Jenkins

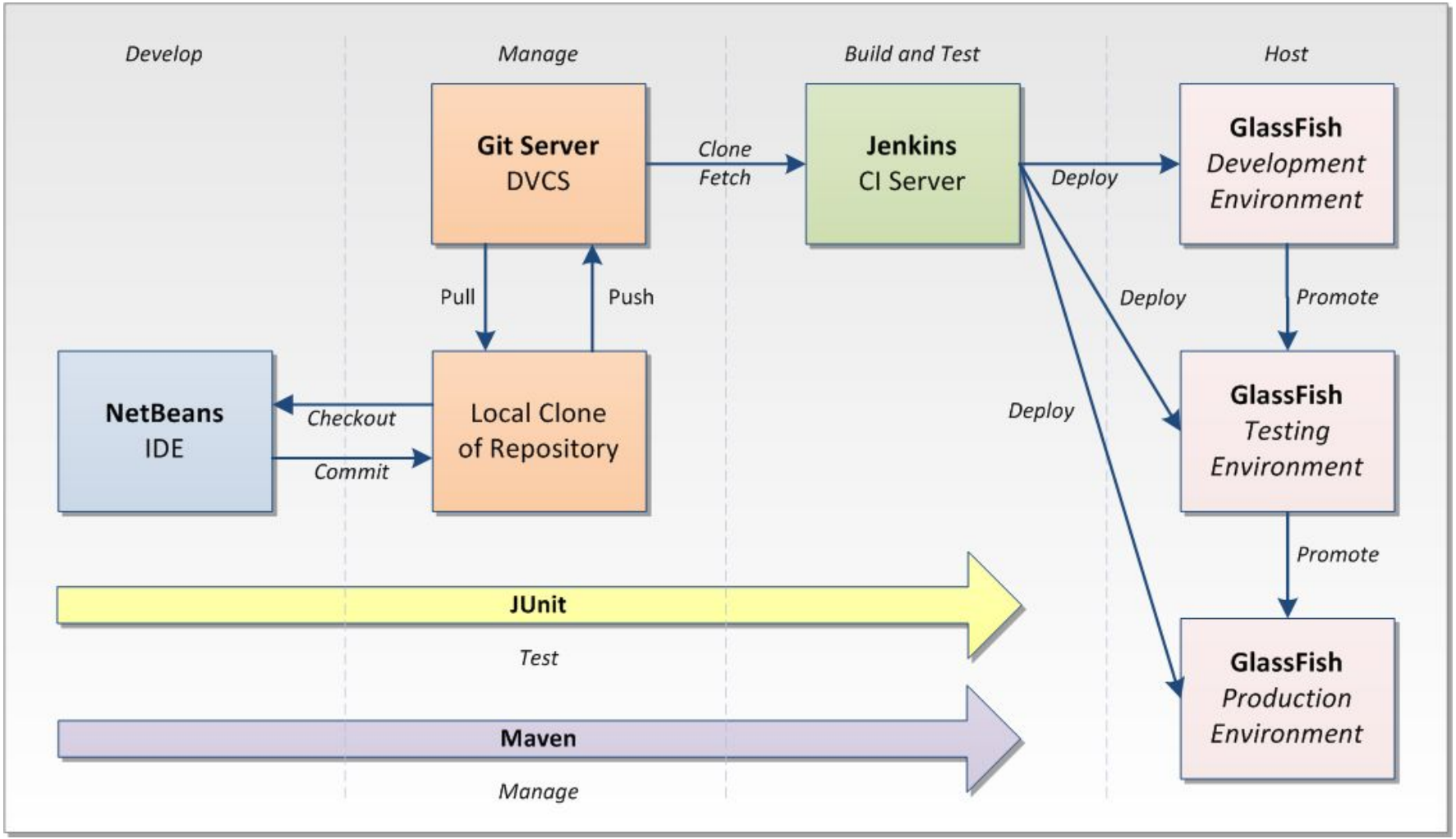
Continuous Integration and Continuous Delivery (CICD)

Introduction

Jenkins is an open source automation tool written in Java with plugins built for Continuous Integration purpose. Jenkins is used to build and test your software projects continuously making it easier for developers to integrate changes to the project, and making it easier for users to obtain a fresh build. It also allows you to continuously deliver your software by integrating with a large number of testing and deployment technologies.

