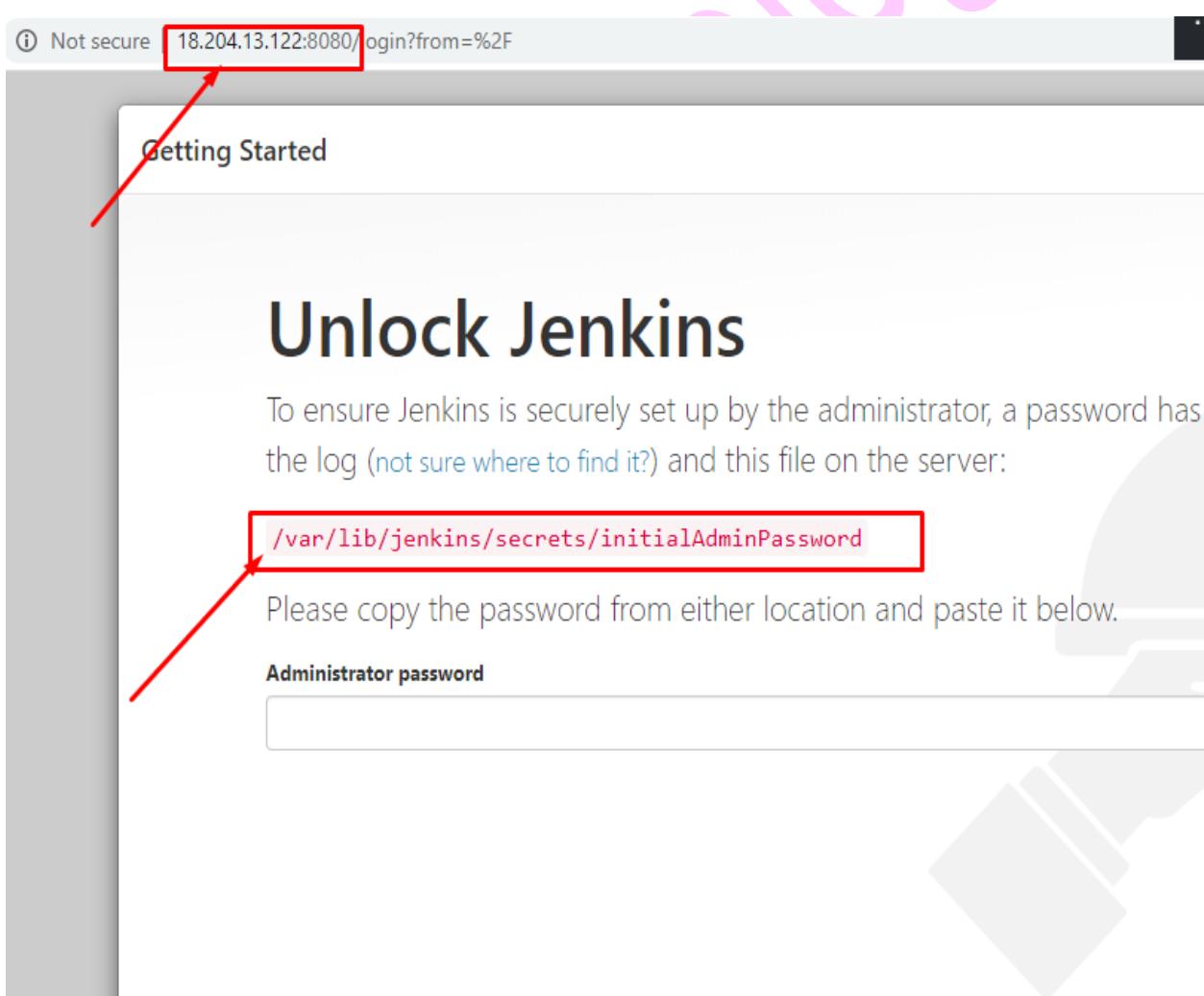
Jenkins Installation:

Note: Here we are going to install jenkins @specific version i.e "2.19.4-1.1". At the end we are going to disable the repo to ma

1. wget -O /etc/yum.repos.d/jenkins.repo <https://pkg.jenkins.io/redhat-stable/jenkins.repo>
2. rpm --import <http://pkg.jenkins.io/redhat-stable/jenkins.io.key>
3. yum list --show-duplicates jenkins :> Helps to show the duplicates of the packages.
4. yum install -y jenkins\*
5. yum-config-manager --disable jenkins :> Disables the jenkins repo to avoid auto updates
6. Before restarting the services make sure that port 8080 is free ... using below  
`netstat -nap|grep :8080`
7. service jenkins start
8. chkconfig jenkins on
9. Now open your serverip:8080 it will ask you to do copy the password from "/var/lib/jenkins/secrets/initialAdminPassword". Open cat /var/lib/jenkins/secrets/initialAdminPassword  
75aa5a6089dd4900b0b7c77d01f106ca

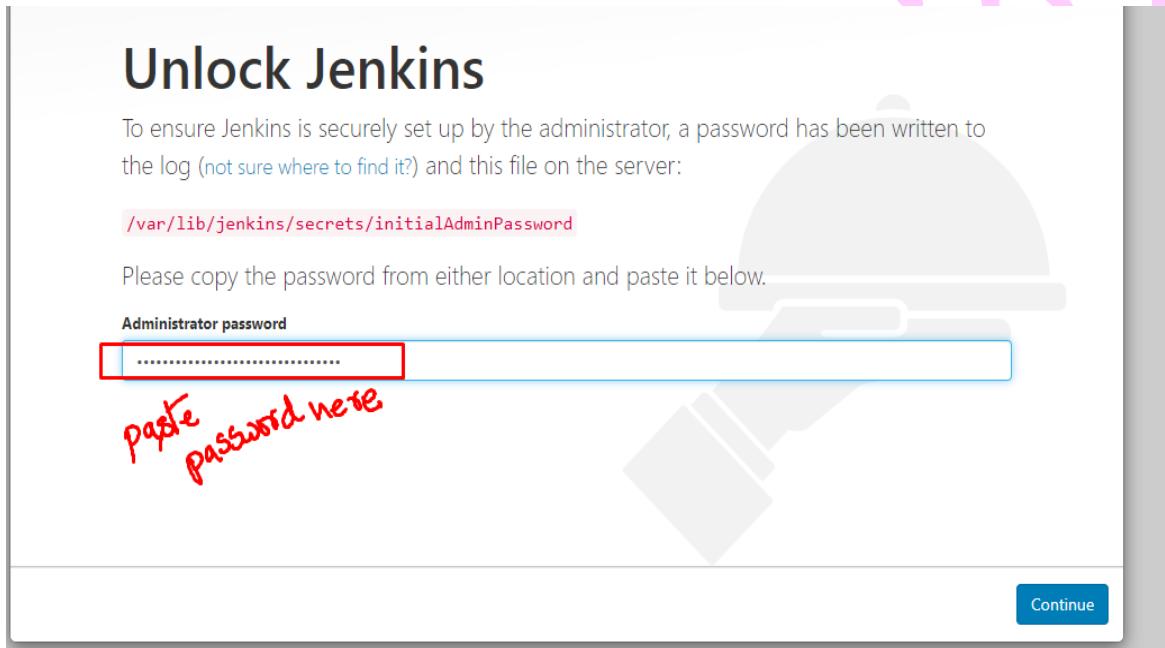
## Verification:

```
[root@jenkins-master ~]# netstat -nap|grep :8080
[root@jenkins-master ~]# systemctl enable jenkins
jenkins.service is not a native service, redirecting to /sbin/chkconfig.
Executing /sbin/chkconfig jenkins on
[root@jenkins-master ~]# systemctl restart jenkins
[root@jenkins-master ~]# netstat -nap|grep :8080
[root@jenkins-master ~]# netstat -nap|grep :8080
tcp6       0      0 :::8080          :::*        LISTEN      3911/java
[root@jenkins-master ~]# ps -ef|grep -i 3911
jenkins   3911  1 99 02:47 ?        00:00:19 /etc/alternatives/java -Dcom.sun.akuma.Daemon=daemonized -Djava.awt.headless=true -DJENKINS_HOME=/var/lib/jenkins -jar /usr/lib/jenkins/jenkins.war --logfile=/var/log/jenkins/jenkins.log --webroot=/var/cache/jenkins/war --daemon --httpPort=8080 --debug=5 --handlerCountMax=100 --handlerCountMaxIdle=20
root      3968  3595  0 02:47 pts/0    00:00:00 grep --color=auto -i 3911
[root@jenkins-master ~]#
```



## In Server:

```
[root@jenkins-master ~]# cat /var/lib/jenkins/secrets/initialAdminPassword  
27dfc3f39d5f49c096331a8605919be7  
[root@jenkins-master ~]#
```



The image shows a screenshot of a web-based Jenkins setup interface. At the top, it says "Unlock Jenkins". Below that, a message states: "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:". A red box highlights the path `/var/lib/jenkins/secrets/initialAdminPassword`. Below this, a message says "Please copy the password from either location and paste it below." A red box surrounds a password input field with the placeholder "Administrator password" and the handwritten note "paste password here". In the bottom right corner of the input field, there is a blue "Continue" button.

# 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

### Create First Admin User

Username:	admin
Password:	.....
Confirm password:	.....
Full name:	admin
E-mail address:	admin@gmail.com

fill all details

Jenkins 2.235.3

Skip and continue as admin

Save and Continue

## Getting Started

# Instance Configuration

Jenkins URL:

<http://18.204.13.122:8080/>

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 2.235.3

[Not now](#)

[Save and Finish](#)



# Jenkins is ready!

You have skipped the configuration of the Jenkins URL.

To configure the Jenkins URL, go to "Manage Jenkins" page.

Your Jenkins setup is complete.

[Start using Jenkins](#)

The screenshot shows the Jenkins dashboard. At the top, there is a navigation bar with the Jenkins logo, a search bar, and a user menu showing 'admin' with a red notification badge. Below the navigation bar, the main content area has a large 'Welcome to Jenkins!' heading. It includes two calls-to-action: 'Create an agent or configure a cloud' and 'Create a job'. On the left side, there is a sidebar with links like 'New Item', 'People', 'Build History', 'Manage Jenkins', 'My Views', 'Lockable Resources', and 'New View'. At the bottom, there is a 'Build Queue' section indicating 'No builds in the queue.'

