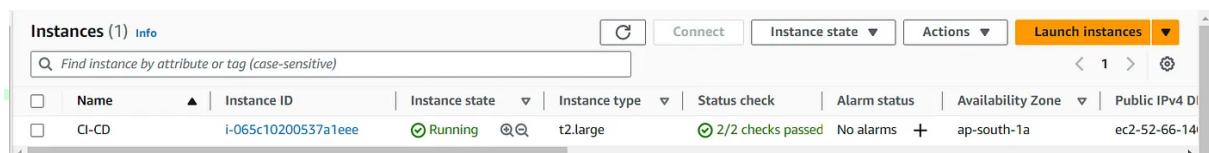


# DEPLOY THE REACTJS APP IN KUBERNETES WITH DEVSECOPS CICD PIPELINE

## STEP1:Launch an Ubuntu(22.04) T2 Large Instance



Step 2 —

### 2A - Install Jenkins

<https://github.com/AWS-AZURE-Bootcamp5/tools-setup-project1/blob/main/jenkins.sh>

Go to AWS EC2 Security Group and open Inbound Port 8080 and in server get the password

```
<EC2 Public IP Address:8080>  
sudo cat /var/lib/jenkins/secrets/initialAdminPassword
```

Upcoming New 100% Job Ready DevOps/SRE Course - <https://heydevops.in/batch6>  
Previous Batches Highest Package - 36+ LPA [ Testimonials in website ]  
Checkout Students Placements - <https://heydevops.in/success-stories>  
Checkout 14+ Real time projects - <https://heydevops.in/batch6>

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:

`/var/lib/jenkins/secrets/initialAdminPassword`

Please copy the password from either location and paste it below.

Administrator password

[Continue](#)

## Unlock 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 2.414.1

Create a user click on save and continue.

**Upcoming New 100% Job Ready DevOps/SRE Course - <https://heydevops.in/batch6>**  
**Previous Batches Highest Package - 36+ LPA [ Testimonials in website ]**  
**Checkout Students Placements - <https://heydevops.in/success-stories>**  
**Checkout 14+ Real time projects - <https://heydevops.in/batch6>**

## 2B — Install Docker and setup Sonarqube

<https://github.com/AWS-AZURE-Bootcamp5/tools-setup-project1/blob/main/docker.sh>

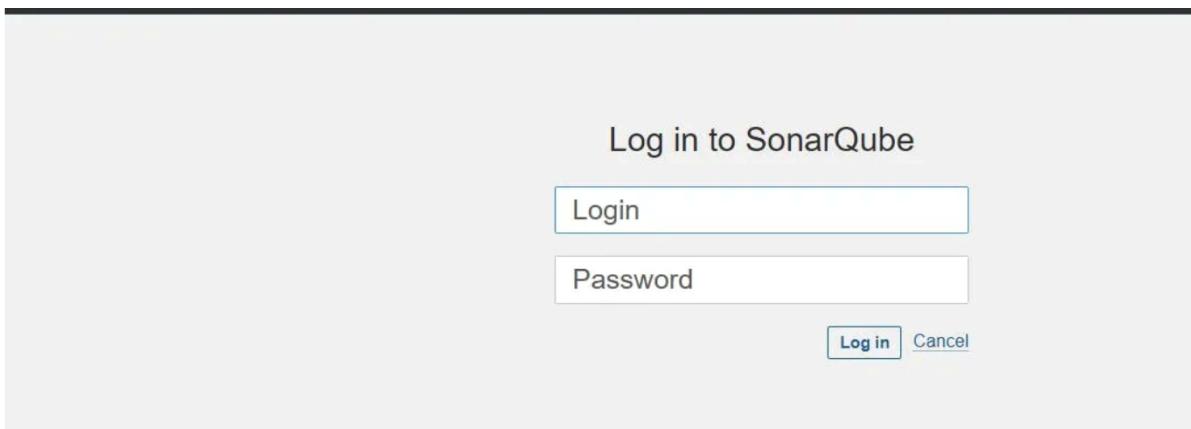
After the docker installation, we created a Sonarqube container (Remember to add 9000 ports in the security group).

