

Jenkins & CI

By Chidanand

Definition & Usage

- What is Jenkins?
- Where it used & why?

Definition

Jenkins is a self contained open source automation server which can be used for automation of all sorts of software tasks like building, testing & deploying software

SDLC Context

- One of the important components of Continuous Integration & Deployment process
- Continuous Integration implements continuous processes of applying Quality Control – Small pieces of efforts applied frequently
- Continuous Integration emerged in the Extreme Programming (XP) community, and XP advocates Martin Fowler and Kent Beck first wrote about continuous integration circa 1999

Why Continuous Integration

- Integration is hard, effort increase exponentially with
 - Number of components
 - Number of bugs
 - Time since last integration

Benefits of Continuous Integration

- Project Management
 - Detect system development problems earlier
 - Reduce risks of cost, schedule, and budget
- Code Quality
 - Measurable and visible code quality
 - Continuous automatic regression unit test

Continuous Integration – Best practice

- Single Source Repository.
- Automate the Build and Test
- Everyone Commits Every Day
- Keep the Build Fast
- Everyone can see what's happening
- Automate Deployment (Optional)

.....Most importantly improve Automated Test % Coverage....or if possible, choose Test Driven Development

Jenkins/Hudson History

Hudson

- Work started in the summer of 2004 at Sun by *Kohsuke Kawaguchi*.
- First release: February 2005.
- In May, 2008, Hudson won the Duke's Choice Award in the Developer Solutions category.
- In 2010, Sun is acquired by Oracle... discussions, negotiations.
- Kohsuke Kawaguchi leaves Oracle.

Jenkins/Hudson History

Jenkins

- In 2010, majority of Hudson's community votes for forking and name changing.
- Initial release of Jenkins: 2nd of February, 2011.
- Oracle still supports and develops Hudson.
- Around 2010-2011, CloudBees is created and declared Jenkins as its backbone product.

Jenkins features (courtesy plugins)

- Trigger a build
- Get source code from repository
- Automatically build and test
- Generate report & notify
- Deploy
- Distributed build

Jenkins Plugins – extended features

- Build triggers
- Source code management
- Build tools
- Build wrappers
- Build notifiers
- Build reports
- Artifact uploaders
- UI plugins
- Code coverage
- Authentication and user management

.....And lots more

Jenkins Pre-Req

- Pre-requisites to install Jenkins
 - JDK 1.7 or above
 - 2 GB RAM (recommended)
 - Set PATH properly
- Can be installed as a stand-alone Web Application or can be run within a web server
- Please set up dependent ENV/PATH variables
 - JAVA_HOME : C:\Program Files\Java\jdk1.8.0_162
 - M2_HOME : C:\apache-maven-3.5.3
 - Git Bash : Append to System PATH variable : C:\Program Files\Git\cmd

JENKINS_HOME

- Root/Parent folder where Jenkins stores all its files, jobs & workspace
- C:\Program Files (x86)\Jenkins

Or

- C:\User_Home_DIR\.jenkins
- Modify JENKINS_HOME environment variable before you launch JENKINS installation for the first time

Types of Installation

- Windows or Linux/Unix Services
- Generic War File
- Installed on any Java based Web Servers –
Apache Tomcat

Downloading and Installing

- <https://jenkins.io/download/>
- Better to choose a LTS version
- You can choose WAR
- Or Windows Installer

Download Jenkins 2.46.2 for:

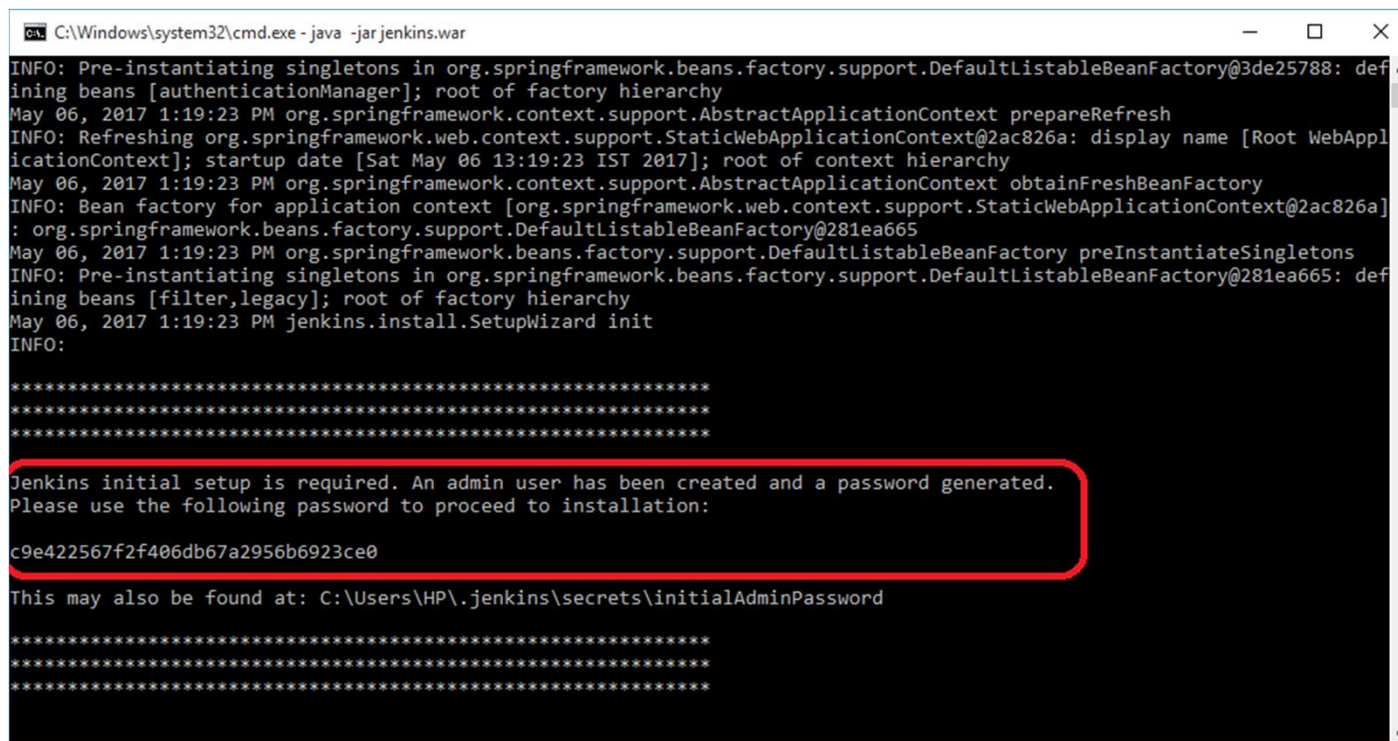
Docker
FreeBSD
Gentoo
Mac OS X
OpenBSD
openSUSE
Red Hat/Fedora/CentOS
Ubuntu/Debian
Windows
Generic Java package (.war)

Starting Jenkins using the WAR file

- Open the command prompt.
- From the command prompt, browse to the directory where the jenkins.war file is present.
- Run the following command
 - `java -jar jenkins.war`
- Winstone is fast, reliable servlet container which runs Jenkins
- Once the extraction and initialization is complete, you can see a console message which says “Jenkins is fully up and running”