## User Management

### Creating User:

The screenshot shows the Jenkins dashboard. On the left sidebar, there is a list of links: New Item, People, Build History, Manage Jenkins, My Views, Lockable Resources, and New View. The 'Manage Jenkins' link is highlighted with a red box and a red arrow pointing to it from the top-left.

### Welcome to Jenkins!

Create an agent or configure a cloud to set up distributed builds. [Learn more](#).

Create a job to start building your software project.

#### Build Queue

No builds in the queue.

#### Build Executor Status

1 Idle  
2 Idle



#### Configure Global Security

Secure Jenkins: define who is allowed to access/use the system.



#### Manage Credentials

Configure credentials



#### Configure Credentials

Configure the credential types



#### Manage Users

Create/delete/modify users that can log in to this Jenkins

### Status Information



#### System Information



#### System Log



#### Load Statistics

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 have no direct Jenkins access.

User ID	Name
 admin	admin

## Create User

Username:	dvsbatch3
Password:	.....
Confirm password:	.....
Full name:	dvsbatch3
E-mail address:	dvsbatch3@gmail.com

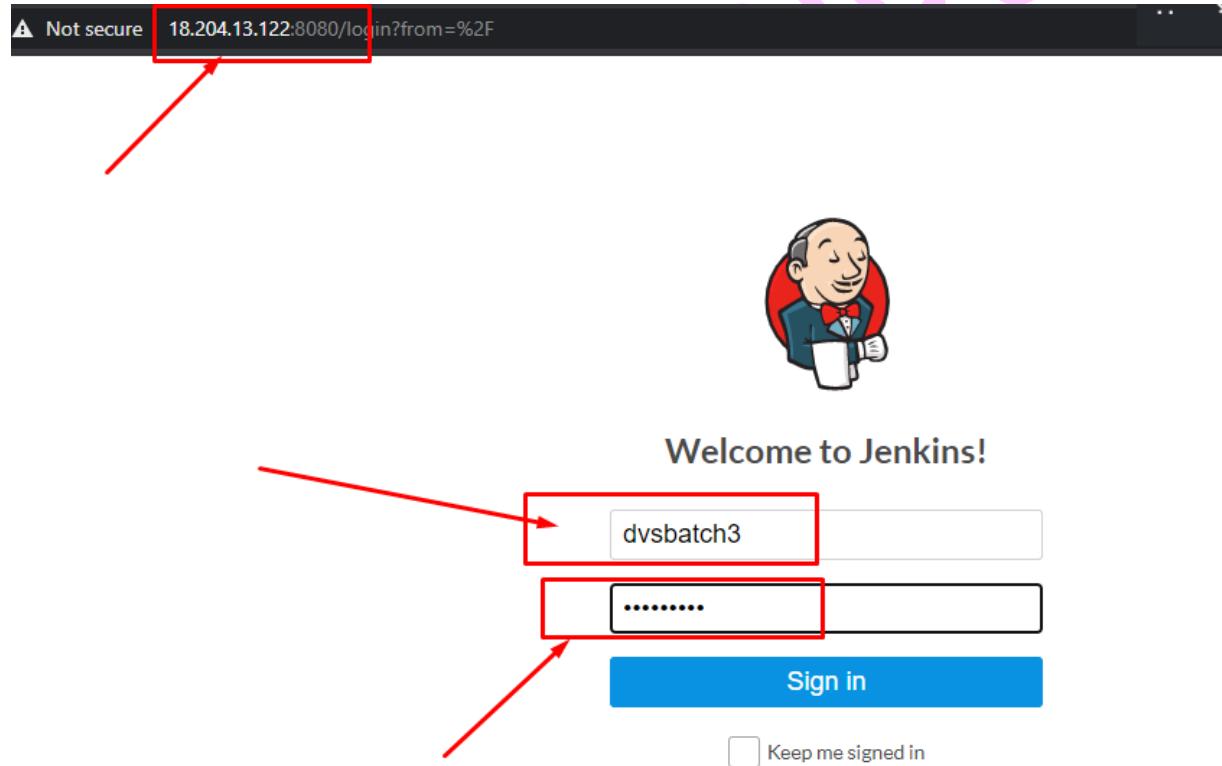
[Create User](#)

## 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 or have no direct Jenkins access.

User ID	Name
 admin	admin
 dvsbatch3	dvsbatch3

### Verifying the Access:



A Not secure 18.204.13.122:8080/login?from=%2F

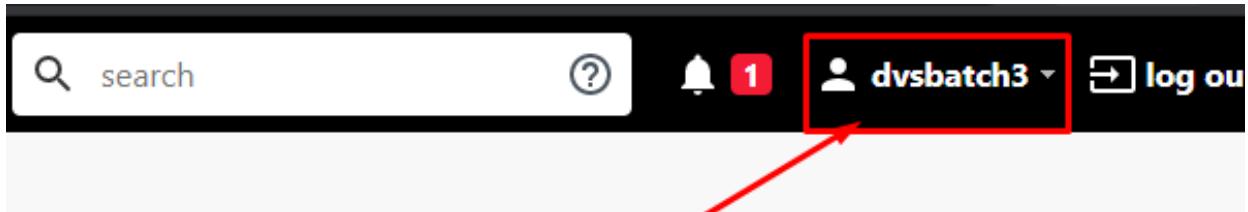
Welcome to Jenkins!

dvsbatch3

.....

Sign in

Keep me signed in



[add descripti](#)

distributed builds. [Learn more.](#)

### Restricting User Access:

A screenshot of the Jenkins dashboard. At the top, there is a navigation bar with the Jenkins logo, a search bar, and a 'Jenkins' link. Below the navigation bar is a sidebar with links: 'New Item', 'People', 'Build History', 'Manage Jenkins' (which is highlighted with a red box and a red arrow), 'My Views', 'Lockable Resources', and 'New View'. The main content area features a large 'Welcome to Jenkins!' heading, a callout for 'Create an agent or configure a cloud' to set up distributed builds, and another callout for 'Create a job' to start building your software project. Below these calls to action are sections for 'Build Queue' (showing 'No builds in the queue.') and 'Build Executor Status' (showing '1 Idle' and '2 Idle').

## Security



### Configure Global Security

Secure Jenkins; define who is allowed to access/use the system.



### Manage Credentials

Configure credentials



### Configure C

Configure the types



### Manage Users

Create/delete/modify users that can log in to this Jenkins

## Status Information



### System Information

Displays various environmental information to assist trouble-shooting.



### System Log

System log captures output from java.util.logging output related to Jenkins.



### Load Statist

Check your res see if you need your builds.



### About Jenkins

#### Authorization

Strategy

#### Authorization

- Anyone can do anything
- Legacy mode
- Logged-in users can do anything

Matrix-based security

User/group	Overall	Credentials	Agent	Job	Run
	ManageDomains	Update	View	Configure	Disconnect
Anonymous Users	<input type="checkbox"/>				
Authenticated Users	<input type="checkbox"/>				

Add user or group...

Project-based Matrix Authorization Strategy

#### Markup Formatter

18.204.13.122:8080 says

User or group name:

**OK** **Cancel**

	Overall	Credentials	Agent	Job	Run	View	SCM Lockable Resources
User/group	<input checked="" type="checkbox"/> Read	<input checked="" type="checkbox"/> Discover	<input checked="" type="checkbox"/> Delete	<input checked="" type="checkbox"/> Create	<input checked="" type="checkbox"/> Configure	<input checked="" type="checkbox"/> Workspace	<input checked="" type="checkbox"/> View
Administrator	<input checked="" type="checkbox"/> Update	<input checked="" type="checkbox"/> Delete	<input checked="" type="checkbox"/> Cancel	<input checked="" type="checkbox"/> Connect	<input checked="" type="checkbox"/> Build	<input checked="" type="checkbox"/> Read	<input checked="" type="checkbox"/> Unlock
	<input checked="" type="checkbox"/> ManageDomains	<input checked="" type="checkbox"/> Create	<input checked="" type="checkbox"/> Configure	<input checked="" type="checkbox"/> Disconnect	<input checked="" type="checkbox"/> Delete	<input checked="" type="checkbox"/> Tag	<input checked="" type="checkbox"/> Reserve
	<input checked="" type="checkbox"/> Delete	<input checked="" type="checkbox"/> Read	<input checked="" type="checkbox"/> Create	<input checked="" type="checkbox"/> Build	<input checked="" type="checkbox"/> Discover	<input checked="" type="checkbox"/> Delete	<input checked="" type="checkbox"/> Create
	<input checked="" type="checkbox"/> Create	<input checked="" type="checkbox"/> Administer	<input checked="" type="checkbox"/> View	<input checked="" type="checkbox"/> Run	<input checked="" type="checkbox"/> View	<input checked="" type="checkbox"/> Update	<input checked="" type="checkbox"/> Configure

**Add user or group...**

Project-based Matrix Authorization Strategy

Plain text

Treats all input as plain text. HTML unsafe characters like < and & are escaped to their respective character entities.

Points  Fixed :   Random  Disable

**Agent protocols**

Matrix-based security

	Overall	Credentials	Agent	Job	Run	View	SCM Lockable Resources
User/group	<input checked="" type="checkbox"/> Read	<input checked="" type="checkbox"/> Discover	<input checked="" type="checkbox"/> Delete	<input checked="" type="checkbox"/> Create	<input checked="" type="checkbox"/> Configure	<input checked="" type="checkbox"/> Workspace	<input checked="" type="checkbox"/> View
Administrator	<input checked="" type="checkbox"/> Update	<input checked="" type="checkbox"/> Delete	<input checked="" type="checkbox"/> Cancel	<input checked="" type="checkbox"/> Connect	<input checked="" type="checkbox"/> Build	<input checked="" type="checkbox"/> Read	<input checked="" type="checkbox"/> Unlock
	<input checked="" type="checkbox"/> ManageDomains	<input checked="" type="checkbox"/> Create	<input checked="" type="checkbox"/> Configure	<input checked="" type="checkbox"/> Disconnect	<input checked="" type="checkbox"/> Delete	<input checked="" type="checkbox"/> Tag	<input checked="" type="checkbox"/> Reserve
	<input checked="" type="checkbox"/> Delete	<input checked="" type="checkbox"/> Read	<input checked="" type="checkbox"/> Create	<input checked="" type="checkbox"/> Build	<input checked="" type="checkbox"/> Discover	<input checked="" type="checkbox"/> Delete	<input checked="" type="checkbox"/> Create
	<input checked="" type="checkbox"/> Create	<input checked="" type="checkbox"/> Administer	<input checked="" type="checkbox"/> View	<input checked="" type="checkbox"/> Run	<input checked="" type="checkbox"/> View	<input checked="" type="checkbox"/> Update	<input checked="" type="checkbox"/> Configure

**Add user or group...**

Project-based Matrix Authorization Strategy

Plain text

○ Anyone can do anything

○ Legacy mode

○ Logged-in users can do anything

**(●) Matrix-based security**

18.204.13.122:8080 says

User or group name:

**OK** **Cancel**

	Overall	Credentials	Agent	Job
User/group	ManageDomains	Update	View	Delete
	Read	Create	Configure	Create
admin	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

**Add user or group...**

Project-based Matrix Authorization Strategy

Plain text

Treats all input as plain text. HTML unsafe characters like < and & are escaped to their respective character entities.