`docker run -d --name sonar -p 9000:9000 sonarqube:lts-community`

```
ubuntu@ip-172-31-42-253:~$ sudo chmod 777 /var/run/docker.sock
ubuntu@ip-172-31-42-253:~$ docker run -d --name sonar -p 9000:9000 sonarqube:lts-community
Unable to find image 'sonarqube:lts-community' locally
lts-community: Pulling from library/sonarqube
44ba2882f8eb: Pull complete
2cabe577fa36: Pull complete
c204813b496a: Pull complete
b1f7b17ee74fb: Pull complete
38617faac714: Pull complete
706f20ff58f5e: Pull complete
65a29568c257: Pull complete
Digest: sha256:a1a18fbab960d6c3d4ea8b4455a5a6560654511c88a6816f1603f764d5dcc77c
Status: Downloaded newer image for sonarqube:lts-community
4b60c96bf9ad3d62289436af7f752fdb04993092d0ca30b5e2f2e32301b50139
ubuntu@ip-172-31-42-253:~$ docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED            STATUS              PORTS              NAMES
4b60c96bf9ad        sonarqube:lts-community   "/opt/sonarqube/dock..."   9 seconds ago      Up 5 seconds          0.0.0.0:9000->9000/tcp, :::9000->9000/tcp   sonar
ubuntu@ip-172-31-42-253:~$
```

Now our Sonarqube is up and running

Upcoming New 100% Job Ready DevOps/SRE Course - <https://heydevops.in/batch6>  
 Previous Batches Highest Package - 36+ LPA [ Testimonials in website ]  
 Checkout Students Placements - <https://heydevops.in/success-stories>  
 Checkout 14+ Real time projects - <https://heydevops.in/batch6>



**Enter username and password, click on login and change password**

**username admin / password admin**

**Update New password, This is Sonar Dashboard.**

The screenshot shows the SonarQube dashboard with a navigation bar at the top. Below the navigation, there's a section titled "How do you want to create your project?" with a sub-instruction: "Do you want to benefit from all of SonarQube's features (like repository import and Pull Request decoration)? Create your project from your favorite DevOps platform. First, you need to set up a DevOps platform configuration." Five cards are displayed below, each representing a different platform:

- From Azure DevOps**: Shows the Azure DevOps logo and a "Set up global configuration" link.
- From Bitbucket Server**: Shows the Bitbucket Server logo and a "Set up global configuration" link.
- From Bitbucket Cloud**: Shows the Bitbucket Cloud logo and a "Set up global configuration" link.
- From GitHub**: Shows the GitHub logo and a "Set up global configuration" link.
- From GitLab**: Shows the GitLab logo and a "Set up global configuration" link.

## 2C — Install Trivy [ IAC ]

<https://github.com/AWS-AZURE-Bootcamp5/tools-setup-project1/blob/main/trivy.sh>

**Upcoming New 100% Job Ready DevOps/SRE Course - <https://heydevops.in/batch6>**  
**Previous Batches Highest Package - 36+ LPA [ Testimonials in website ]**  
**Checkout Students Placements - <https://heydevops.in/success-stories>**  
**Checkout 14+ Real time projects - <https://heydevops.in/batch6>**

## **Step 3 — Install Plugins like JDK, Sonarqube Scanner, NodeJS, OWASP Dependency Check**

### **3A — Install Plugin**

Go to Manage Jenkins → Plugins → Available Plugins →

Install below plugins

1 → Eclipse Temurin Installer (Install without restart)

2 → Sonarqube Scanner (Install without restart)

3 → NodeJS Plugin (Install Without restart)

### **3B — Configure Java and Nodejs in Global Tool Configuration**

Goto Manage Jenkins → Tools → Install JDK(17) and NodeJs(16)→ Click on Apply and Save

**Upcoming New 100% Job Ready DevOps/SRE Course - <https://heydevops.in/batch6>**

**Previous Batches Highest Package - 36+ LPA [ Testimonials in website ]**

**Checkout Students Placements - <https://heydevops.in/success-stories>**

**Checkout 14+ Real time projects - <https://heydevops.in/batch6>**

The first screenshot shows the 'JDK Installations' configuration page. It has a 'Name' field set to 'jdk17' and an 'Install automatically' checkbox checked. Under 'Install from adoptium.net', the 'Version' dropdown is set to 'jdk-17.0.8.1+1'. A 'Add Installer' button is visible at the bottom.

The second screenshot shows the 'NodeJS' configuration page. It has a 'Name' field set to 'node16' and an 'Install automatically' checkbox checked. Under 'Install from nodejs.org', the 'Version' dropdown is set to 'NodeJS 16.2.0'. There is a note about forcing the 32-bit architecture if available. A 'Global npm packages to install' section is also present.

## 3C — Create a Job

create a job as Devsecops\_demo Name, select pipeline and click ok.

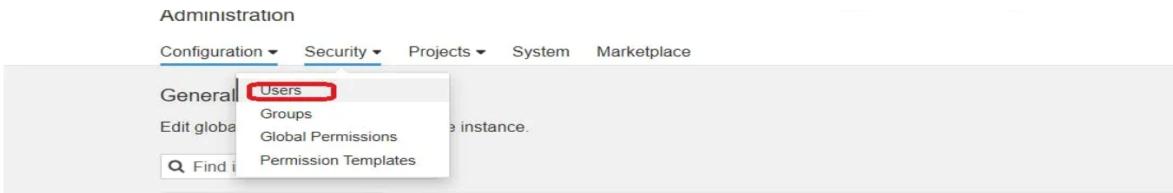
## Step 4 — Configure Sonar Server in Manage Jenkins

Grab the Public IP Address of your EC2 Instance, Sonarqube works on Port 9000, so <Public IP>:9000.