With Jenkins, organizations can accelerate the software development process through automation. Jenkins integrates development life-cycle processes of all kinds, including build, document, test, package, stage, deploy, static analysis and much more.

Jenkins achieves Continuous Integration with the help of plugins. Plugins allows the integration of Various DevOps stages. If you want to integrate a particular tool, you need to install the plugins for that tool. For example: Git, Maven 2 project, Amazon EC2, HTML publisher etc.

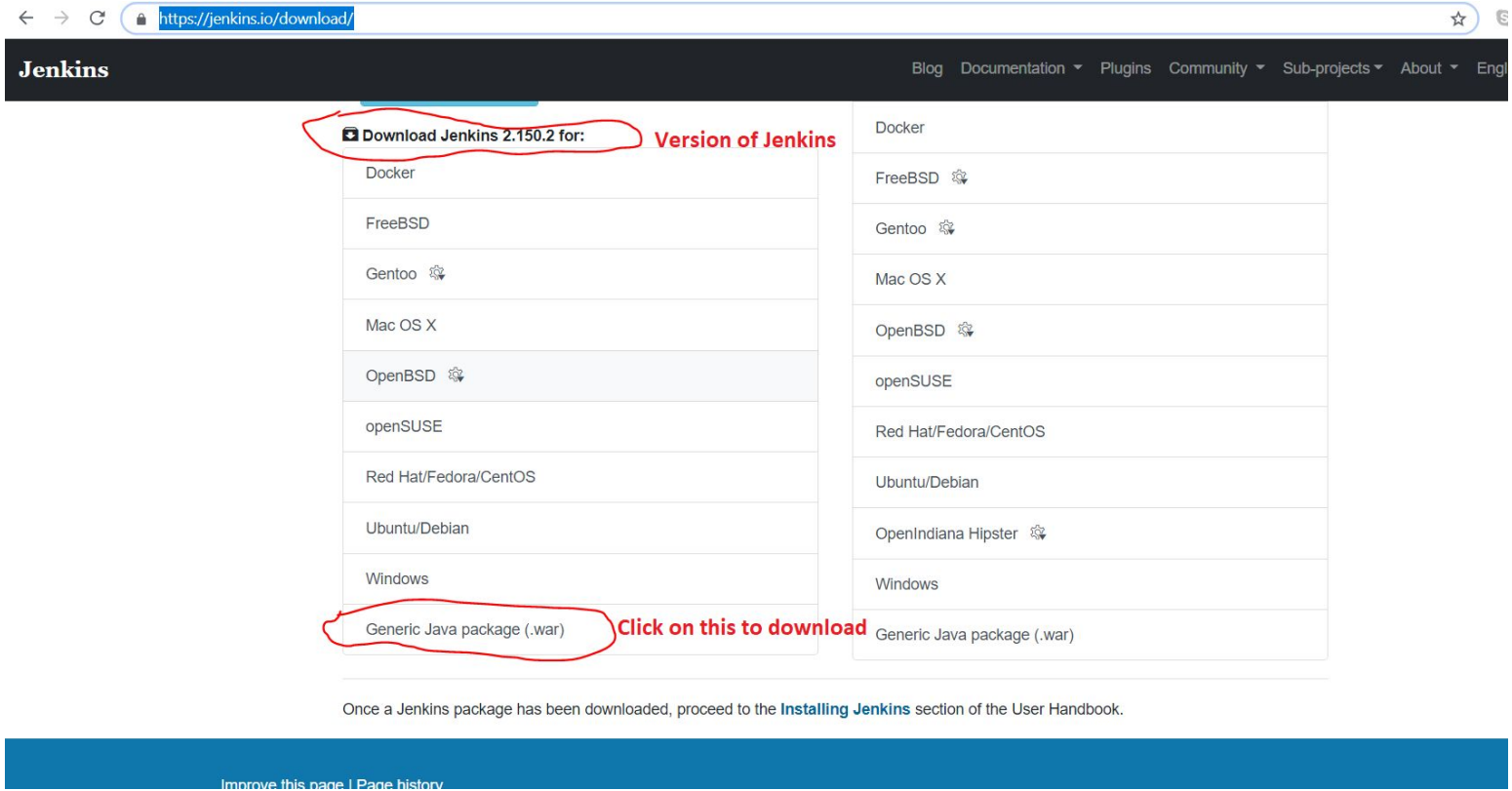

Develop
Manage
Build and Test
Host
Git Server
DVCS
Clone
Fetch
Jenkins
CI Server
Deploy
GlassFish
Development
Environment
Pull
Push
Deploy
Promote
NetBeans
IDE
Checkout
Commit
Local Clone
of Repository
Deploy
GlassFish
Testing
Environment
Promote
JUnit
Test
GlassFish
Production
Environment
Maven
Manage