20%

	Overall	Credentials	Agent	Job
User/group	ManageDomains	Update	View	Delete
	Read	Create	Configure	Create
admin	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
dvsbatch3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

The update permission is necessary to modify

**Add user or group...**

Project-based Matrix Authorization Strategy

**Agent → Master Security**

Enable Agent → Master Access Control  
Rules can be tweaked [here](#)

**SSH Server**

SSHD Port  Fixed :   Random  Disable

**Save** **Apply**

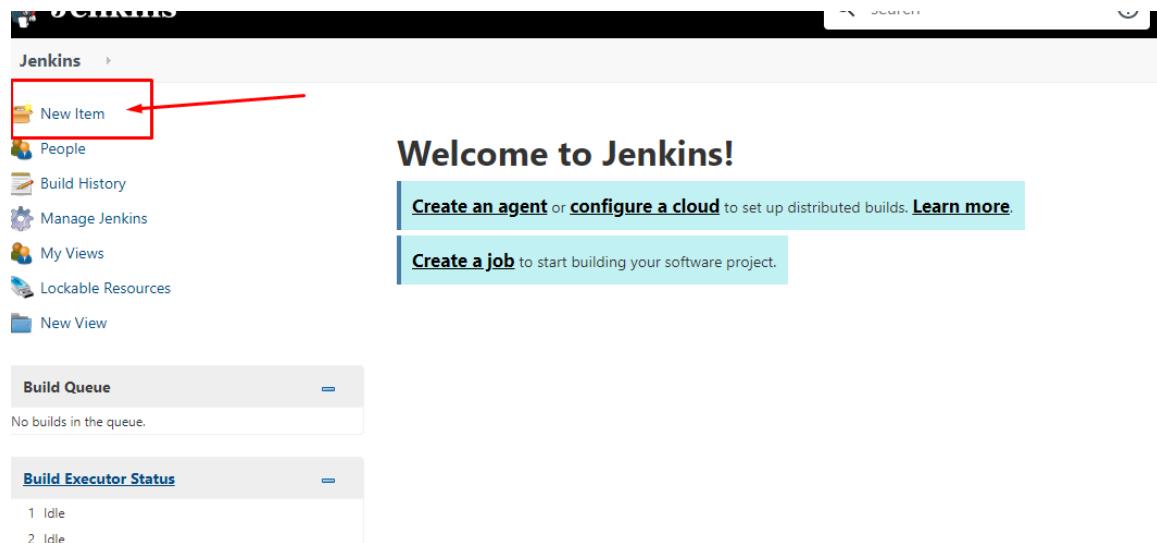
**Jenkins**

Jenkins > dvsbatch3

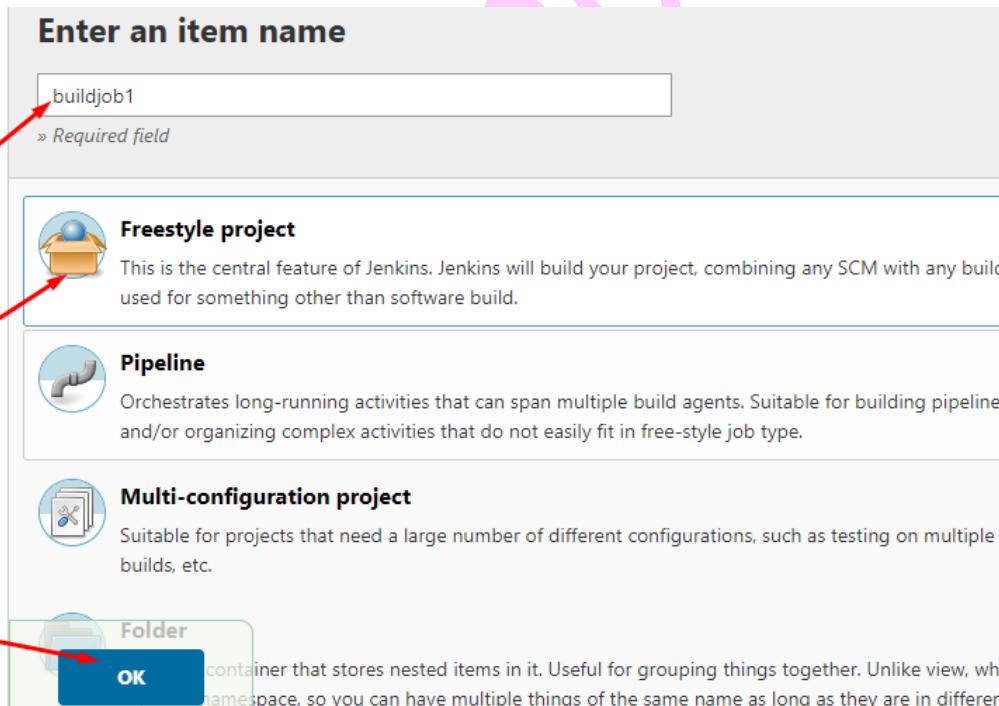
People Status Builds Configure My Views Credentials

**dvsbatch3**  
Jenkins User ID: dvsbatch3

## Creating freestyle jobs



The screenshot shows the Jenkins dashboard. On the left, there's a sidebar with links: 'Jenkins', 'New Item' (which has a red box and arrow pointing to it), 'People', 'Build History', 'Manage Jenkins', 'My Views', 'Lockable Resources', and 'New View'. The main area is titled 'Welcome to Jenkins!' with a message: 'Create an agent or configure a cloud to set up distributed builds. [Learn more](#)'. Below that is another message: 'Create a job to start building your software project.' Underneath these messages are two sections: 'Build Queue' (No builds in the queue) and 'Build Executor Status' (1 Idle, 2 Idle). A pink ribbon graphic is visible on the right side of the dashboard.

The screenshot shows the 'Enter an item name' dialog box. In the input field, 'buildjob1' is typed, with a red arrow pointing to it. Below the input field, a note says '» Required field'. There are three options listed: 'Freestyle project' (selected, shown with a blue border and icon), 'Pipeline' (shown with a blue icon), and 'Multi-configuration project' (shown with a blue icon). At the bottom right of the dialog is a large blue 'OK' button with a red arrow pointing to it. The background of the dialog has decorative pink and purple shapes.

Jenkins > buildjob1 >

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

Description [Plain text] Preview

Discard old builds  GitHub project  This build requires lockable resources  This project is parameterized  Throttle builds  Disable this project  Execute concurrent builds if necessary

Advanced...

**Build**

Execute shell

Command `#!/bin/bash  
echo "Welcome To Dvs Devops!!!!"`

See the list of available environment variables Advanced...

Add build step ▾

**Post-build Actions**

Add post-build action ▾

Save Apply

Jenkins > buildjob1

[Back to Dashboard](#)

[Status](#)

[Changes](#)

[Workspace](#)

[Build Now](#)

[Delete Project](#)

[Configure](#)

[Rename](#)

## Project buildjob1

[Workspace](#)

[Recent Changes](#)

### Permalinks

[Build History](#) [trend](#)

find

[Atom feed for all](#) [Atom feed for failures](#)

Configure

Rename

Recent Changes

Build History

trend

find

#1 - Jul 30, 2020 1:53 AM

[Atom feed for all](#) [Atom feed for failures](#)

A screenshot of a Jenkins build page. At the top, it shows "Jenkins > buildjob1 > #1". Below that, there's a sidebar with links: "Back to Project", "Status", "Changes", "Console Output" (which is highlighted with a red box and has a red arrow pointing to it), "Edit Build Information", and "Delete build '#1'". To the right, there's a large blue circular icon with a white play button, followed by the text "Build #1 (Jul 30, 2020 1:53:29 AM)". Underneath the date, there are two status indicators: "No changes." with a notepad icon and "Started by user admin" with an orange diamond icon. At the bottom left, it says "#1".