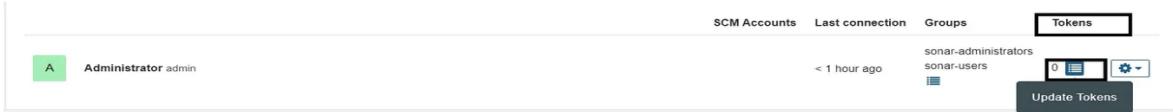
Goto your Sonarqube Server.

Click on Administration → Security → Users → Click on Tokens and Update Token → Give it a name → and click on Generate Token

**Upcoming New 100% Job Ready DevOps/SRE Course - <https://heydevops.in/batch6>**  
**Previous Batches Highest Package - 36+ LPA [ Testimonials in website ]**  
**Checkout Students Placements - <https://heydevops.in/success-stories>**  
**Checkout 14+ Real time projects - <https://heydevops.in/batch6>**



click on update Token



Create a token with a name and generate

The screenshot shows the 'Tokens of Administrator' page. It includes a 'Generate Tokens' form and a table of tokens:

**Generate Tokens**

Name	Expires in
Enter Token Name	30 days
<small>New token "Jenkins" has been created. Make sure you copy it now, you won't be able to see it again!</small>	
Copy	qu_21d162984c1c72c78b39665f96480185c99dc2f9

**Tokens**

Name	Type	Project	Last use	Created	Expiration	
Jenkins	User		Never	September 8, 2023	October 8, 2023	

copy Token

Goto Jenkins Dashboard → Manage Jenkins → Credentials → Add Secret Text. It should look like this

**Upcoming New 100% Job Ready DevOps/SRE Course - <https://heydevops.in/batch6>**  
**Previous Batches Highest Package - 36+ LPA [ Testimonials in website ]**  
**Checkout Students Placements - <https://heydevops.in/success-stories>**  
**Checkout 14+ Real time projects - <https://heydevops.in/batch6>**

Dashboard > Manage Jenkins > Credentials > System > Global credentials (unrestricted) >

### New credentials

Kind	Secret text
Scope ?	Global (Jenkins, nodes, items, all child items, etc)
Secret	<b>POST THE TOKEN HERE</b>
ID ?	Sonar-token
Description ?	Sonar-token
<b>Create</b>	

Credentials that should be available irrespective of domain specification to requirements matching.

ID	Name	Kind	Description	Action
 Sonar-token	sonar	Secret text	sonar	

Now, go to Dashboard → Manage Jenkins → System and Add like the below image.

**Upcoming New 100% Job Ready DevOps/SRE Course - <https://heydevops.in/batch6>**  
**Previous Batches Highest Package - 36+ LPA [ Testimonials in website ]**  
**Checkout Students Placements - <https://heydevops.in/success-stories>**  
**Checkout 14+ Real time projects - <https://heydevops.in/batch6>**

The screenshot shows the Jenkins 'System' configuration page under 'Manage Jenkins'. The 'SonarQube servers' section is active. It includes an optional checkbox for enabling environment variables, which is unchecked. Below it is a 'SonarQube installations' section with a table. The first row in the table has a red border and contains fields for 'Name' (set to 'sonar-server'), 'Server URL' (set to 'http://13.232.17.191:9000'), and 'Server authentication token' (set to 'Sonar-token'). A dropdown menu labeled 'Add' is visible. At the bottom of the table are 'Save' and 'Apply' buttons.

Click on Apply and Save

**The Configure System option** is used in Jenkins to configure different server

**Global Tool Configuration** is used to configure different tools that we install using Plugins

We will install a sonar scanner in the tools.

**Upcoming New 100% Job Ready DevOps/SRE Course - <https://heydevops.in/batch6>**  
**Previous Batches Highest Package - 36+ LPA [ Testimonials in website ]**  
**Checkout Students Placements - <https://heydevops.in/success-stories>**  
**Checkout 14+ Real time projects - <https://heydevops.in/batch6>**

SonarQube Scanner installations

Add SonarQube Scanner

≡ SonarQube Scanner

Name

sonar-scanner

Install automatically ?