TOPICS

1. Installing and Configuring Jenkins
2. How to Find Home Directory
3. How to create Users, Manage Users, Roles
4. Getting Started with JOBS
5. Running Job remotely
6. Scheduling Build Jobs
7. Jenkins Integrations with GIT
8. CICD using GIT, Maven and Tomcat server
9. Sending Email Notification using Jenkins

1. Installing and Configuring Jenkins

Step 1:
Download JDK, Tomcat & Jenkins.war
Login into instance ---> change to root user ---> update your instance ---> change to OPT dir ---> download JDK, Tomcat7 & Jenkins.war
$sudo su root
#cd /
#apt-get update
#cd /opt/
#wget https://jdk1.8……..tar.gz
#wget https://apache-tomcat…….tar.gz
Open your firefox browser → https://jenkins.io/download/



#wget http://ftp-chi.osuosl.org/pub/jenkins/war-stable/2.150.2/jenkins.war

Step 2: Setting paths for Java / JDK and tomcat

Unzip JDK and apache-tomcat

#tar -xzf jdk1.8……..tar.gs ------> for unzip jdk

#tar -xzf apache-tomcat7……tar.gz ------> for unzip tomcat

#mv jdk1.8…… jdk1.8 ---> for changing name to jdk1.8

#mv apache-tomcat----- tomcat7 ----> for changing name to tomcat7

#vi ~/.profile

```
export JAVA_HOME="/opt/jdk1.8"
export PATH=$JAVA_HOME/bin:$PATH
export CATALINA_HOME="/opt/tomcat7"
export PATH=$CATALINA_HOME/bin:$PATH
~
```