## Console Output

```
Started by user admin
Running as SYSTEM
Building in workspace /var/lib/jenkins/workspace/buildjob1
[buildjob1] $ /bin/bash /tmp/jenkins4229061891226135451.sh
Welcome To Dvs Devops!!!!
[finished: SUCCESS]
```

Verfying Build job in the jenkins server:

```
[root@jenkins-master ~]# cd /var/lib/jenkins/
[root@jenkins-master jenkins]# ls
com.cloudbees.hudson.plugins.folder.config.AbstractFolderConfiguration.xml config.xml
hudson.model.UpdateCenter.xml
hudson.plugins.git.GitTool.xml
identity.key.enc
jenkins.install.InstallUtil.lastExecVersion
jenkins.install.UpgradeWizard.state
jenkins.security.apitoken.ApiTokenPropertyConfiguration.xml
jenkins.security.QueueItemAuthenticatorConfiguration.xml
jenkins.security.UpdateSiteWarningsConfiguration.xml
jenkins.telemetry.Correlator.xml
jobs
logs
[root@jenkins-master jenkins]# cd jobs/
[root@jenkins-master jobs]# ls
buildjob1
[root@jenkins-master jobs]# pwd
/var/lib/jenkins/jobs
[root@jenkins-master jobs]#
```

```
[root@jenkins-master jobs]# cd /var/lib/jenkins/workspace/
[root@jenkins-master workspace]# ls -l
total 0
drwxr-xr-x 2 jenkins jenkins 6 Jul 30 01:53 buildjob1
[root@jenkins-master workspace]# cd buildjob1/
[root@jenkins-master buildjob1]# ls -l
```

## Adding a slave node

Do the below on slave1

```
[root@ip-172-31-33-185 ~]#
[root@ip-172-31-33-185 ~]# hostnamectl set-hostname slave1
[root@ip-172-31-33-185 ~]# pass
[root@slave1 ~]# id -a jenkins
id: jenkins: no such user
[root@slave1 ~]# useradd jenkins
[root@slave1 ~]# id -a jenkins
uid=1001(jenkins) gid=1001(jenkins) groups=1001(jenkins)
[root@slave1 ~]# java -version
bash: java: command not found
[root@slave1 ~]# yum install java-1.8.0-openjdk-devel -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
Resolving Dependencies
--> Running transaction check
--> Package java-1.8.0-openjdk-devel.x86_64 1:1.8.0.252.b09-2.amzn2.0.1 will be installed
--> Processing Dependency: java-1.8.0-openjdk(x86-64) = 1:1.8.0.252.b09-2.amzn2.0.1 for package: 1:java-1.8.0-openjdk-d
1-1.8.0.252.b09-2.amzn2.0.1.x86_64
--> Processing Dependency: libjvm.so() (64bit) for package: 1:java-1.8.0-openjdk-devel-1.8.0.252.b09-2.amzn2.0.1.x86_64
```

Install java

Created user

No user

## Do below in master

```
[jenkins@jenkins-master buildjob1]# su -s /bin/bash - jenkins
Last login: Thu Jul 30 02:34:26 UTC 2020 on pts/0
-bash-4.2$ whoami
jenkins
-bash-4.2$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/var/lib/jenkins/.ssh/id_rsa):
Created directory '/var/lib/jenkins/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /var/lib/jenkins/.ssh/id_rsa.
Your public key has been saved in /var/lib/jenkins/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:E2PXvF88+j7rWdaeazoQ8A710n5eQ25ulZ04SdNqIaw jenkins@jenkins-master
The key's randomart image is:
+---[RSA 2048]---+
|          . |
|         . o |
|        + B . |
|       . * =   o |
|      S * . * o |
|     . * * @o |
|    . * X O |
|   E   B X=|
| .oXOO|
+---[SHA256]---+
```

```
-bash-4.2$ hostname
jenkins-master
-bash-4.2$ id -a
uid=996(jenkins) gid=994(jenkins) groups=994(jenkins)
-bash-4.2$ cat ~/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAQABAAQDPRYN3+LqZGkMrj0V [jenkins@slave1 ~]$ whoami
jenkins
[jenkins@slave1 ~]$ pwd
/home/jenkins
[jenkins@slave1 ~]$ mkdir .ssh
[jenkins@slave1 ~]$ chmod 700 .ssh
[jenkins@slave1 ~]$ cd .ssh/
[jenkins@slave1 .ssh]$ touch authorized_keys
[jenkins@slave1 .ssh]$ chmod 600 authorized_keys
[jenkins@slave1 .ssh]$ echo "ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQI9KoLQvHlHaHg7pG6949foEdfxBmiPVKC4Kd jenkins@jenkins-masxyQtJ16NK+mluolmX39+UnXO3YSbInPSB0ZBvhqj6PnUoZTM5G9eFVGibMZZZGgUf2NaRR+R030QCuS534Es8MJ1ltCXfEXnA2Cj1Vt91bfald7Zj3Zpp7eF5BsxwURB12xs1zWzgE6i02NLRxWeCVMSmnoyobeLTwRzh2fBKoEqzot+XwrlDWMmjREx+F6n9cHaka54Fqb/kS9rN5RqbJe5419BftT5gxstTwk3eb9KoLQvHlHaHg7pG6949foEdfxBmiPVKC4Kd jeer" >authorized_keys
[jenkins@slave1 .ssh]$
```

Copy public key of master to slave1

```
jenkins@slave1:~  
-bash-4.2$ ssh slave1  
The authenticity of host 'slave1 (172.31.33.185)' can't be established.  
ECDSA key fingerprint is SHA256:r5FP9EXD+ZeZkDrYSXn9eYUBmq5HzpWR8o4iugtSSPE.  
ECDSA key fingerprint is MD5:db:1e:69:8b:cb:8a:e9:88:51:58:70:14:f0:e3:5c:0f.  
Are you sure you want to continue connecting (yes/no)? yes  
Warning: Permanently added 'slave1,172.31.33.185' (ECDSA) to the list  
of known hosts.  
Last login: Thu Jul 30 02:38:32 2020  
  
      _ |  _ | _ / )  Amazon Linux 2 AMI  
     _ | \_ | _ |  
  
https://aws.amazon.com/amazon-linux-2/  
[jenkins@slave1 ~]s  
[jenkins@slave1 ~]$
```

### **Configure Slave Node in jenkins :**

The screenshot shows the Jenkins dashboard. On the left sidebar, there is a list of links: New Item, People, Build History, Manage Jenkins, My Views, Lockable Resources, and New View. The 'Manage Jenkins' link is highlighted with a red box and has a red arrow pointing towards it from the top-left. The main content area displays a table of build jobs. The columns are labeled S, W, Name (with a downward arrow), Last Success, and Last Failure. One job is listed: buildjob1, which last succeeded 52 minutes ago. Below the table, there are links for Icon, Legend, Atom feed for all, and Atom feed.

S	W	Name ↓	Last Success	Last Failure
		<a href="#">buildjob1</a>	52 min - #1	N/A

Icon: [S](#) [M](#) [L](#)

[Legend](#) [Atom feed for all](#) [Atom feed](#)

## System Configuration



**Configure System**  
Configure global settings and paths.



**Global Tool Configuration**  
Configure tools, their locations and automatic installers.



**Manage Plugins**  
Add, remove, disable or enable can extend the functionality of



**Manage Nodes and Clouds**  
Add, remove, control and monitor the various nodes that Jenkins runs jobs on.

## Security



**Configure Global Security**  
Secure Jenkins; define who is allowed to access/use the system.



**Manage Credentials**  
Configure credentials



**Configure Credential Pro**  
Configure the credential provider



**Manage Users**

Jenkins > Nodes >

[Back to Dashboard](#)

[Manage Jenkins](#)

[New Node](#)

[Configure Clouds](#)

[Node Monitoring](#)

S	Name ↓	Architecture	Clock Difference	Free Disk Space	Free Swap Space
	<a href="#">master</a>	Linux (amd64)	In sync	6.21 GB	
	Data obtained	10 min	10 min	10 min	10 min

**Build Queue**

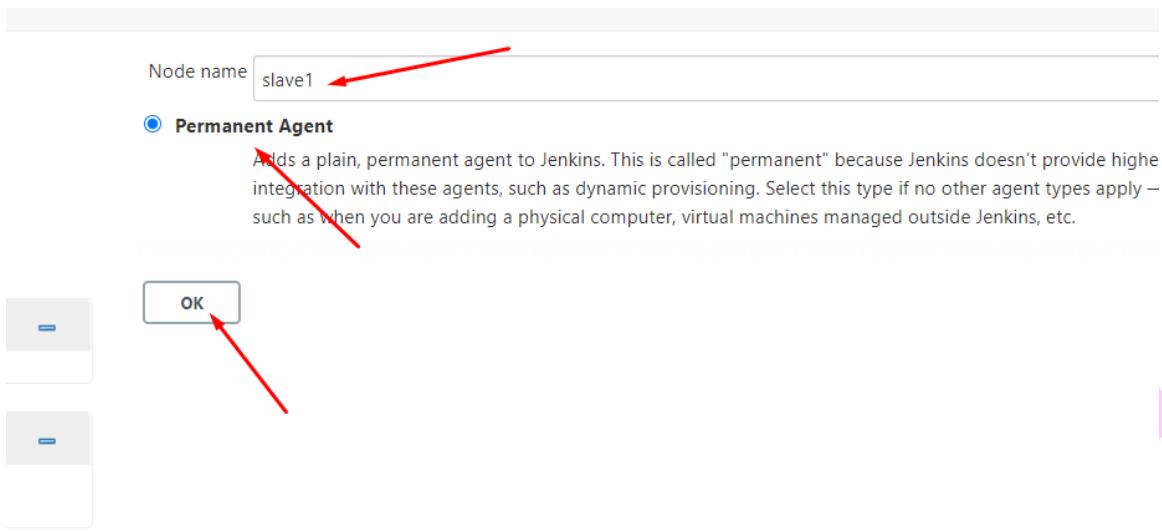
No builds in the queue.

**Build Executor Status**

1 Idle

2 Idle

DVS Technologies, Opp Home Town, Beside Biryani Zone, Marathahalli, Bangalore Phone: 9632558585 Mobile: 8892499499 Mail : [dvs.training@gmail.com](mailto:dvs.training@gmail.com) Web: [www.dvstechnologies.in](http://www.dvstechnologies.in)



A screenshot of the same Jenkins node configuration dialog, but with several fields highlighted by red boxes. The highlighted fields are: "Name" (value "slave1"), "Description" (value "slave1"), "# of executors" (value "4"), "Remote root directory" (value "/home/jenkins"), "Labels" (value "slave1"), "Usage" (value "Use this node as much as possible"), and "Launch method" (value "Launch agents via SSH"). A large red arrow points from the text "DV Technologies" at the bottom of the page to the "Usage" dropdown. Another large red arrow points from the text "DV Technologies" to the "Launch method" dropdown.

