

## DevSecOps Training

### DAY – 5

#### IP Address Details:

A	B	C	D	E
S.No	Ip Address- Day5	User Name	Password	Participants Name
Trainer	44.203.124.126	root	redhat	Trainer
1	3.83.116.137	root	redhat	alpana
2	54.208.173.121	root	redhat	baljeet
3	18.208.133.155	root	redhat	Devaraj
4	44.201.211.160	root	redhat	ganesh
5	3.93.60.46	root	redhat	kaustubh
6	54.210.113.138	root	redhat	manish
7	54.85.1.66	root	redhat	manoj
8	54.175.170.49	root	redhat	naveen
9	3.93.79.60	root	redhat	neha
10	54.89.184.246	root	redhat	pravallika
11	3.83.159.126	root	redhat	ramanand sai
12	44.202.87.241	root	redhat	revanth
13	54.89.49.35	root	redhat	rohit
14	54.164.56.49	root	redhat	rudra
15	18.207.216.80	root	redhat	sahitya
16	18.207.202.127	root	redhat	sampat
17	34.227.178.237	root	redhat	sangamesh
18	3.85.219.101	root	redhat	sashi
19	52.207.214.215	root	redhat	shashidhar
20	3.81.226.126	root	redhat	shweta
21	34.201.136.59	root	redhat	sudheer
22	3.86.98.61	root	redhat	uday
23	54.205.240.96	root	redhat	vaishnavi
24	54.152.199.178	root	redhat	vishali
25	54.221.114.138	root	redhat	sankalp
26	44.201.108.177	root	redhat	balakrishna

Trainer ip: 44.203.124.126

#### Details of the machine:

IP Address: 54.152.199.178

---

Trainer is trying to set up the Kubernetes cluster:

```

[root@master ~]# l
-bash: l: command not found
[root@master ~]# ls
calico.yaml  code  devsecopsjava  docker  harsh.py  JenkinsTest
[root@master ~]# kubectl get nodes
NAME      STATUS    ROLES          AGE     VERSION
master    Ready     control-plane,master  2d20h   v1.23.6
node1     Ready     <none>        2d20h   v1.23.6
node2     Ready     <none>        2d20h   v1.23.6
node3     Ready     <none>        2d20h   v1.23.6
node4     Ready     <none>        2d20h   v1.23.6
node5     Ready     <none>        2d20h   v1.23.6
[root@master ~]# cd /var/www/html/
[root@master html]# ls
config  index.html
[root@master html]# kubectl get nodes --kubeconfig config
NAME      STATUS    ROLES          AGE     VERSION
master    Ready     control-plane,master  2d20h   v1.23.6
node1     Ready     <none>        2d20h   v1.23.6
node2     Ready     <none>        2d20h   v1.23.6
node3     Ready     <none>        2d20h   v1.23.6
node4     Ready     <none>        2d20h   v1.23.6
node5     Ready     <none>        2d20h   v1.23.6
[root@master html]# wget 3.137.37.209/config ;

```

To get the access to Kubernetes cluster:

```

[root@master html]# wget 3.137.37.209/config ; mkdir $HOME/.kube/ ; cp config $HOME/.kube/ ; kubectl get nodes

```

TASK:

1. Install Kubectl command in your machine (Make sure kubectl is working!)
2. If kubectl is working already, then no need to follow this step: Run the below command to get the access of the running kubernetes cluster

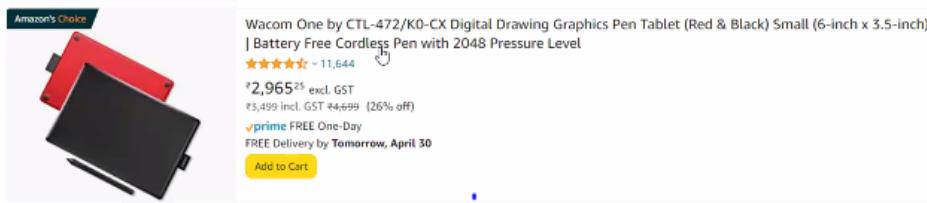
`wget 3.137.37.209/config; mkdir $HOME/.kube/; cp config $HOME/.kube/; kubectl get nodes`

```

[root@vishali ~]# wget 3.137.37.209/config ; mkdir $HOME/.kube/ ; cp config $HOME/.kube/ ; kubectl get nodes
--2022-04-29 04:09:51-- http://3.137.37.209/config
Connecting to 3.137.37.209:80... failed: Connection refused.
mkdir: cannot create directory '/root/.kube/': File exists
cp: overwrite '/root/.kube/config'? yes
NAME      STATUS    ROLES          AGE     VERSION
master    Ready     control-plane,master  2d20h   v1.23.6
node1     Ready     <none>        2d20h   v1.23.6
node2     Ready     <none>        2d20h   v1.23.6
node3     Ready     <none>        2d20h   v1.23.6
node4     Ready     <none>        2d20h   v1.23.6
node5     Ready     <none>        2d20h   v1.23.6