≡ Install from Maven Central

Version

SonarQube Scanner 5.0.1.3006

Add Installer ▾

Add SonarQube Scanner

Save

Apply

In the Sonarqube Dashboard add a quality gate also

Administration → Configuration → Webhooks



**Upcoming New 100% Job Ready DevOps/SRE Course - <https://heydevops.in/batch6>**

**Previous Batches Highest Package - 36+ LPA [ Testimonials in website ]**

**Checkout Students Placements - <https://heydevops.in/success-stories>**

**Checkout 14+ Real time projects - <https://heydevops.in/batch6>**

The screenshot shows the SonarQube Administration page with the 'Administration' tab selected. Under 'Configuration', the 'Webhooks' option is highlighted with a red box. A search bar at the top right contains the placeholder 'Search for projects...'. On the right, there is a green button labeled 'Create User'. Below the search bar, a table lists users with columns for SCM Accounts, Last connection, Groups, and Tokens. One user, 'Administrator admin', is listed with a green status icon. The table footer indicates '1 of 1 shown'.

Click on Create

The screenshot shows the SonarQube Administration page with the 'Administration' tab selected. Under 'Configuration', the 'Webhooks' option is selected. A blue box highlights the 'Create' button on the right. The page displays a message about webhooks and their purpose. It also shows a note stating 'No webhook defined.'

Add details

The screenshot shows a modal dialog box containing the following webhook URL:

```
#in url section of quality gate
<http://jenkins-public-ip:8080>/sonarqube-webhook/
```

**Upcoming New 100% Job Ready DevOps/SRE Course - <https://heydevops.in/batch6>**  
**Previous Batches Highest Package - 36+ LPA [ Testimonials in website ]**  
**Checkout Students Placements - <https://heydevops.in/success-stories>**  
**Checkout 14+ Real time projects - <https://heydevops.in/batch6>**

## Create Webhook

All fields marked with \* are required

**Name \***

Jenkins



**URL \***

http://54.176.129.1:8080/sonarqube-webhook/



Server endpoint that will receive the webhook payload, for example:  
"http://my\_server/foo". If HTTP Basic authentication is used, HTTPS is  
recommended to avoid man in the middle attacks. Example:  
"https://myLogin:myPassword@my\_server/foo"

**Secret**

[redacted]

If provided, secret will be used as the key to generate the HMAC hex  
(lowercase) digest value in the 'X-Sonar-Webhook-HMAC-SHA256'  
header.

**Create**

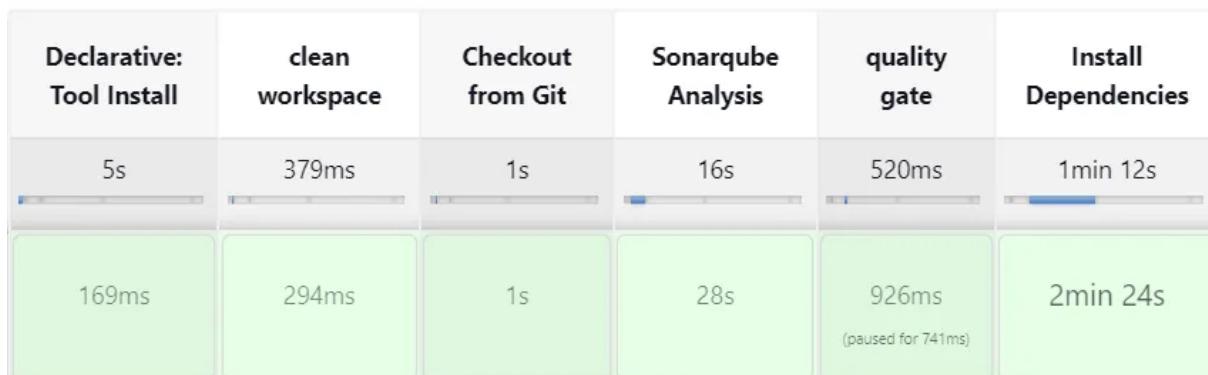
**Cancel**

Let's go to our Pipeline and add the script in our Pipeline Script.

<https://github.com/AWS-AZURE-Bootcamp5/Devsecops-Project1/blob/main/Jenkinsfile1>

Click on Build now, you will see the stage view like this

**Upcoming New 100% Job Ready DevOps/SRE Course - <https://heydevops.in/batch6>**  
**Previous Batches Highest Package - 36+ LPA [ Testimonials in website ]**  
**Checkout Students Placements - <https://heydevops.in/success-stories>**  
**Checkout 14+ Real time projects - <https://heydevops.in/batch6>**



## Step 5 — Install OWASP Dependency Check Plugins

Go to Dashboard → Manage Jenkins → Plugins → OWASP Dependency-Check. Click on it and install it without restart.

The screenshot shows the Jenkins Manage Jenkins > Plugins interface. On the left sidebar, there are tabs for Updates, Available plugins (which is selected), Installed plugins, Advanced settings, and Download progress. The main area is titled "Plugins" and contains a search bar with the placeholder "Search available plugins". A list of available plugins is shown, with "OWASP Dependency-Check 5.4.2" highlighted. This plugin is categorized under Security, DevOps, Build Tools, and Build Reports. It was released 8 days 17 hr ago. There is an "Install" button with a checkmark icon next to the plugin name.