source ~/.profile ---> to update environment variables

java -version ----> to check java version, for confirmation of installation

Move jenkins.war file from /opt/ to /opt/tomcat7/webapp/

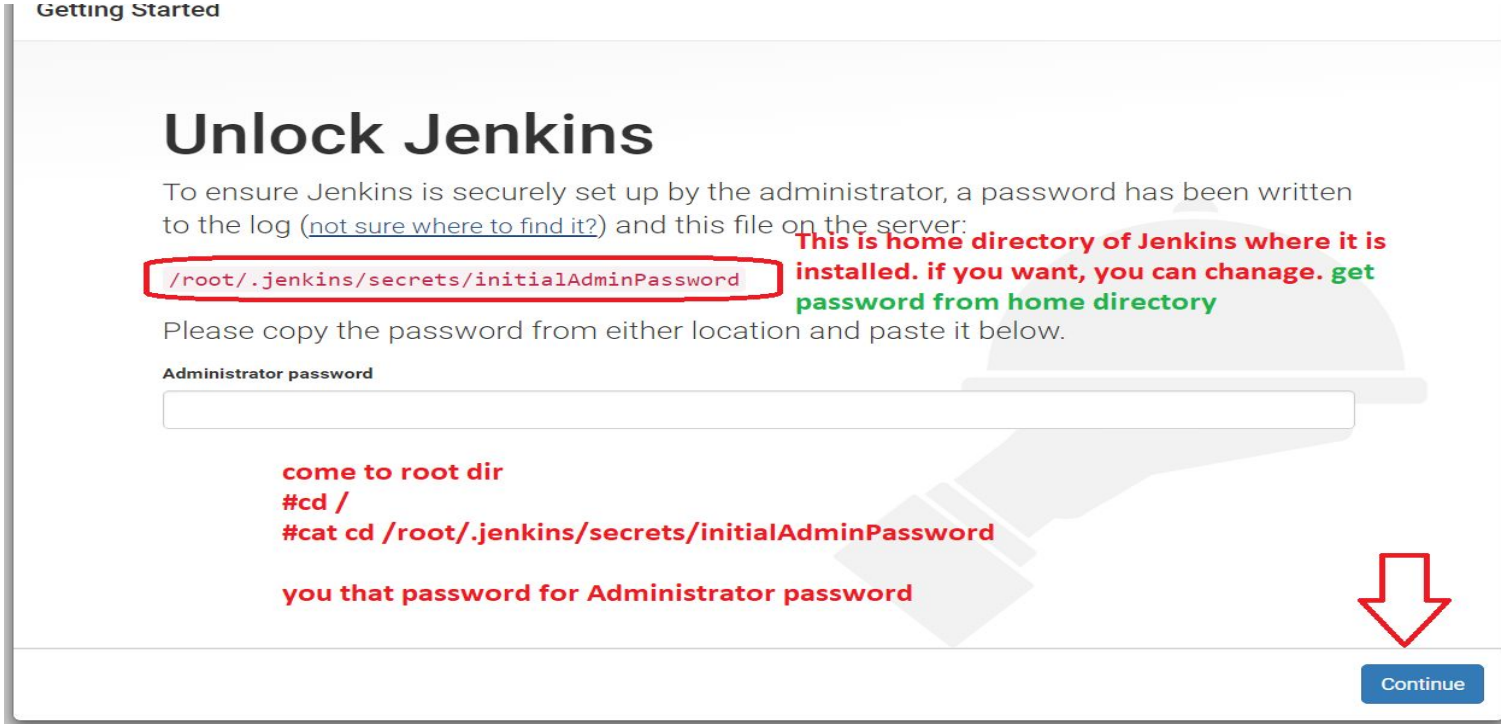
#mv jenkins.war tomcat7/webapps/

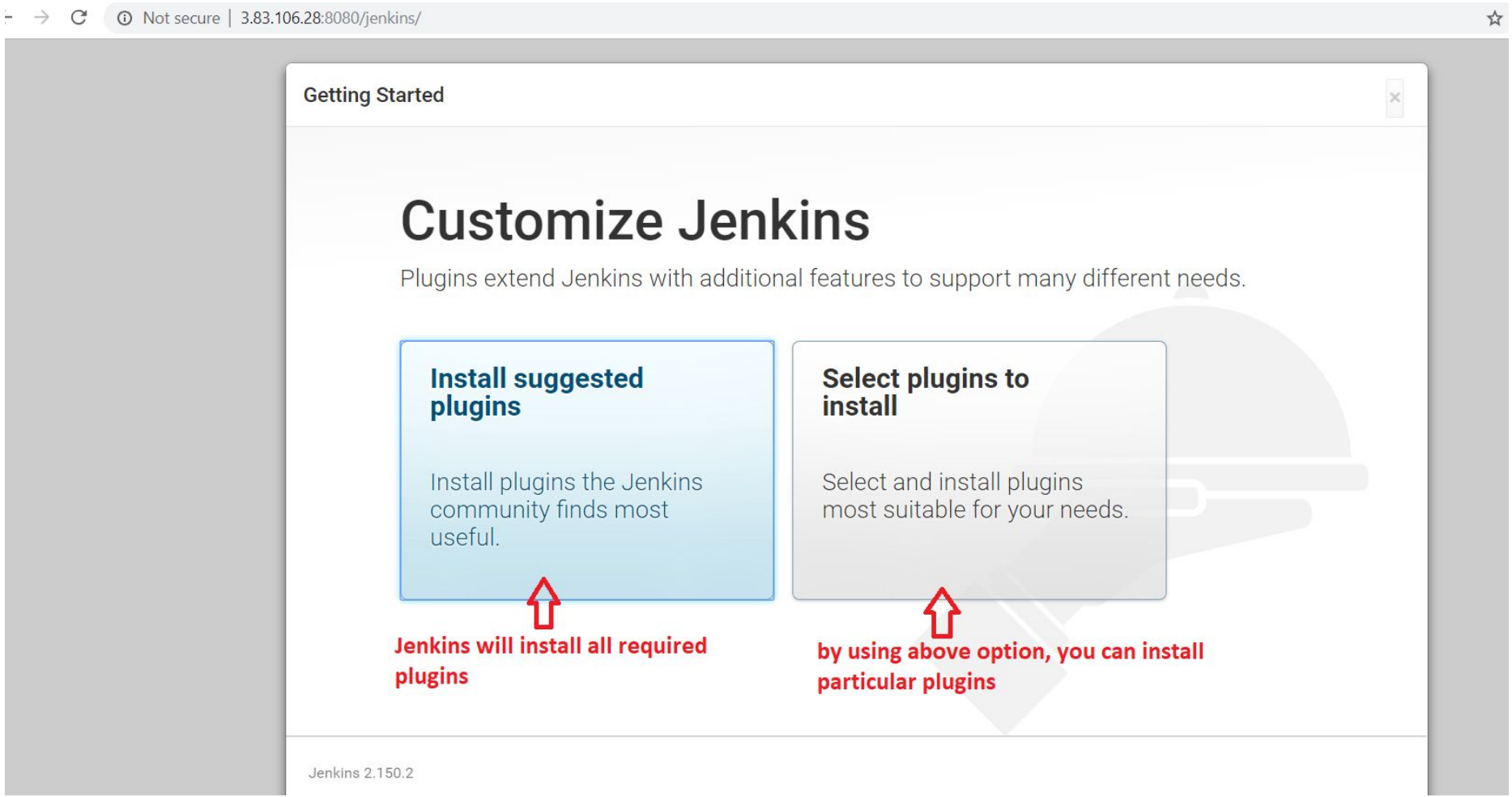
#startup.sh -----> to start tomcat server