Initial Set up

- The very first time you run Jenkins, Initial User 'admin' will be created and a random password is generated.
- Please make a note of the password



```
C:\Windows\system32\cmd.exe - java -jar jenkins.war
INFO: Pre-instantiating singletons in org.springframework.beans.factory.support.DefaultListableBeanFactory@3de25788: defining beans [authenticationManager]; root of factory hierarchy
May 06, 2017 1:19:23 PM org.springframework.context.support.AbstractApplicationContext prepareRefresh
INFO: Refreshing org.springframework.web.context.support.StaticWebApplicationContext@2ac826a: display name [Root WebApplicationContext]; startup date [Sat May 06 13:19:23 IST 2017]; root of context hierarchy
May 06, 2017 1:19:23 PM org.springframework.context.support.AbstractApplicationContext obtainFreshBeanFactory
INFO: Bean factory for application context [org.springframework.web.context.support.StaticWebApplicationContext@2ac826a]: org.springframework.beans.factory.support.DefaultListableBeanFactory@281ea665
May 06, 2017 1:19:23 PM org.springframework.beans.factory.support.DefaultListableBeanFactory preInstantiateSingletons
INFO: Pre-instantiating singletons in org.springframework.beans.factory.support.DefaultListableBeanFactory@281ea665: defining beans [filter,legacy]; root of factory hierarchy
May 06, 2017 1:19:23 PM jenkins.install.SetupWizard init
INFO:

*****
*****
*****

Jenkins initial setup is required. An admin user has been created and a password generated.
Please use the following password to proceed to installation:

c9e422567f2f406db67a2956b6923ce0

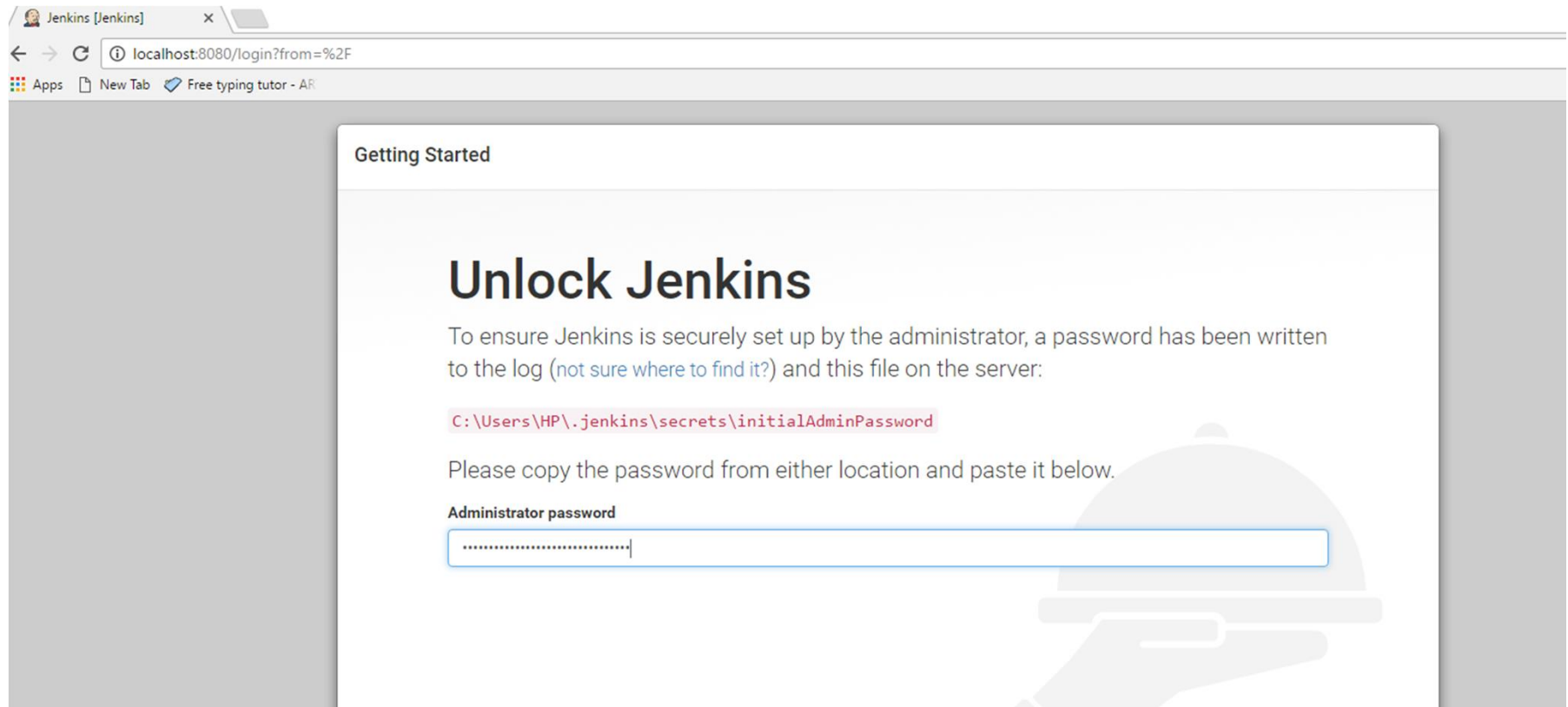
This may also be found at: C:\Users\HP\.jenkins\secrets\initialAdminPassword

*****
*****
*****
```