```

**Trainer uses this device to explain:**



## Agenda:

**Do SAST and DAST testing:**

Create a “My SAST Projects” view as a list view:

## Flow:

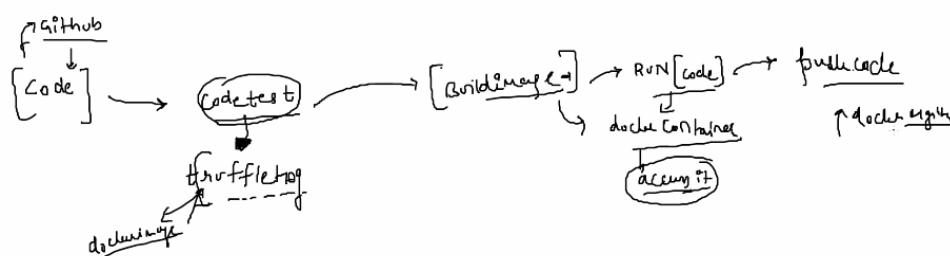
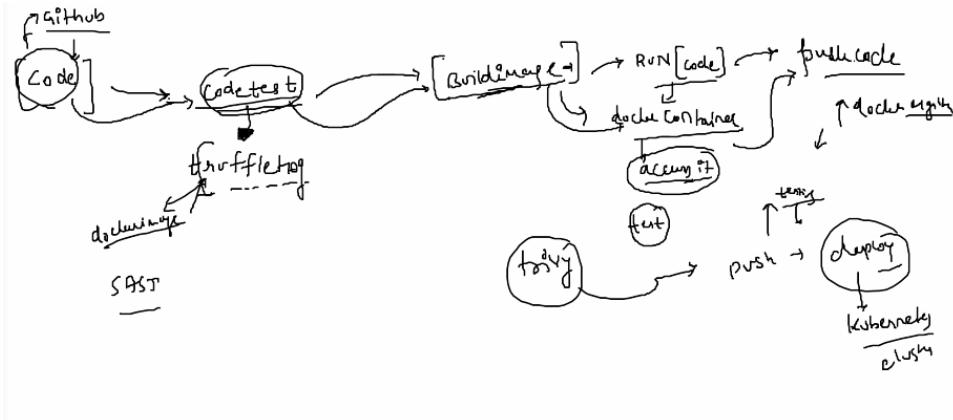


Image testing is done by Trivy:



Delete all the pods:

```
[root@master ~]# kubectl delete all --all --force
warning: Immediate deletion does not wait for confirmation that
run on the cluster indefinitely.
pod "collinsapp-7c455fbb89-tngxn" force deleted
pod "mayank-6876cf4c88-q2j2l" force deleted
service "collinsapp" force deleted
service "mayank" force deleted
deployment.apps "collinsapp" force deleted
deployment.apps "mayank" force deleted
[root@master ~]# kubel
```

Get pods:

```
deployment.apps 'mayank' force deleted
[root@master ~]# kubectl get pods
NAME           READY   STATUS    RESTARTS   AGE
collinsapp-7c455fbb89-fsl9c  1/1   Terminating   0          4s
[root@master ~]# kubectl get all
NAME           READY   STATUS    RESTARTS   AGE
pod/collinsapp-7c455fbb89-fsl9c  1/1   Terminating   0          6s
[root@master ~]#
```

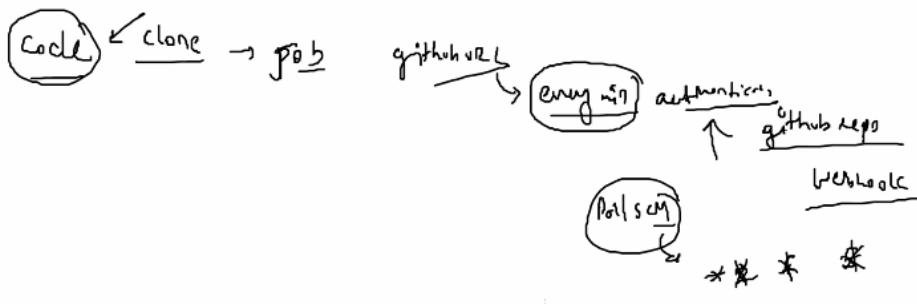
Run the above command:

```
https://aws.amazon.com/amazon-linux-2/
17 package(s) needed for security, out of 34 available
Run "sudo yum update" to apply all updates.
[root@vishali ~]# kubectl delete all --all --force
warning: Immediate deletion does not wait for confirmation that the running resource
has been terminated. The resource may continue to run on the cluster indefinitely.
No resources found
[root@vishali ~]# kubectl get pods
No resources found in default namespace.
[root@vishali ~]#
```