Name	slave1
Description	slave1
# of executors	4
Remote root directory	/home/jenkins
Labels	slave1
Usage	Use this node as much as possible
Launch method	Launch agents via SSH

Launch method: Launch agents via SSH

Host: slave1

Credentials: - none - **Add**

The selected Jenkins user cannot be found

Host Key Verification Strategy: Known hosts file Verification Jenkins Credentials Provider

Availability: Keep this agent online as much as possible

**Advanced...**

### Add Credentials

Domain: Global credentials (unrestricted)

Kind: SSH Username with private key

Scope: Global (Jenkins, nodes, items, all child items, etc)

ID: slave1privatekey

Description: slave1privatekey

Username: jenkins

Private Key: Enter directly

No Stored Value **Add**

Usage	Use this node as much as possible
Scope	Global (Jenkins, nodes, items, all child items, etc)
ID	slave1privatekey
Description	slave1privatekey
Username	jenkins
Private Key	<input checked="" type="radio"/> Enter directly <input type="radio"/> Key <span style="border: 1px solid red; padding: 2px;">-----BEGIN RSA PRIVATE KEY----- MIIEpaIBAAKCAQEAz0WDd+16MrpDK49FcYsckLSzejSvppbqJZ19z0gdGQb4ao+j51KGUzORvXhVRomzGWWROFH9jWlCyK/W4E8KPx8JArkud+BLPDCSJbzq13xFwNg09VbfZW32pXe2Y92aae3heQbMcFEQ1IcU2U+SfYScCT4B0ojtjS0cVng1TEpp6MqG3i08Ec4dnwSqOhKs0RMfhEpXB2pGueBam9hLucWbBjP1z1QxrHA/5EvazeUamyXueNf3m/SqQCoLxyB2h6u6RuvePX6BXH8Qz0j1uSgnCnQIDAQABAoIBAHmf1gvg8PsYflavQ8LFG3AkUUmXi1lxkrZvaRD918TAHHHD63tRTb2p/wfi/WDFImnDfTKXgQhkzzJ+VOAkhyGj5e212MR51yBtAI/Y/wbcmmf+rJySoPzbByXm4pozcE8tqGooA+CG+SeZ5cw+S9as09DpZ0je10QCzGiv8TpB4muWqb1LPE3Vdr1uExeGol5KyZrt8GHnEn8AqVg9hjL314a0p5uV2RhxGotsf2SKsgH1vXUvHvttAVCYG82QFWCXU+MwEcgYE8RJJd+AkBzPipWX1V5j/u091uRkms5dYxZ8rvTICDTFYEHMV+OR6TFRxKya7TwXkuOmlgg/e0L1+echKzvBoye53VwDzHKUlfrcog17NiObHwvGtbFANC0KjYi8w+Up+GcdCPvTPRbn</span>
Passphrase	

*copy the private key*

Launch method

Launch agents via SSH

Host slave1

Credentials jenkins (slave1privatekey) Add

Host Key Verification Strategy Not verifying Verification Strategy

Availability Keep this agent online as much as possible

Advanced...

Node Properties

Disable deferred wipeout on this node

Environment variables

Tool Locations

Save

Nodes

S	Name ↓	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Tem
	master	Linux (amd64)	In sync	6.21 GB	0 B	
	slave1	Linux (amd64)	In sync	6.46 GB	0 B	
	Data obtained 3.7 sec		3.6 sec	3.6 sec	3.6 sec	3.6 sec

queue.

Job Status

Executing build job in slave1

Jenkins

- 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

master

[add descrip](#)

All	W	Name ↓	Last Success	Last Failure	Last Duration
		buildjob1	1 hr 1 min - #1	N/A	0.13 sec

Icon: S M L

Legend: [Atom feed for all](#) [Atom feed for failures](#) [Atom feed for just latest build](#)

**Enter an item name**

testingslave1  
» Required field

**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.

for 54.90.27.4... **OK**

testingslave1

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

Disable this project

Execute concurrent builds if necessary

Restrict where this project can be run

Label Expression: slave1

slave1 is serviced by 1 node. Permissions or other restrictions provided by plugins may prevent this job from running on those nodes. **Advanced...**

**Source Code Management**

None

Git

Subversion

## Build

Execute shell

Command

```
#!/bin/bash  
uname -a  
hostname -i|
```

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

Add build step ▾

**Save** **Apply**

A screenshot of the Jenkins 'Build' configuration page. It shows a 'Execute shell' section with a command input field containing '#!/bin/bash\nuname -a\nhostname -i|'. Below the command is a link to 'the list of available environment variables'. At the bottom are 'Save' and 'Apply' buttons. A red arrow points from the top-left towards the 'Save' button.

Jenkins > testingslave1 >

Back to Dashboard Status Changes Workspace **Build Now** Delete Project Configure Rename

## Project testingslave1

Workspace Recent Changes

### Permalinks

Build History trend — find Atom feed for all Atom feed for failures

A screenshot of the Jenkins 'testingslave1' project page. It displays a sidebar with links: Back to Dashboard, Status, Changes, Workspace, Build Now (which has a red arrow pointing to it), Delete Project, Configure, and Rename. The main area shows the title 'Project testingslave1', workspace and recent changes links, and a 'Permalinks' section. Below that is a 'Build History' section with a search bar and atom feeds. A large pink checkmark is overlaid on the left side of the page.

[Back to Dashboard](#)

[Status](#)

[Changes](#)

[Workspace](#)

[Build Now](#)

[Delete Project](#)

[Configure](#)

[Rename](#)

## Project testingslave1



[Workspace](#)



[Recent Changes](#)

### Permalinks

#### Build History

trend

find

x

#1

Jul 30, 2020 2:57 AM



[Atom feed for all](#)



[Atom feed for failures](#)

- [Last build \(#1\), 24 sec ago](#)
- [Last stable build \(#1\), 24 sec ago](#)
- [Last successful build \(#1\), 24 sec ago](#)
- [Last completed build \(#1\), 24 sec ago](#)

Back to Project

Status

Changes

Console Output

Edit Build Information

Delete build '#1'



### Build #1 (Jul 30, 2020 2:



No changes.



Started by user [admin](#)

Jenkins > testingslave1 > #1

Back to Project Status Changes Console Output View as plain text Edit Build Information Delete build '#1'

## Console Output

Started by user [admin](#)  
Running as SYSTEM  
Building remotely on [slave1](#) in workspace /home/jenkins/workspace,  
[testingslave1] \$ /bin/bash /tmp/jenkins1298549368843896533.sh  
Linux slave1 4.14.186-146.268.amzn2.x86\_64 #1 SMP Tue Jul 14 18:  
fe80::c04:b8ff:fe74:cf9%eth0 172.31.33.185  
Finished: SUCCESS

## Plugin Manager

Installing thin backup plugin :

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

All +

S	W	Name ↓	Last Success	Last Failure
blue	sun	<a href="#">buildjob1</a>	1 hr 12 min - #1	N/A
blue	sun	<a href="#">testingslave1</a>	8 min 22 sec - #1	N/A

Icon: S M L Legend Atom feed for all Atom feed for failu

Build Queue - No builds in the queue.

Build Executor Status - master



### Global Tool Configuration

Configure tools, their locations and automatic installers.



### Manage Plugins

Add, remove, disable or enable plugins that can extend the functionality of Jenkins.



### Manage Credentials

Configure credentials



### Configure Credential Providers

thinbackup

Updates

Available

Installed

Advanced

Enabled Name ↓

Version Previous

png not installed

The screenshot shows the Jenkins Update Center interface. A search bar at the top contains the text "thinbackup". Below it, tabs for "Updates", "Available" (which is selected), "Installed", and "Advanced" are visible. A table lists available plugins, with "ThinBackup" from the "Miscellaneous" category highlighted. The table includes columns for "Name", "Version", and "Released". The "ThinBackup" entry shows version 1.9 and a release date of 3 yr 7 mo ago. A note below the table states: "Backups the most important global and job specific configuration files." At the bottom of the page are three buttons: "Install without restart" (highlighted with a red box and arrow), "Download now and install after restart", and "Check now". A status message says "Update information obtained: 1 day 0 hr ago".

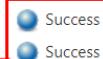
## Installing Plugins/Upgrades

### Preparation

- Checking internet connectivity
- Checking update center connectivity
- Success

### ThinBackup

#### Loading plugin extensions



Success

Success

[Go back to the top page](#)

(you can start using the installed plugins right away)

Restart Jenkins when installation is complete and no jobs are running

Back to Dashboard  
 Manage Jenkins  
 Manage Plugins

## Installing Plugins/Upgrades

### Preparation

- Checking internet connectivity
- Checking update center connectivity
- Success

### ThinBackup

#### Loading plugin extensions



Success



Success

[Go back to the top page](#)

(you can start using the installed plugins right away)

Restart Jenkins when installation is complete and no jobs are running

thinbackup

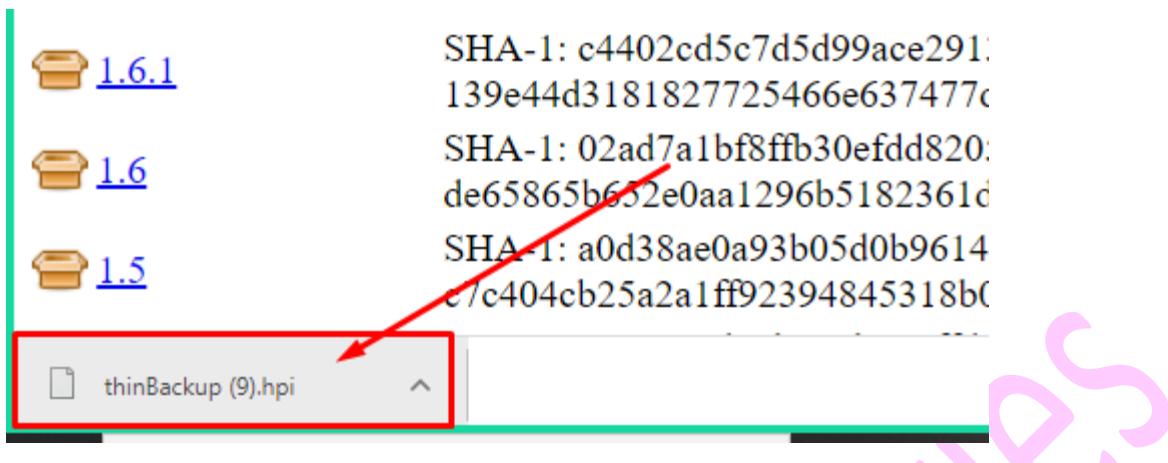
Updates Available Installed Advanced

Enabled	Name ↓	Version	Previously installed version	Uninstall
<input checked="" type="checkbox"/>	ThinBackup Backups the most important global and job specific configuration files.	1.9		<a href="#">Uninstall</a>