Access Jenkins

- Open a web browser and type in the URL <http://localhost:8080>
- Since we are accessing Jenkins for the first time, you are greeted with the Unlock Jenkins Screen
- Please copy and paste the password from console – previous step

Access Jenkins



Selecting Plugins

- You can select plugins to install

Getting Started

Customize Jenkins

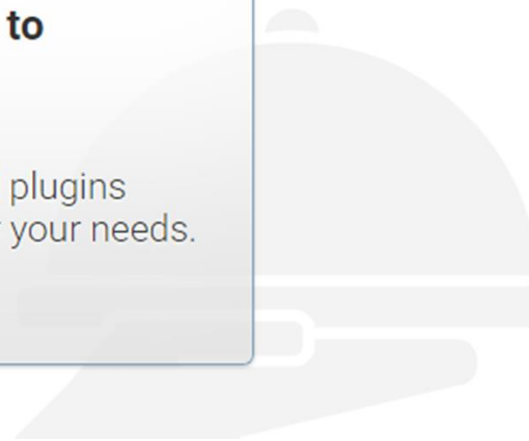
Plugins extend Jenkins with additional features to support many different needs.

Install suggested plugins

Install plugins the Jenkins community finds most useful.

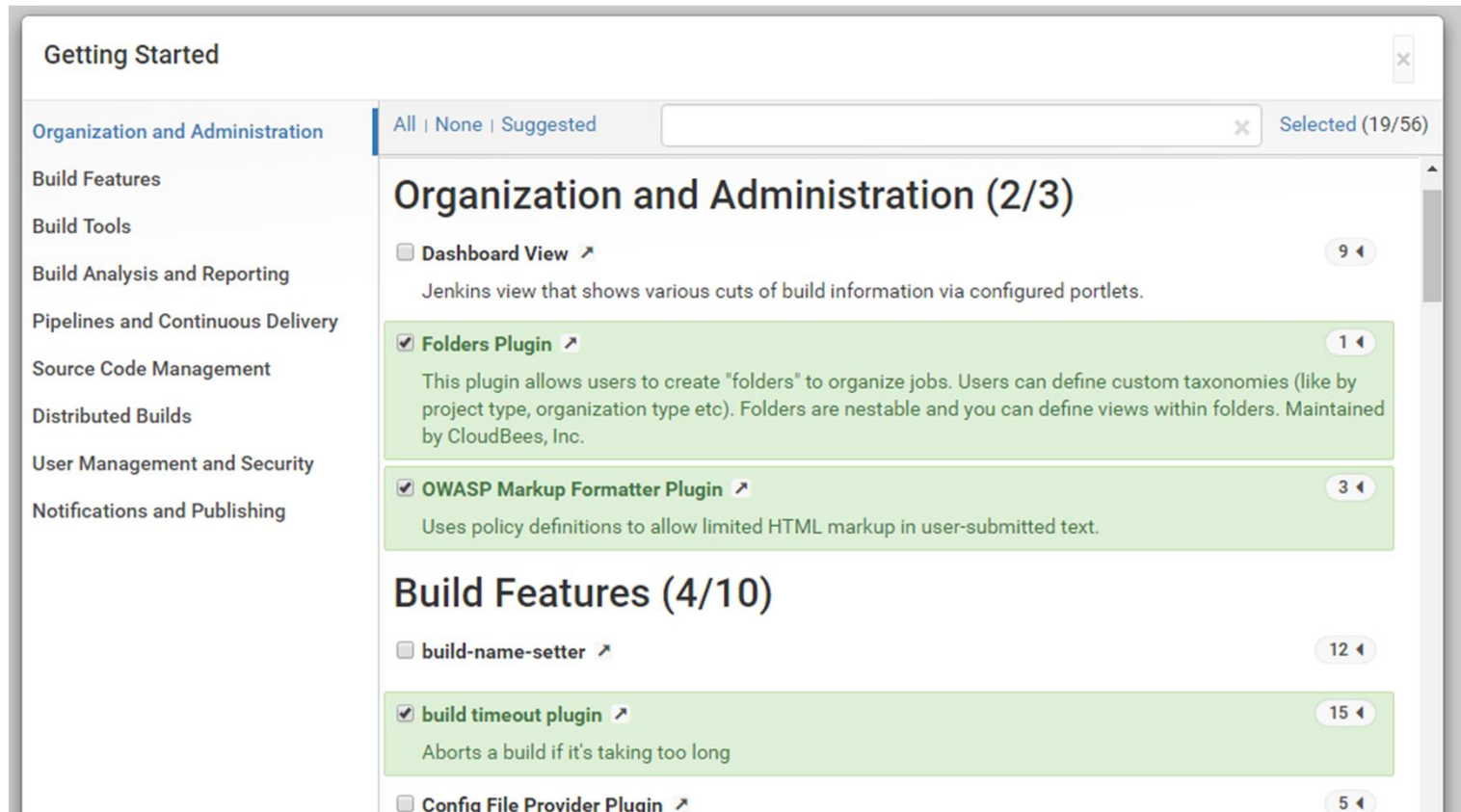
Select plugins to install

Select and install plugins most suitable for your needs.



Browse the list of Plugins and choose

- Browse all the sections and check what you are about to install






















Contd..

- Click Install and wait for installation of plugins to complete

Getting Started

Getting Started

 Folders Plugin	 OWASP Markup Formatter Plugin	 build timeout plugin	 Credentials Binding Plugin	** bouncycastle API Plugin