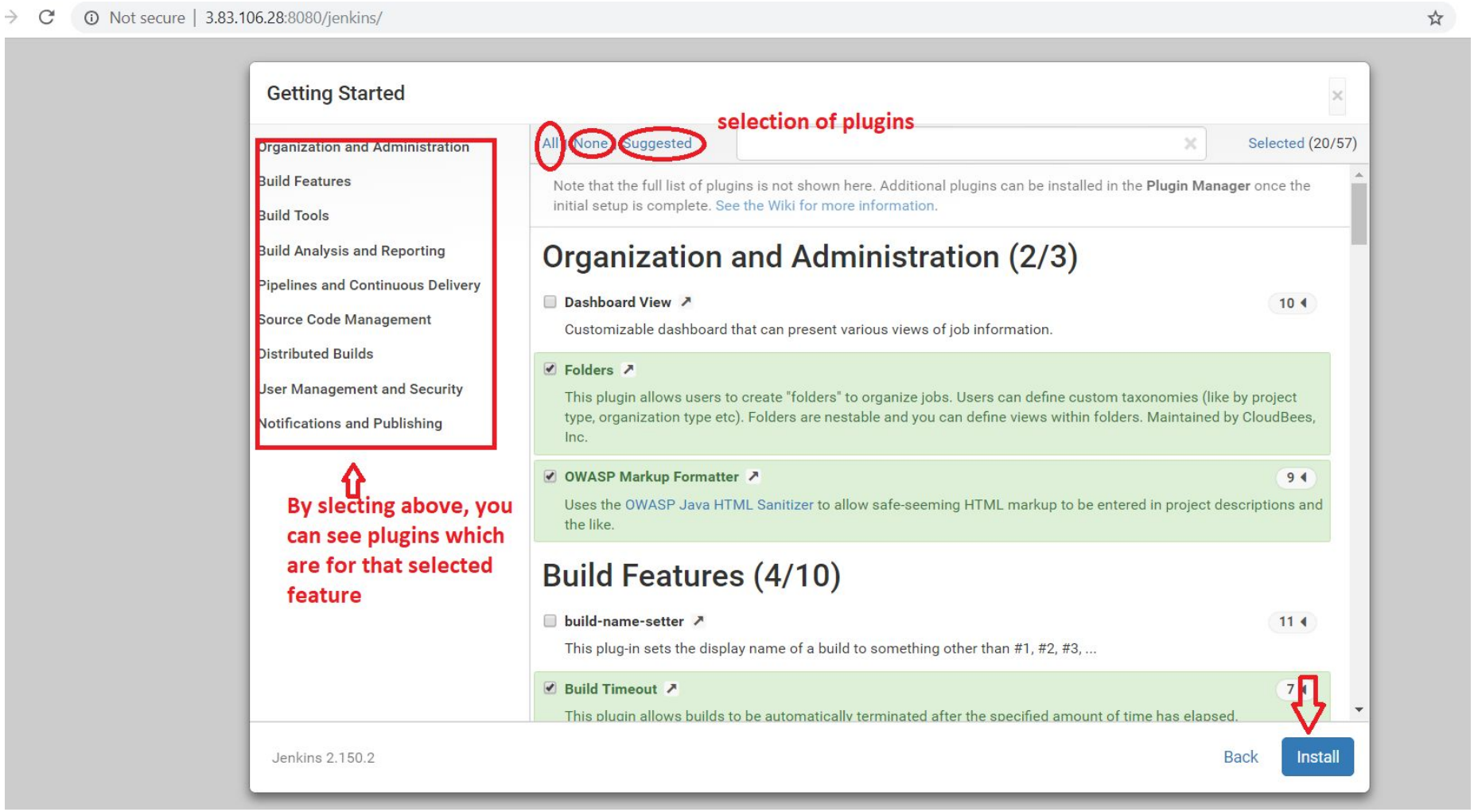
```
root@ip-172-31-91-116:/opt/tomcat7/webapps# ls
docs  examples  host-manager  jenkins.war  manager  ROOT
root@ip-172-31-91-116:/opt/tomcat7/webapps# startup.sh
Using CATALINA_BASE:   /opt/tomcat7
Using CATALINA_HOME:   /opt/tomcat7
Using CATALINA_TMPDIR: /opt/tomcat7/temp
Using JRE_HOME:        /opt/jdk1.8
Using CLASSPATH:       /opt/tomcat7/bin/bootstrap.jar:/opt/tomcat7/bin/tomcat-ju
li.jar
Tomcat started.
root@ip-172-31-91-116:/opt/tomcat7/webapps#
```