## Downgrade the plugin version:

permalink to the latest

- [1.9](#) SHA-1: a651c33c219f91310cb89d06a97c706103bb2f13, SHA-256: a17b94e35a525b741e6a8b05342730f96a244bdc26b3d435164ba42fa6d198da
- [1.8](#) SHA-1: f5456144eff2778d1ee77af7a21345638375bdad, SHA-256: d5b435adca3b598a82863a87b466a010ebcd134e483660ea4b51aa384095a3e8
- [1.7.4](#) SHA-1: 07594b84c1d2d79d4d3a544bd179dbe997bb6287, SHA-256: c288d1f5eee319f8884bc1cba5567eee32762718cead19e6f83ee1b32a656664
- [1.7.3](#) SHA-1: 87fb3eb6dfac1aa37eac164aadеб0ebc85b00d18, SHA-256: 1c4e8b8cce664ed2725abdfе83191cb3594f23b92cdb168a1d124c7c1a7f4b5aa
- [1.7.2](#) SHA-1: e3cf6fd0f120350acde4dd2b2c1df3ea2c6d6203, SHA-256: 9828902a2a472f133a5e1fa02c0539447cc9330ce3d07c86687b26036fe3c0f0
- [1.7](#) SHA-1: bab693ecf3155beefdc50f416d8c9e5e94cd6af0, SHA-256: e966248dca7ec3e5618c09f5e009af99292f760c12e985feedff3769def0a9d
- [1.6.2](#) SHA-1: e9204f6642eb79bfe9ae4dd5fce69a2bd6351873, SHA-256: ff36f2a1bb1c4a9529195babba549f5f9c93eb31f22343739eb6ed89187883da
- [1.6.1](#) SHA-1: c4402cd5c7d5d99ace291387149b6ac889766e78, SHA-256: 139e44d3181827725466e637477d5f32e8888fcce342d885734d1f690a7a4507
- [1.6](#) SHA-1: 02ad7a1bf8ffb30efdd82050ba826ece41e86eab, SHA-256: de65865b652e0aa1296b5182361d21900b4e8bf0a5a0633bd480ae25c55e88bf



Jenkins ▾ > **Plugin Manager**

Back to Dashboard Manage Jenkins Update Center

Updates Available Installed **Advanced**

### HTTP Proxy Configuration

Server

Port

User name

Password

No Proxy Host

**Submit**

### Upload Plugin

You can upload an .hpi file to install a plugin from outside the central plugin repository.

File:  Choose File No file chosen

**Upload**

### Update Site

URL  https://updates.jenkins.io/update-center.json

**Submit**

## Upload Plugin

You can upload an .hpi file to install a plugin from outside the central plugin repository.

File:  thinBackup (9).hpi

**Upload**

## Update Site

URL

## Installing Plugins/Upgrades

### Preparation

- Checking internet connectivity
- Checking update center connectivity
- Success

ThinBackup Success

Loading plugin extensions Success

### Preparation

thinBackup Warning

thinBackup plugin is already installed. Jenkins needs to be restarted for the update to take effect.

[Go back to the top page](#)

(you can start using the installed plugins right away)

Restart Jenkins when installation is complete and no jobs are running

Page generated:  
Jul 30, 2020 3:22:11 AM UTC

[REST API](#)

Jenkins 2.235.3



Please wait while Jenkins is restarting ...

Your browser will reload automatically when Jenkins is ready.

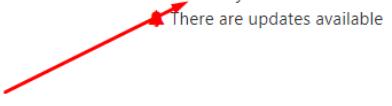
## Verification:

 No implementation of access control for builds is present. It is recommended that you install [Authorize Project Plugin](#) or another plugin implementing the [QueueItemAuthenticator](#) extension point.

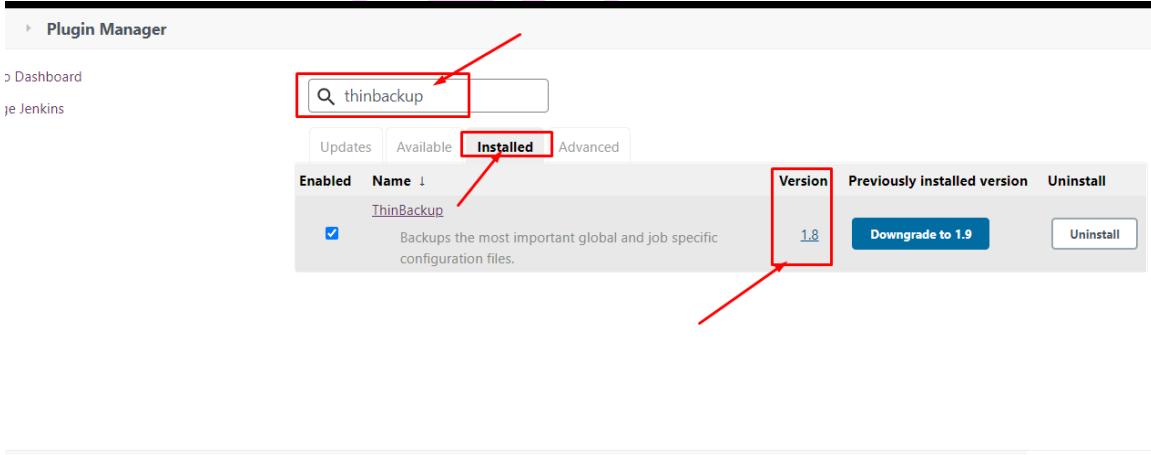
### System Configuration

 **Configure System**  
Configure global settings and paths.

 **Global Tool Configuration**  
Configure tools, their locations and automatic installers.

 **Manage Plugins**  
Add, remove, disable or enable plugins that can extend the functionality of Jenkins.  
 There are updates available

 **Manage Nodes and Clouds**  
Add, remove, control and monitor the various nodes that Jenkins runs jobs on.

  
Plugin Manager  
Dashboard Jenkins  
Search bar: thinbackup  
Filter buttons: Updates Available Installed Advanced  
Table columns: Enabled Name Description Version Previously installed version Uninstall  
Row: ThinBackup (Enabled, Name: ThinBackup, Description: Backups the most important global and job specific configuration files., Version: 1.8, Previously installed version: 1.9, Uninstall button)

## Uninstall:

Screenshot of the Jenkins plugin manager showing the 'Installed' tab. A red arrow points to the 'Uninstall' button for the 'ThinBackup' plugin.

Enabled	Name ↓	Version	Previously installed version	Uninstall
<input checked="" type="checkbox"/>	<a href="#">ThinBackup</a> Backups the most important global and job specific configuration files.	1.8	<a href="#">Downgrade to 1.9</a>	<a href="#">Uninstall</a>

## Working with Parameters

Screenshot of the Jenkins dashboard. A red arrow points to the 'New Item' link in the left sidebar.

Jenkins

- 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.

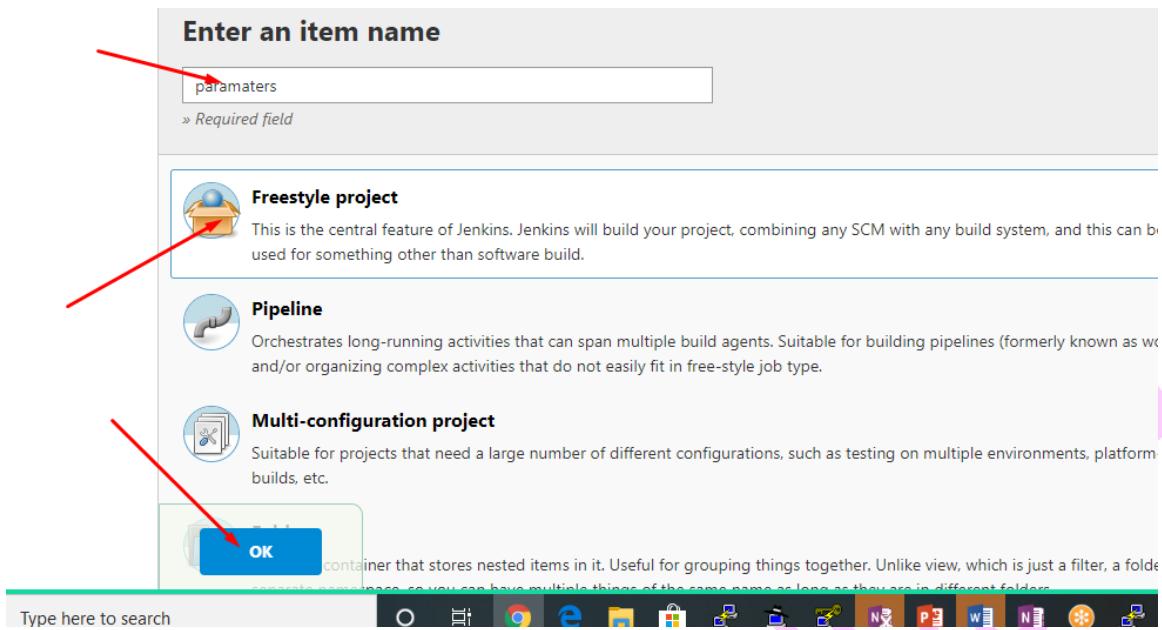
Build Executor Status

master  
1 Idle

All	W	Name ↓	Last Success	Last Failure	Last Duration
●	☀	<a href="#">buildjob1</a>	23 hr - #1	N/A	0.13 sec
●	☀	<a href="#">testingslave1</a>	22 hr - #1	N/A	2.7 sec

Icon: S M L

Legend: [Atom feed for all](#) [Atom feed for failures](#) [Atom feed](#)



## GLOBAL VARIABLE REFERENCE:

The screenshot shows the Jenkins Pipeline Syntax documentation. The URL in the address bar is '54.165.110.104:8080/pipeline-syntax/'. The left sidebar has links for 'Snippet Generator', 'Declarative Directive Generator', 'Declarative Online Documentation', 'Steps Reference', 'Global Variables Reference' (which is highlighted with a red box and has a red arrow pointing to it), 'Online Documentation', 'Examples Reference', and 'IntelliJ IDEA GDScript'. The main content area has a heading 'Overview' and a 'Steps' section with a sample step 'archiveArtifacts: Archive the artifacts'.