 Timestamper	 Workspace Cleanup Plugin	 Ant Plugin	 Gradle Plugin	
 Pipeline	 GitHub Organization Folder Plugin	 Pipeline: Stage View Plugin	 Git plugin	
 Subversion Plug-in	 SSH Slaves plugin	 Matrix Authorization Strategy Plugin	 PAM Authentication plugin	
 LDAP Plugin	 Email Extension Plugin	 Mailer Plugin		

** - required dependency

Trouble Shooting

- Some times Jenkins fails to install certain plugins and the process just hangs. This is a known issue even in LTS versions
- Delete the .jenkins folder from Home Directory
- Retry clean installation by Unchecking the Failed Plugins
- For example if Pipeline and Git Folder Organization plugins failed, then uncheck them and install plugins
- Sometimes you get a continue option to install failed plugins later

Initial User Creation

- After installation of plugins, you need to create an Admin User

Getting Started

Create First Admin User

Username:

Password:

Confirm password:

Full name:

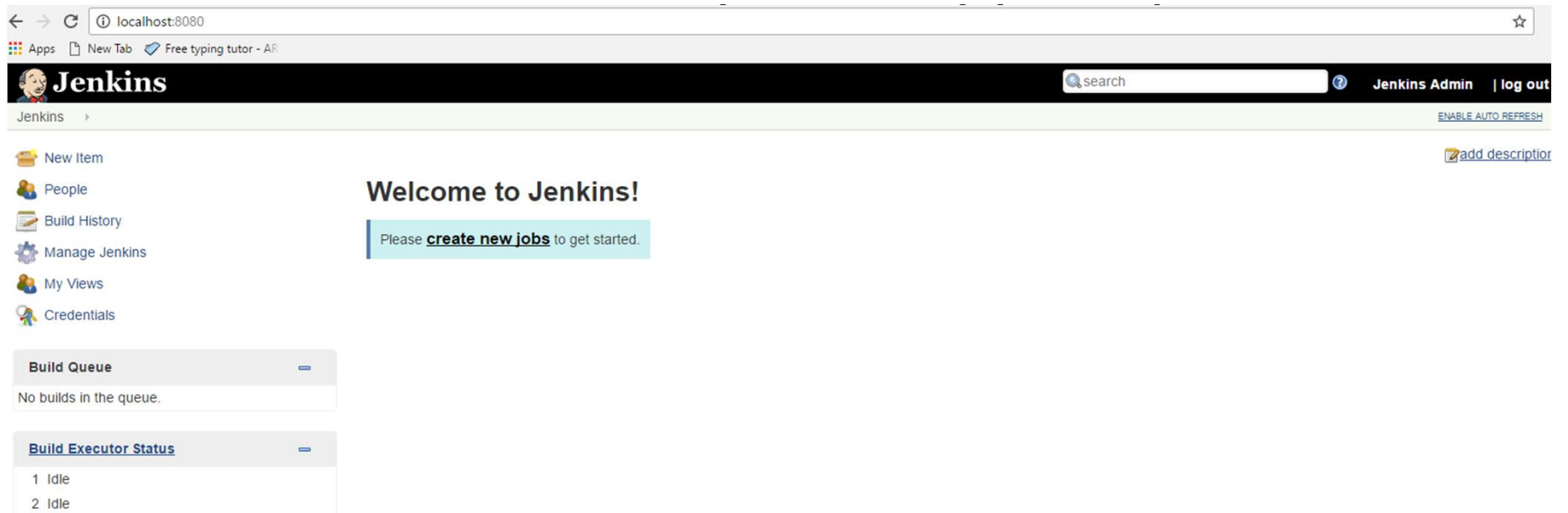
E-mail address:

Jenkins is ready!

Your Jenkins setup is complete.

[Start using Jenkins](#)

Jenkins Dashboard



The screenshot shows the Jenkins Dashboard in a web browser. The browser's address bar displays 'localhost:8080'. The Jenkins header includes the logo, a search bar, and links for 'Jenkins Admin' and 'log out'. A sidebar on the left contains navigation links: 'New Item', 'People', 'Build History', 'Manage Jenkins', 'My Views', and 'Credentials'. The main content area features a 'Welcome to Jenkins!' message with a prompt to 'create new jobs'. Below this, there are two expandable sections: 'Build Queue' (showing 'No builds in the queue.') and 'Build Executor Status' (showing two idle executors).

localhost:8080

Jenkins

search

Jenkins Admin | log out

Jenkins

ENABLE AUTO REFRESH

add descriptor

New Item

People

Build History

Manage Jenkins

My Views

Credentials

Build Queue

No builds in the queue.

Build Executor Status

1 Idle

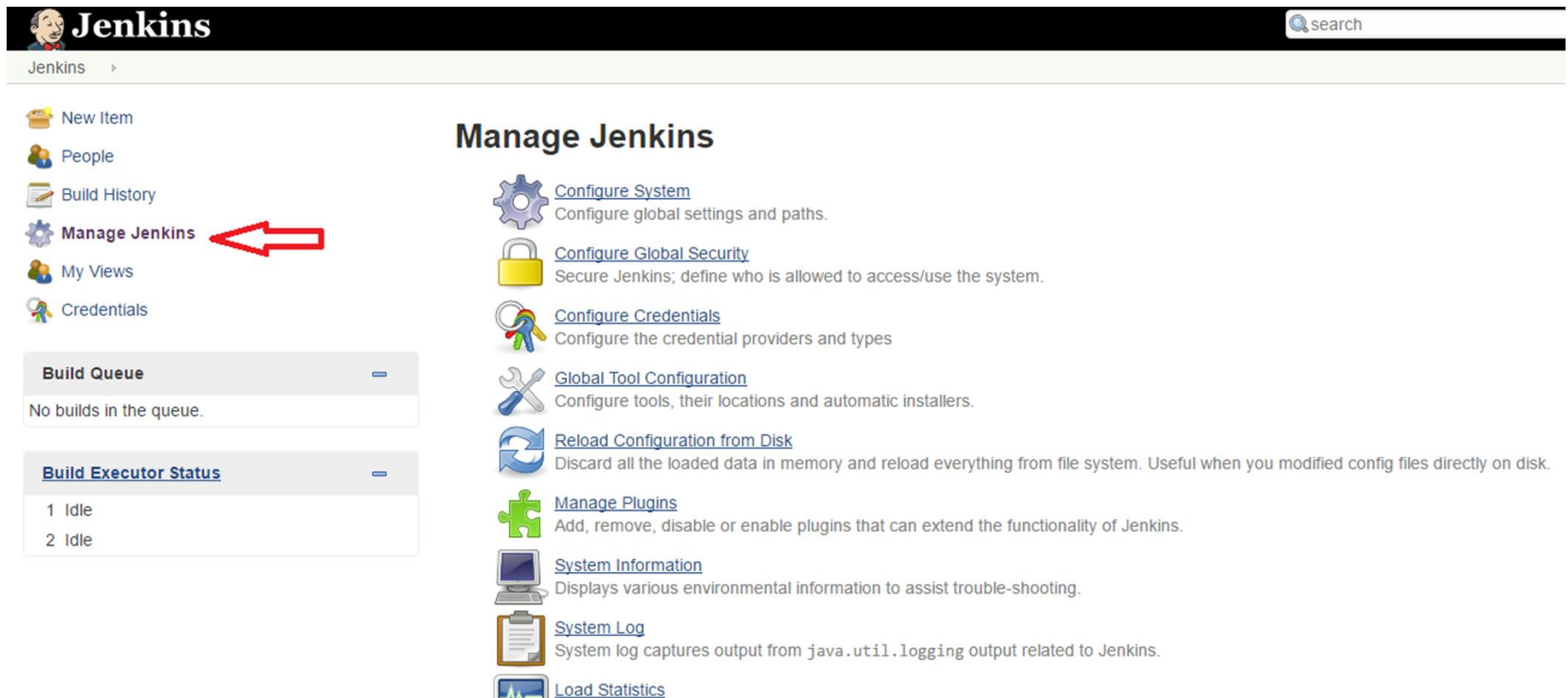
2 Idle

Welcome to Jenkins!










Please [create new jobs](#) to get started.

Managing Jenkins

- You can click on Manage Jenkins Link in Left



The screenshot displays the Jenkins web interface. At the top, there is a black header with the Jenkins logo and a search bar. Below the header, a light green navigation bar contains the text 'Jenkins' followed by a right-pointing arrow. The left sidebar features a list of links: 'New Item', 'People', 'Build History', 'Manage Jenkins', 'My Views', and 'Credentials'. A red arrow points to the 'Manage Jenkins' link. Below this list are two expandable sections: 'Build Queue' (showing 'No builds in the queue.') and 'Build Executor Status' (showing two 'Idle' executors). The main content area is titled 'Manage Jenkins' and contains a list of configuration links, each with an icon and a description:

-  [Configure System](#)
Configure global settings and paths.
-  [Configure Global Security](#)
Secure Jenkins; define who is allowed to access/use the system.
-  [Configure Credentials](#)
Configure the credential providers and types
-  [Global Tool Configuration](#)
Configure tools, their locations and automatic installers.