Copy instance public ip, open browser

<instance public ip>:8080/jenkins (tomcat runs on 8080 port default)




Not secure | 3.83.106.28:8080/jenkins/
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.
Jenkins will install all required plugins
by using above option, you can install particular plugins
Jenkins 2.150.2


Not secure | 3.83.106.28:8080/jenkins/
Getting Started
selection of plugins
Organization and Administration
Build Features
Build Tools
Build Analysis and Reporting
Pipelines and Continuous Delivery
Source Code Management
Distributed Builds
User Management and Security
Notifications and Publishing
All
None
Suggested
Selected (20/57)
Note that the full list of plugins is not shown here. Additional plugins can be installed in the Plugin Manager once the initial setup is complete. See the Wiki for more information.
Organization and Administration (2/3)
Dashboard View
10
Customizable dashboard that can present various views of job information.
Folders
This plugin allows users to create "folders" to organize jobs. Users can define custom taxonomies (like by project type, organization type etc). Folders are nestable and you can define views within folders. Maintained by CloudBees, Inc.
OWASP Markup Formatter
9
Uses the OWASP Java HTML Sanitizer to allow safe-seeming HTML markup to be entered in project descriptions and the like.
Build Features (4/10)
build-name-setter
11
This plug-in sets the display name of a build to something other than #1, #2, #3, ...
Build Timeout
7
This plugin allows builds to be automatically terminated after the specified amount of time has elapsed.
By slecting above, you can see plugins which are for that selected feature
Jenkins 2.150.2
Back
Install