First, we configured the Plugin and next, we had to configure the Tool

Go to Dashboard → Manage Jenkins → Tools →

**Upcoming New 100% Job Ready DevOps/SRE Course - <https://heydevops.in/batch6>**  
**Previous Batches Highest Package - 36+ LPA [ Testimonials in website ]**  
**Checkout Students Placements - <https://heydevops.in/success-stories>**  
**Checkout 14+ Real time projects - <https://heydevops.in/batch6>**

## Dependency-Check installations

[Add Dependency-Check](#)**Dependency-Check**

DP-Check

 Install automatically ?**Install from github.com**

## Version

dependency-check 6.5.1

[Add Installer ▾](#)

Now go configure → Pipeline and add OWASP and TRIVY stage to your pipeline and build.



You will see that in status, a graph will also be generated and Vulnerabilities.

**Upcoming New 100% Job Ready DevOps/SRE Course - <https://heydevops.in/batch6>**

**Previous Batches Highest Package - 36+ LPA [ Testimonials in website ]**

**Checkout Students Placements - <https://heydevops.in/success-stories>**

**Checkout 14+ Real time projects - <https://heydevops.in/batch6>**

## Dependency-Check Results

Severity Distribution			
13	39	13	
File Name	Vulnerability	Severity	Weakness
ansi-html@0.0.7	NVD CVE-2021-23424	High	NVD-CWE-noinfo
ansi-regex@4.1.0	NVD CVE-2021-3807	High	CWE-1333
async@2.6.3	NVD CVE-2021-43138	High	CWE-1321
browserslist@4.14.2	NVD CVE-2021-23364	Medium	CWE-1333
css-what@3.4.2	OSSINDEX CVE-2022-21222	High	CWE-1333
decode-uri-component@0.2.0	NVD CVE-2022-38778	Medium	CWE-20
decode-uri-component@0.2.0	NVD CVE-2022-38900	High	CWE-20
ejs@2.7.4	OSSINDEX CVE-2022-29078	High	CWE-94
eventsourcing@1.1.0	NVD CVE-2022-1650	Critical	CWE-212
express@4.17.1	OSSINDEX CVE-2022-24999	High	CWE-1321

## Step 6 — Docker Image Build and Push

We need to install the Docker tool in our system,

Goto Dashboard → Manage Plugins → Available plugins → Search for Docker and install these plugins

Docker

Docker Commons

Docker Pipeline

Docker API

Docker-build-step

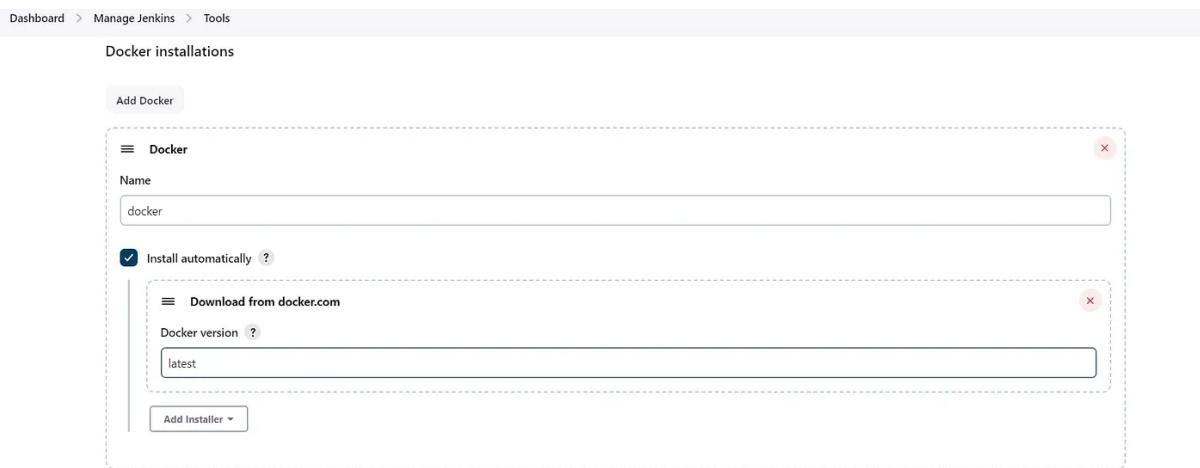
Upcoming New 100% Job Ready DevOps/SRE Course - <https://heydevops.in/batch6>

Previous Batches Highest Package - 36+ LPA [ Testimonials in website ]