-  [Reload Configuration from Disk](#)
Discard all the loaded data in memory and reload everything from file system. Useful when you modified config files directly on disk.
-  [Manage Plugins](#)
Add, remove, disable or enable plugins that can extend the functionality of Jenkins.
-  [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 Statistics](#)

Configure Jenkins – Executors contd

- # of executors -- refers to the total number of concurrent job executions that can take place on the Jenkins machine
- recommendation is to keep this number the same as the number of CPU on the machines for better performance
- Preferred setting is 2

Configure Jenkins – Host Name or IP

- By default, the Jenkins URL points to localhost.
- If you have a domain name setup for your machine, set this to the domain name

Jenkins Location

Jenkins URL

http://localhost:8080/



⚠ Please set a valid host name, instead of localhost

Optionally specify the HTTP address of the Jenkins installation, such as `http://yourhost.yourdomain/jenkins/`. This value is used to let Jenkins know how to refer to itself, ie. to display images or to create links in emails.

This is necessary because Jenkins cannot reliably detect such a URL from within itself.



Other Configurations

- Manage Plugins - install a wide variety of third-party plugins
- System Information - displays a list of all the current Java system properties and system environment variables
- System Log - convenient way to view the Jenkins log files in real time, mainly used for trouble shooting
- Load Statistics -- displays graphical data on how busy the Jenkins instance is in terms of the number of concurrent builds and the length of the build queue which gives an idea of how long your builds need to wait before being executed. These statistics can give a good idea of whether extra capacity or extra build nodes is required from an infrastructure perspective
- Manage nodes - A build node is another machine that Jenkins can use to execute its builds. You can set up build nodes.
- Prepare for Shutdown - To shut down Jenkins cleanly, you can use the Prepare for Shutdown link, which prevents any new builds from being started.