## PRINTING DEFAULT VARIABLES/PARAMETERS:

Jenkins Job Configuration - Build

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

**Build**

**Execute shell**

Command:

```
#!/bin/bash
echo "Generic Jenkins Variables"
echo "Build Number $BUILD_NUMBER"
echo "Node Name $NODE_NAME"
echo "Job Name $JOB_NAME"
echo "Executor Number $EXECUTOR_NUMBER"
echo "Workspace $WORKSPACE"
```

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

Advanced...

Add build step ▾

**Post-build Actions**

Save Apply

A screenshot of the Jenkins Job Configuration interface. The 'Build' tab is selected. Under 'Execute shell', a command is defined to print various Jenkins environment variables. Below the command is a link to see the list of available environment variables. There is an 'Advanced...' button. At the bottom, there are 'Save' and 'Apply' buttons. A red arrow points from the 'Build Now' link in the sidebar of the second screenshot to this 'Save' button.

Jenkins > parameters

[Back to Dashboard](#) [Status](#) [Changes](#) [Workspace](#) [Build Now](#) [Delete Project](#) [Configure](#) [Rename](#)

## Project parameters

[Workspace](#) [Recent Changes](#)

## Permalinks

[Build History](#) [trend](#) [find](#) [Atom feed for all](#) [Atom feed for failures](#)

Page generated: Jul 31, 2020 1:49:20

A screenshot of the Jenkins dashboard. On the left, there's a sidebar with links: Back to Dashboard, Status, Changes, Workspace, Build Now (which has a red arrow pointing to it), Delete Project, Configure, and Rename. The main area is titled 'Project parameters'. It shows two links: 'Workspace' and 'Recent Changes'. Below that is a 'Permalinks' section with a 'Build History' table. The table includes a search bar ('find'), a trend dropdown ('trend'), and two atom feed links ('Atom feed for all' and 'Atom feed for failures'). At the bottom right, it says 'Page generated: Jul 31, 2020 1:49:20'.

Delete Project

Configure

Rename

Recent Changes

## Permalinks

Build History [trend](#)

find

#1 Jul 31, 2020 1:49 AM

Atom feed for #1 Took 2.3 sec [Feed for failures](#)

Jenkins > parameters > #1

[Back to Project](#)

[Status](#)

[Changes](#)

[Console Output](#) (highlighted with a red arrow)

[View as plain text](#)

[Edit Build Information](#)

[Delete build '#1'](#)

## Console Output

```
Started by user admin
Running as SYSTEM
Building remotely on slave1 (prasad javaapp) in workspace /home/jenkins/workspace/paramaters
[parameters] $ /bin/bash /tmp/jenkins7401803402011633750.sh
Generic Jenkins Variables
Build Number 1
Node Name slave1
Job Name paramters
Executor Number 1
Workspace /home/jenkins/workspace/paramaters
Finished: SUCCESS
```

## USER DEFINED PARAMETERS:

parameters >

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

[Plain text] Preview

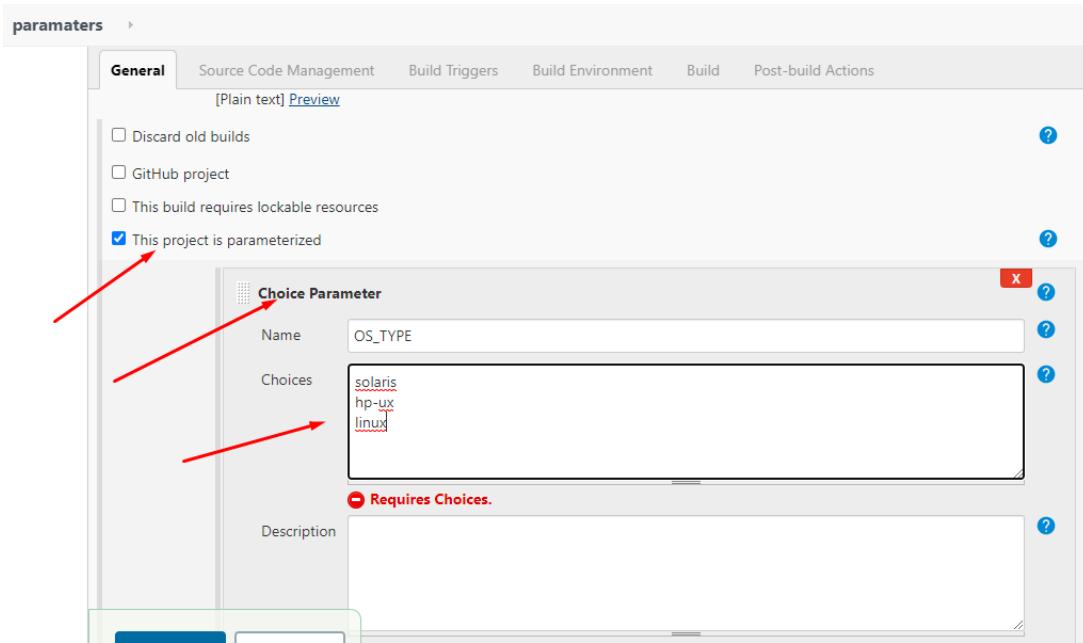
Discard old builds  
 GitHub project  
 This build requires lockable resources  
 This project is parameterized

**Choice Parameter**

Name: OS\_TYPE  
Choices:  
solaris  
hp-ux  
linux

Requires Choices.

Description:



parameters >

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

**String Parameter**

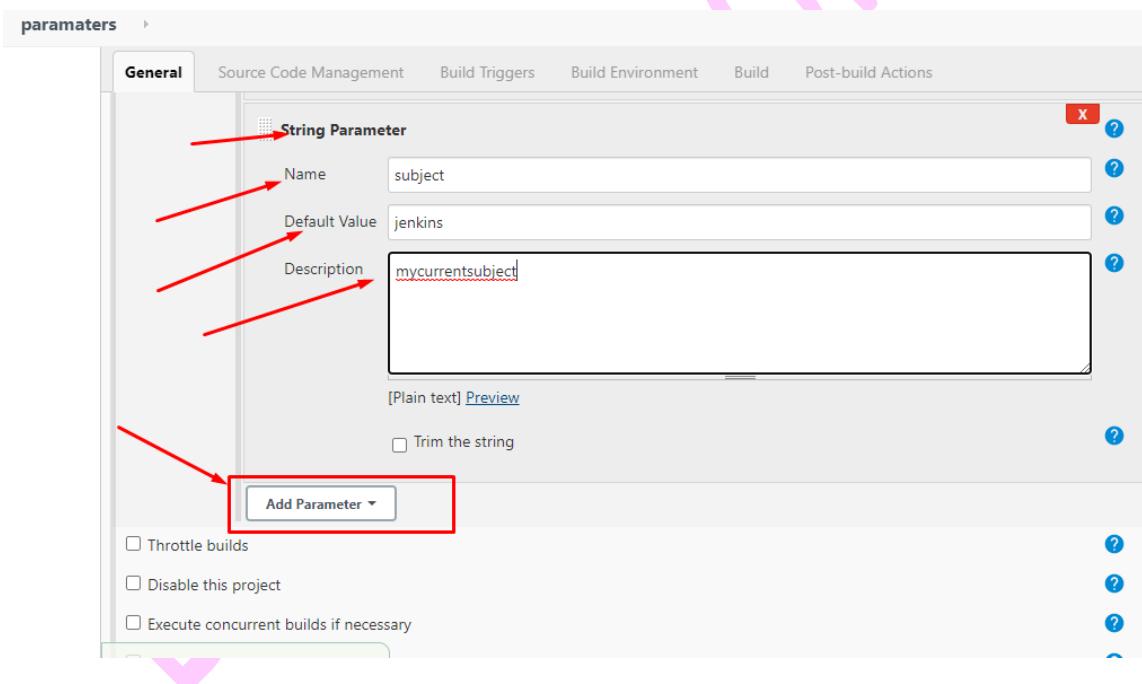
Name: subject  
Default Value: jenkins  
Description: mycurrentsubject

[Plain text] Preview

Trim the string

Add Parameter ▾

Throttle builds  
 Disable this project  
 Execute concurrent builds if necessary



The screenshot shows the Jenkins configuration page for a job named 'env-vars.html'. At the top, there is a section titled 'Execute shell' with a command input field containing the following script:

```
#!/bin/bash  
echo "OS TYPE IS $OS_TYPE"  
echo "CURRENT SUBJECT IS $subject"
```

Below this, a link 'See the list of available environment variables' is visible. To the right of the command field is an 'Advanced...' button. Further down, under 'Post-build Actions', there is a 'Save' button highlighted with a red arrow.

The screenshot shows the 'Project parameters' page for the 'env-vars.html' project. The left sidebar includes links for Back to Dashboard, Status, Changes, Workspace, Build with Parameters (which is highlighted with a red box and a red arrow), Delete Project, Configure, and Rename. The main content area is titled 'Project parameters' and contains sections for 'Workspace' and 'Recent Changes'. Below this is a 'Permalinks' section listing recent builds. At the bottom right, it says 'Page generated Jul 31, 2020 1:1'.

**Project parameters**

Workspace

Recent Changes

**Permalinks**

- Last build (#1), 5 min 40 sec ago
- Last stable build (#1), 5 min 40 sec ago
- Last successful build (#1), 5 min 40 sec ago
- Last completed build (#1), 5 min 40 sec ago

Page generated Jul 31, 2020 1:1

parameters >

ashboard

## Project parameters

This build requires parameters:

Parameters

ject

istory trend =

il 31, 2020 1:49 AM

ed for all Atom feed for failures

OS\_TYPE: linux  
subject: jenkins  
mycurrentsubject

Build

Jenkins > parameters >

Back to Dashboard Status Changes Workspace Build with Parameters Delete Project Configure Rename

## Project parameters

Workspace Recent Changes

### Permalinks

- Last build (#2), 0.23 sec ago
- Last stable build (#2), 0.23 sec ago
- Last successful build (#2), 0.23 sec ago
- Last completed build (#2), 0.23 sec ago

Build History trend =

find

#2 Jul 31, 2020 1:56 AM  
#1 Jul 31, 2020 1:49 AM

Atom feed for all Atom feed for failures

Jenkins > parameters > #2

Back to Project Status Changes **Console Output** View as plain text Edit Build Information Delete build '#2' Parameters Previous Build

## Console Output

Started by user [admin](#)  
Running as SYSTEM  
Building remotely on [slave1](#) (prasad javaapp) in workspace /home/jenkins/workspace/parameters \$ /bin/bash /tmp/jenkins3074722440491991912.sh  
OS TYPE IS linux  
CURRENT SUBJECT IS jenkins  
Finished: SUCCESS

## Working with Git integration

GIT CLONE:

All

### Enter an item name

gitclone » Required field

**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.

**OK**

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

**Source Code Management**

None  
 Git

**Repositories**

Repository URL: `https://github.com/shan5a6/myweb.git`

Credentials: - none -  Jenkins

**Branches to build**

Branch Specifier (blank for 'any'): `*/master`

**Repository browser**

(Auto)

**Additional Behaviours**

Subversion

**Jenkins Credentials Provider: Jenkins**

**Add Credentials**

Domain: Global credentials (unrestricted)  
Kind: Username with password  
Scope: Global (Jenkins, nodes, items, all child items, etc)  
Username: shan5a6  
Password: .....  
ID: gitaccess  
Description: gitaccess

Screenshot of the Jenkins configuration interface for a new job named "Project gitclone". The "Source Code Management" section is selected, showing a Git configuration. Red arrows point to the "Repository URL" field containing "https://github.com/shan5a6/myweb.git" and the "Branch Specifier" field containing "/master".