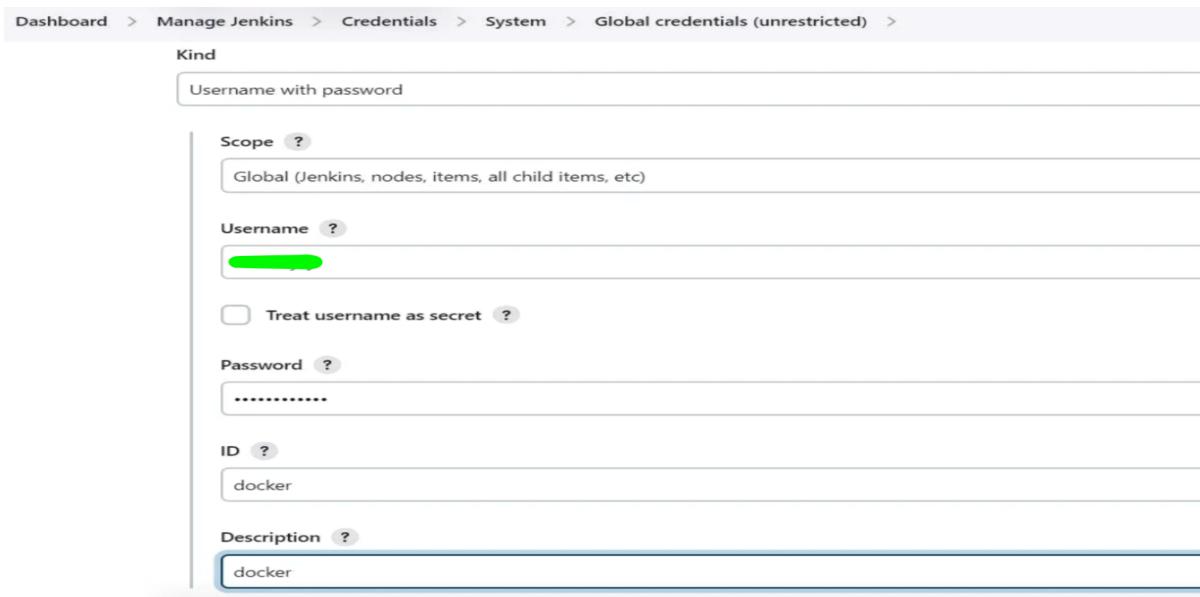
Checkout Students Placements - <https://heydevops.in/success-stories>

Checkout 14+ Real time projects - <https://heydevops.in/batch6>

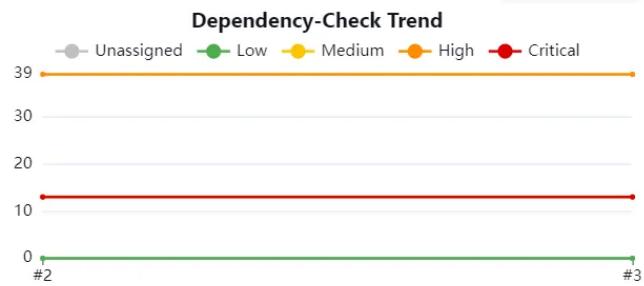
Now, goto Dashboard → Manage Jenkins → Tools →



## Add Docker Hub Username and Password under Global Credentials



**Upcoming New 100% Job Ready DevOps/SRE Course - <https://heydevops.in/batch6>**  
**Previous Batches Highest Package - 36+ LPA [ Testimonials in website ]**  
**Checkout Students Placements - <https://heydevops.in/success-stories>**  
**Checkout 14+ Real time projects - <https://heydevops.in/batch6>**



Declarative: Tool Install	clean workspace	Checkout from Git	Sonarqube Analysis	quality gate	Install Dependencies	OWASP FS SCAN	TRIVY FS SCAN	Docker Build & Push	TRIVY
3s	366ms	1s	19s	451ms	1min 20s	2min 1s	16s	3min 9s	4s
154ms	341ms	1s	25s	315ms	1min 36s	2min 31s	23s	3min 9s	4s

**Now Run the container to see if the game coming up or not by adding below stage**

Declarative: Tool Install	clean workspace	Checkout from Git	Sonarqube Analysis	quality gate	Install Dependencies	OWASP FS SCAN	TRIVY FS SCAN	Docker Build & Push	TRIVY	Deploy to container
144ms	284ms	1s	25s	410ms	1min 47s	2min 43s	23s	2min 7s	36s	789ms
146ms	251ms	1s	26s	305ms	1min 36s	2min 35s	23s	1min 50s	2min 8s	1s

<Jenkins-public-ip:3000>

**Upcoming New 100% Job Ready DevOps/SRE Course - <https://heydevops.in/batch6>**  
**Previous Batches Highest Package - 36+ LPA [ Testimonials in website ]**  
**Checkout Students Placements - <https://heydevops.in/success-stories>**  
**Checkout 14+ Real time projects - <https://heydevops.in/batch6>**