Not secure | 3.83.106.28:8080/jenkins/

Getting Started

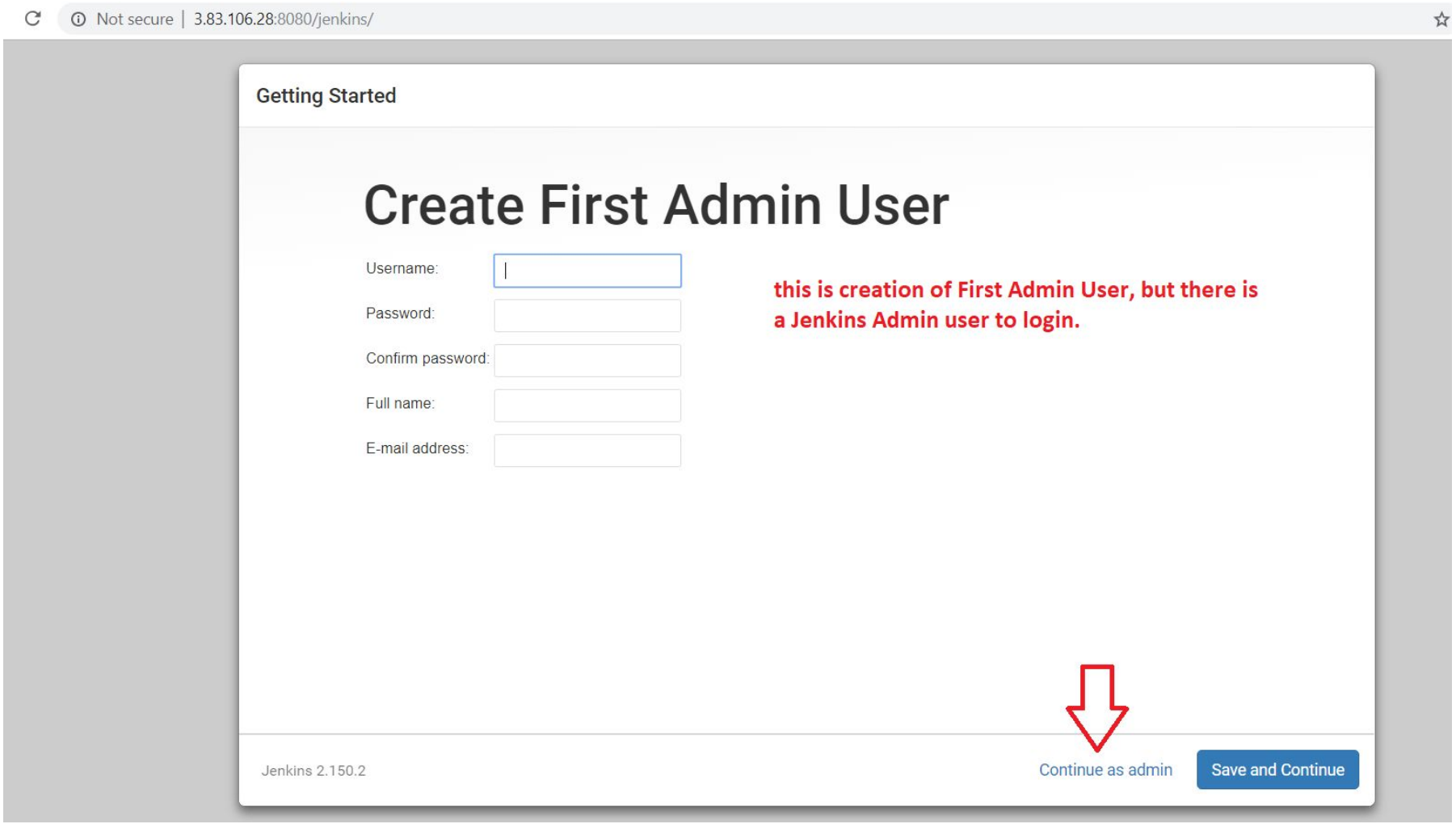
Process of plugins installations

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

```
** Pipeline: Stage Step
** Pipeline: Job
** Pipeline Graph Analysis
** Pipeline: REST API
** JavaScript GUI Lib: Handlebars bundle
** JavaScript GUI Lib: Moment.js bundle
Pipeline: Stage View
** Pipeline: Build Step
** Pipeline: Model API
** Pipeline: Declarative Extension Points API
** Apache HttpComponents Client 4.x API
** JSch dependency
** Git client
** GIT server
** Pipeline: Shared Groovy Libraries
** Display URL API
Mailer
** - required dependency
```

Jenkins 2.150.2


Not secure | 3.83.106.28:8080/jenkins/
Getting Started
Create First Admin User
Username:
Password:
Confirm password:
Full name:
E-mail address:
this is creation of First Admin User, but there is a Jenkins Admin user to login.
Jenkins 2.150.2
Continue as admin
Save and Continue

Not secure | 3.83.106.28:8080/jenkins/

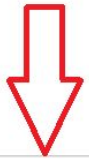
Getting Started

Instance Configuration

Jenkins URL: http://3.83.106.28:8080/jenkins/

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 2.150.2

Not now | Save and Finish

Not secure | 3.83.106.28:8080/jenkins/

Getting Started

Jenkins is ready!