The configuration continues with the "Build" section, which includes an "Add build step" button. Below it is the "Post-build Actions" section, featuring an "Add post-build action" button. A red arrow points to the "Save" button at the bottom of this section.

The screenshot shows the Jenkins dashboard for the "Project gitclone" job. On the left, a sidebar lists project management options: Back to Dashboard, Status, Changes, Workspace, Build Now (highlighted with a red arrow), Delete Project, Configure, and Rename. The main area displays the "Workspace" and "Recent Changes" sections. At the bottom, there are links for "Build History" and "trend", a search bar, and Atom feed links.

Jenkins > gitclone > #2

[Back to Project](#)

[Status](#)

[Changes](#)

**Console Output** 

[View as plain text](#)

[Edit Build Information](#)

[Delete build '#2'](#)

[Git Build Data](#)

[No Tags](#)

[Previous Build](#)

## Console Output

```
Started by user admin
Running as SYSTEM
Building remotely on slave1 (prasad javaapp) in workspace /home/jenkins/workspace/gitclone
using credential gitaccess
Cloning the remote Git repository
Cloning repository https://github.com/shan5a6/myweb.git
> git init /home/jenkins/workspace/gitclone # timeout=10
Fetching upstream changes from https://github.com/shan5a6/myweb.git
> git --version # timeout=10
using GIT_ASKPASS to set credentials gitaccess
> git fetch --tags --force --progress -- https://github.com/shan5a6/myweb.git +refs
> git config remote.origin.url https://github.com/shan5a6/myweb.git # timeout=10
> git config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/*
> git config remote.origin.url https://github.com/shan5a6/myweb.git # timeout=10
Fetching upstream changes from https://github.com/shan5a6/myweb.git
using GIT_ASKPASS to set credentials gitaccess
> git fetch --tags --force --progress -- https://github.com/shan5a6/myweb.git +refs
> git rev-parse refs/remotes/origin/master^{commit} # timeout=10
> git rev-parse refs/remotes/origin/origin/master^{commit} # timeout=10
Checking out Revision e5f21b8c17d5413eeeabe2beb3538cb879dd6699 (refs/remotes/origin/master)
> git config core.sparsecheckout # timeout=10
> git checkout -f e5f21b8c17d5413eeeabe2beb3538cb879dd6699 # timeout=10
Commit message: "Merge pull request #60 from shan5a6/release"
First time build. Skipping changelog.
Finished: SUCCESS
```

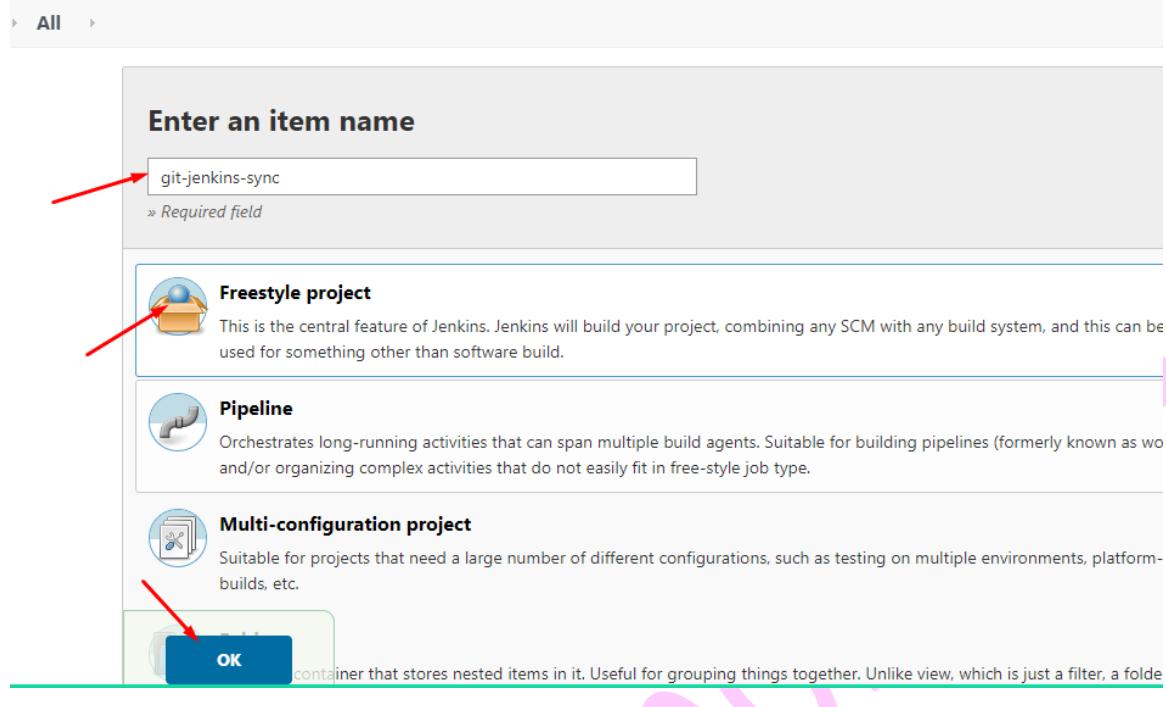
GITHUB with JENKINS Integration:

Jenkins >

 [New Item](#) 

 [People](#)

 [Build History](#)



A screenshot of the Jenkins 'Source Code Management' configuration for the 'git-jenkins-sync' project. The 'General' tab is selected. The 'Source Code Management' tab is active. Under 'Git', the 'Repository URL' is set to 'https://github.com/shan5a6/myweb.git' and the 'Credentials' dropdown shows 'shan5a6/\*\*\*\*\*\*\*\* (qitaccess)'. The 'Branches to build' section has 'Branch Specifier (blank for 'any')' set to '\*/\*master'. The 'Additional Behaviours' section has an 'Add' button. A red arrow points to the 'Advanced...' button in the 'Git' configuration area.

*Mandatory*

The screenshot shows the 'Build Triggers' section of a Jenkins job configuration. The 'GitHub hook trigger for GITScm polling' checkbox is checked and highlighted with a red arrow. Other options like 'Trigger builds remotely' and 'Build periodically' are also present.

The screenshot shows the 'Build' section of a Jenkins job configuration. A 'Execute shell' step is defined with the command: `#!/bin/bash  
echo "Welcome to Dvs Devops!"`. The 'Save' button at the bottom is highlighted with a red arrow.

In Git repo do the below:

The image shows two screenshots of the GitHub interface. The top screenshot displays the main repository page for 'shan5a6/myweb'. It includes a navigation bar with links like 'Code', 'Issues', 'Pull requests', 'Marketplace', and 'Explore'. A red box highlights the 'Settings' button in the top right corner. The bottom screenshot shows the 'Webhooks' section of the 'Settings' page. On the left is a sidebar with options such as 'Options', 'Manage access', 'Security & analysis', 'Branches', 'Webhooks' (which is highlighted with a red box), 'Notifications', 'Integrations', 'Deploy keys', 'Secrets', and 'Actions'. The main content area is titled 'Webhooks' and contains a brief description: '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](#)'. A red box highlights the 'Add webhook' button at the bottom right of this section.

Options  
Manage access  
Security & analysis  
Branches  
**Webhooks**  
Notifications  
Integrations  
Deploy keys  
Secrets  
Actions  
Moderation  
Interaction limits

Webhooks / Add webhook

We'll send a POST request to the URL below with details of any subscribed events. You can also specify which data format you'd like to receive (JSON, x-www-form-urlencoded, etc). More information can be found in our developer documentation.

Payload URL \*

Content type

Secret

Which events would you like to trigger this webhook?

Just the push event.  
 Send me everything.  
 Let me select individual events.

Active  
We will deliver event details when this hook is triggered.

**Add webhook**

Webhooks	
<p>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 <a href="#">Webhooks Guide</a>.</p>	<b>Add webhook</b>
<p><input checked="" type="checkbox"/> <a href="http://54.165.110.104:8080/github-webhook/">http://54.165.110.104:8080/github-webhook/ (push)</a></p>	<b>Edit</b> <b>Delete</b>

## Final Testing:

Perform the below to test the connectivity

The screenshot shows the Jenkins interface for the 'git-jenkins-sync' job. On the left, the build history is displayed, with the first entry (#1) from July 31, 2020, at 2:30 AM highlighted by a red box. A red arrow points from this entry to the terminal log on the right. The terminal log shows the command history and the execution of several Git commands to clone a repository, change directory, and push changes to the master branch, all enclosed in a red box.

```
root@slave1:/tmp/myweb#
[root@slave1 myweb]# history |tail -17
20 git clone https://github.com/shan5a6/myweb.git
21 cd myweb/
22 vi test
23 git status
24 git add .
25 git commit -m "testing the connectivity"
26 git config --global user.email shahan.aix@gmail.com
27 git config --global user.name shan5a6
28 git config --global user.email shahan.aix@gmail.com
29 git commit -m "testing the connectivity"
30 git push origin master
31 history|tail 10
32 history |tail -10
33 history |tail -12
34 history |tail -20
35 history |tail -15
36 history |tail -17
[root@slave1 myweb]#
```