## **ASSIGNMENT < CREATE AND INTEGRATE THE k8 CLUSTER WITH CICD PIPELINE >**

### **Step 8 — Kubernetes Setup**

**Take-Two Ubuntu 20.04 instances one for k8s master and the other one for worker. T2.MEDIUM 15 GB**

**Install Kubectl on Jenkins machine also.**

<https://github.com/AWS-AZURE-Bootcamp5/tools-setup-project1/blob/main/kubectl.sh>

#### **Part 1 — — — — Master Node — — — — —**

Set the hostname for Master Server

**sudo hostnamectl set-hostname K8s-Master**

#### **Part 1.1 — — — — Worker Node — — — — —**

Set the hostname for Worker Server

**sudo hostnamectl set-hostname K8s-Worker**

**Upcoming New 100% Job Ready DevOps/SRE Course - <https://heydevops.in/batch6>**

**Previous Batches Highest Package - 36+ LPA [ Testimonials in website ]**

**Checkout Students Placements - <https://heydevops.in/success-stories>**

**Checkout 14+ Real time projects - <https://heydevops.in/batch6>**

## Part 2 — — — — Both Master & Node — — —

### Install Kubeadm/Kubelet/kubectl

<https://github.com/AWS-AZURE-Bootcamp5/tools-setup-project1/blob/main/kubeadm.sh>

## Part 3 — — — —ON Master Node — — —

```
sudo kubeadm init --pod-network-cidr=10.244.0.0/16
```

#IMPORTANT### in case your in root exit from it and run below commands

```
mkdir -p $HOME/.kube
```

```
sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
```

```
sudo chown $(id -u):$(id -g) $HOME/.kube/config
```

```
kubectl apply -f https://raw.githubusercontent.com/coreos/flannel/master/Documentation/kube-flannel.yml
```

**YOU WILL GET THE TOKEN AND THE COMMAND LIKE BELOW AFTER YOU EXECUTE THE ABOVE COMMAND**

```
sudo kubeadm join <master-node-ip>:<master-node-port> --token <token>  
--discovery-token-ca-cert-hash <hash>
```

**COPY ABOVE TOKEN AND PASTE IN WORKER NODE**

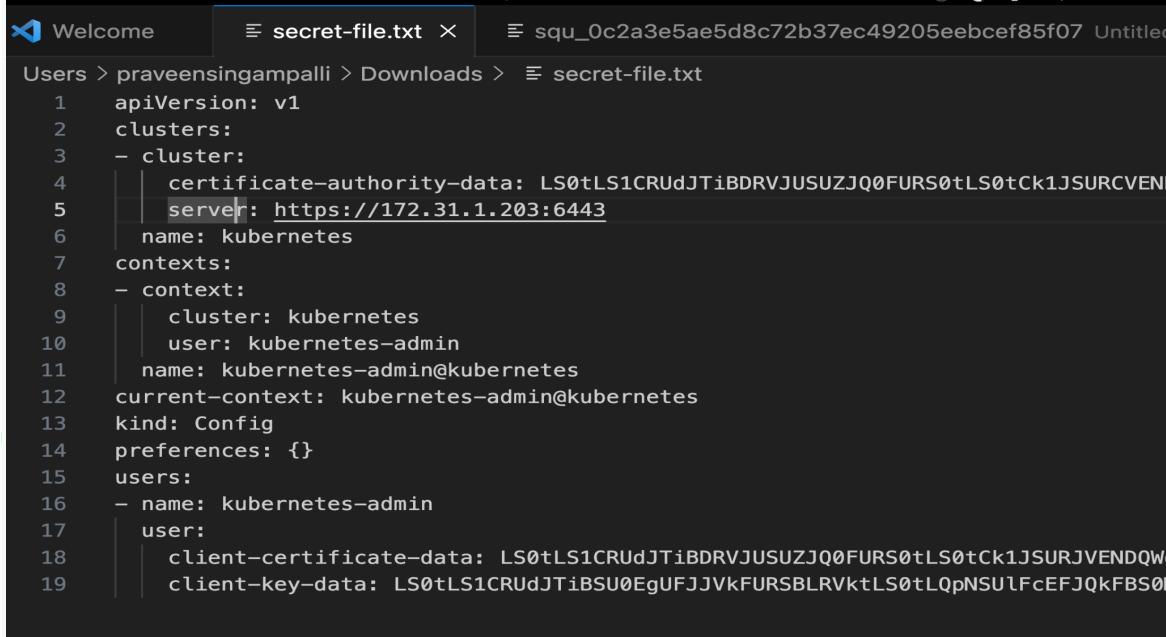
**Upcoming New 100% Job Ready DevOps/SRE Course - <https://heydevops.in/batch6>**