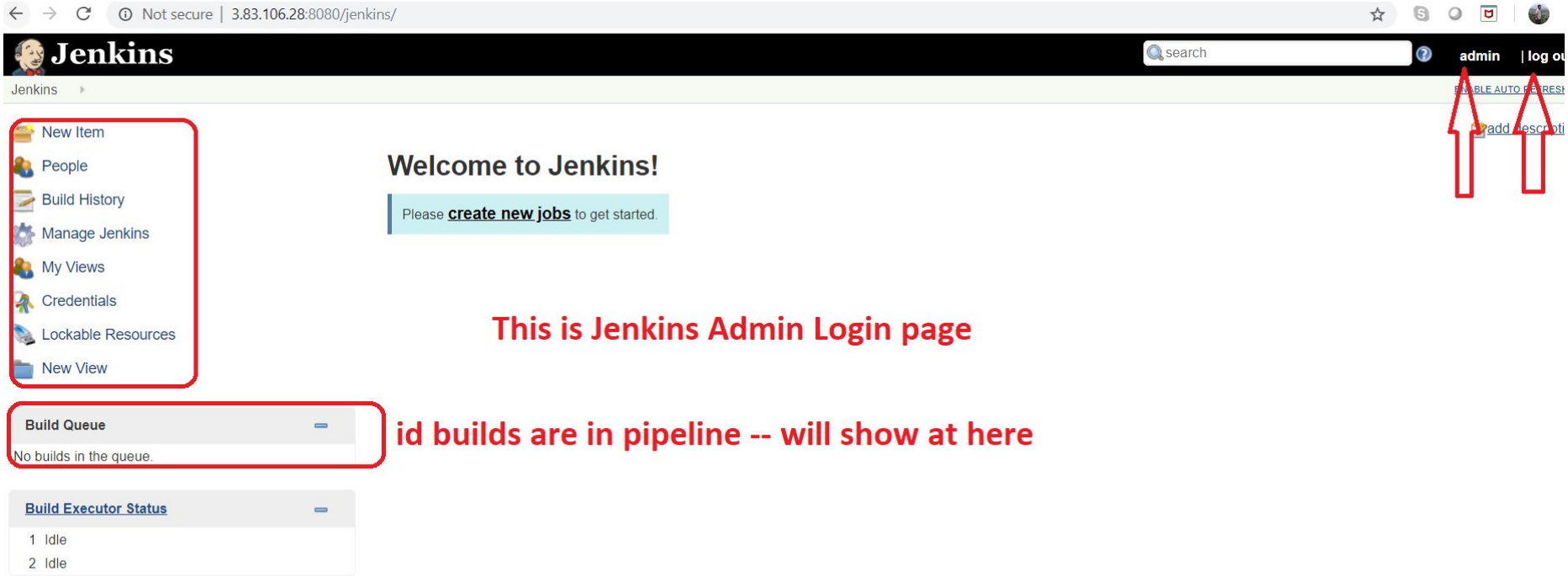
You have skipped the **setup of an admin user**.

To log in, use the username: "admin" and the administrator password you used to access the setup wizard.

Your Jenkins setup is complete.

Start using Jenkins

Jenkins 2.150.2


Not secure | 3.83.106.28:8080/jenkins/
Jenkins
search
admin
| log ou
Jenkins
ENABLE AUTO REFRESH
add Descripti
New Item
People
Build History
Manage Jenkins
My Views
Credentials
Lockable Resources
New View
Welcome to Jenkins!
Please create new jobs to get started.
This is Jenkins Admin Login page
Build Queue
No builds in the queue.
id builds are in pipeline -- will show at here
Build Executor Status
1 Idle
2 Idle

2. How to Find Home Directory of Jenkins

1. By Default, while Jenkins installation, a dir will be created in login user home dir
2. This jenkins home dir contains -- All Configurations, Plugins, logs, jobs & etc

```
root@ip-172-31-91-116:~/.jenkins# ls
config.xml                                    nodeMonitors.xml
hudson.model.UpdateCenter.xml                 nodes
hudson.plugins.git.GitTool.xml                plugins
identity.key.enc                              secret.key
jenkins.CLI.xml                               secret.key.not-so-secret
jenkins.install.InstallUtil.lastExecVersion   secrets
jenkins.install.UpgradeWizard.state           updates
jenkins.model.JenkinsLocationConfiguration.xml userContent
jenkins.telemetry.Correlator.xml              users
jobs                                          workflow-libs
logs
root@ip-172-31-91-116:~/.jenkins#
```

3. Login to Jenkins ---> Manage Jenkins ---> on top, you can see the home dir