**Previous Batches Highest Package - 36+ LPA [ Testimonials in website ]**

**Checkout Students Placements - <https://heydevops.in/success-stories>**

**Checkout 14+ Real time projects - <https://heydevops.in/batch6>**

Copy the config file FROM K8 MASTER to the local laptop and save it with a name **secret-file.txt** and use this at the Kuberentes credential section



```
>Welcome   secret-file.txt  squ_0c2a3e5ae5d8c72b37ec49205eebcef85f07 Untitled
Users > praveensingampalli > Downloads > secret-file.txt
1  apiVersion: v1
2  clusters:
3  - cluster:
4    | certificate-authority-data: LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSURCVEN
5    | server: https://172.31.1.203:6443
6    | name: kubernetes
7  contexts:
8  - context:
9    | cluster: kubernetes
10   | user: kubernetes-admin
11   | name: kubernetes-admin@kubernetes
12 current-context: kubernetes-admin@kubernetes
13 kind: Config
14 preferences: {}
15 users:
16 - name: kubernetes-admin
17   | user:
18   |   client-certificate-data: LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSURJVENDQW
19   |   client-key-data: LS0tLS1CRUdJTiBSU0EgUFJJVkFURSBLRVktLS0tLQpNSUlFcEFJQkFBS0I
```

## PART 4 - Install Kubernetes Plugins

The screenshot shows the Jenkins 'Plugins' page. On the left, there's a sidebar with links: 'Updates', 'Available plugins' (which is selected and highlighted in grey), 'Installed plugins', 'Advanced settings', and 'Download progress'. The main area has a search bar with 'Kuber' typed in. Below it is a table with columns: 'Install', 'Name', and 'Released'. There are four rows of results:

Install	Name	Released
<input checked="" type="checkbox"/>	Kubernetes Credentials 0.11 kubernetes credentials Common classes for Kubernetes credentials	9 days 16 hr ago
<input checked="" type="checkbox"/>	Kubernetes Client API 6.8.1-224.vd388fca_4db_3b_ kubernetes Library plugins (for use by other plugins) Kubernetes Client API plugin for use by other Jenkins plugins.	9 days 17 hr ago
<input checked="" type="checkbox"/>	Kubernetes 4029.v5712230ccb_f8 Cloud Providers Cluster Management kubernetes Agent Management This plugin integrates Jenkins with Kubernetes	9 days 15 hr ago
<input checked="" type="checkbox"/>	Kubernetes CLI 1.12.1 kubernetes Configure kubectl for Kubernetes	8 days 22 hr ago

**PART 5 - Go to manage Jenkins → manage credentials → Click on Jenkins global → add credentials**

## RUN THE PIPELINE

The screenshot shows the 'New credentials' page in Jenkins. The URL in the address bar is: Dashboard > Manage Jenkins > Credentials > System > Global credentials (unrestricted) >. The page title is 'New credentials'. It has several input fields:

- Kind:** A dropdown menu set to 'Secret file'.
- Scope:** A dropdown menu set to 'Global (Jenkins, nodes, items, all child items, etc)'.
- File:** A 'Choose File' button followed by the path 'Secret File.txt'.
- ID:** An input field containing 'k8s'.
- Description:** An input field containing 'k8s'.

At the bottom is a blue 'Create' button.

**Upcoming New 100% Job Ready DevOps/SRE Course - <https://heydevops.in/batch6>**  
**Previous Batches Highest Package - 36+ LPA [ Testimonials in website ]**  
**Checkout Students Placements - <https://heydevops.in/success-stories>**  
**Checkout 14+ Real time projects - <https://heydevops.in/batch6>**

Declarative: Tool Install	clean workspace	Checkout from Git	Sonarqube Analysis	quality gate	Install Dependencies	OWASP FS SCAN	TRIVY FS SCAN	Docker Build & Push	TRIVY	Deploy to container	Deploy to kubernetes
132ms	264ms	1s	25s	295ms	1min 49s	2min 38s	23s	1min 51s	1min 35s	1s	2s
133ms	261ms	1s	25s	284ms	1min 51s	2min 46s	23s	1min 23s	1min 52s	1s	1s

## PART 5 - In the Kubernetes cluster give this command

kubectl `get` all

kubectl `get` svc

## STEP9:Access from a Web browser with

<public-ip-of-slave:service port>

## Step 10: Terminate instances.

Upcoming New 100% Job Ready DevOps/SRE Course - <https://heydevops.in/batch6>

Previous Batches Highest Package - 36+ LPA [ Testimonials in website ]

Checkout Students Placements - <https://heydevops.in/success-stories>

Checkout 14+ Real time projects - <https://heydevops.in/batch6>

**Upcoming New 100% Job Ready DevOps/SRE Course - <https://heydevops.in/batch6>**  
**Previous Batches Highest Package - 36+ LPA [ Testimonials in website ]**  
**Checkout Students Placements - <https://heydevops.in/success-stories>**  
**Checkout 14+ Real time projects - <https://heydevops.in/batch6>**