We will try to fork a project:

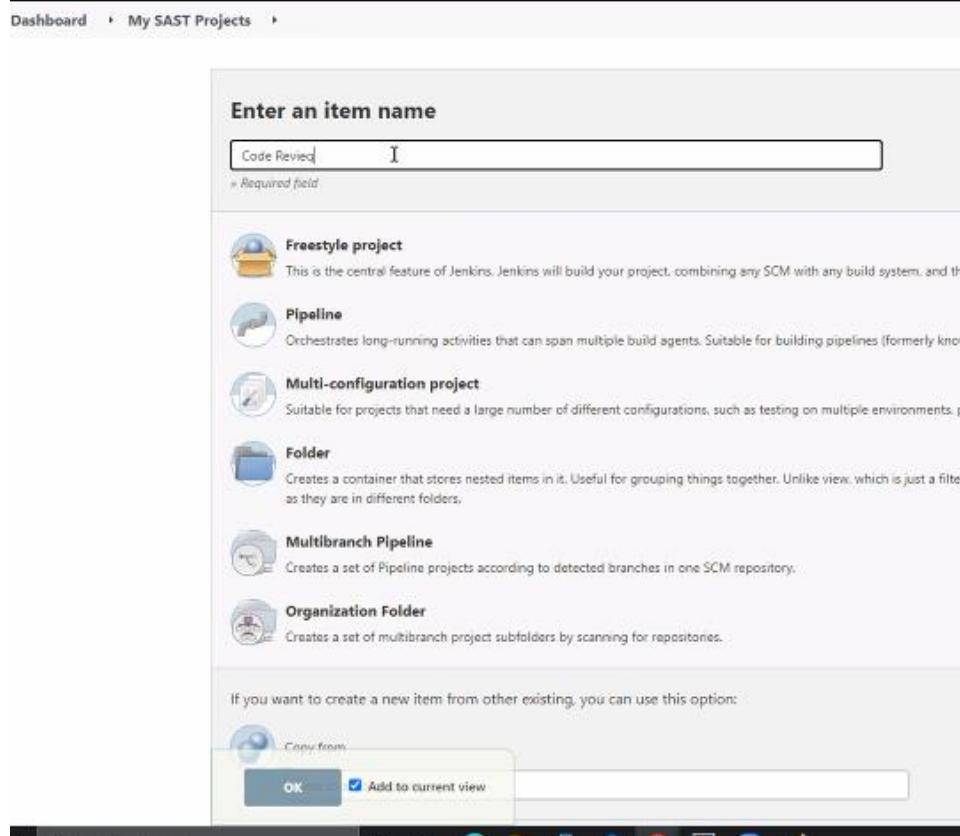
Then we will do a test code using truffle code.

Flow:



Now, go to the jenkins -> My SAST Project (view)

Create a job:



Add GitHub URL and then mention build trigger:

The screenshot shows the 'Build Triggers' configuration for a 'Code Review' project. The 'Build Triggers' tab is selected. Under 'Additional Behaviours', there is an 'Add' button. Under 'Build Triggers', the 'Poll SCM' checkbox is checked. The 'Schedule' section is collapsed. A note below says 'No schedules so will only run due to SCM changes if triggered by a post-commit hook'. The 'Ignore post-commit hooks' checkbox is unchecked. The 'Build Environment' section contains several checkboxes for workspace management, timestamps, and build tools like Ant. At the bottom, there are 'Save' and 'Apply' buttons.

Inspect build log for published Gradle build scans  
 With Ant ?

### Build

Execute shell

?

Command

```
ls
docker version
kubectl get nodes
```

**X**

See the list of available environment variables

Save Apply



We are getting an error: why? kubectl will only take the credentials file from home loc

Dashboard → My SAST Projects → Code Review → #1

```
Commit message: "Update README.md"
First time build. Skipping changelog.
[Code Review] $ /bin/sh -xe /tmp/jenkins2547191623927044506.sh
+ ls
Dockerfile
README.md
webapp
+ docker version
Client:
Version: 20.10.7
API version: 1.41
Go version: go1.15.14
Git commit: f0df358
Built: Wed Nov 17 03:05:36 2021
OS/Arch: linux/amd64
Context: default
Experimental: true

Server:
Engine:
Version: 20.10.7
API version: 1.41 (minimum version 1.12)
Go version: go1.15.14
Git commit: b0f5bc3
Built: Wed Nov 17 03:06:14 2021
OS/Arch: linux/amd64
Experimental: false
containerd:
Version: 1.4.6
GitCommit: d71fcd7d8303cbf684402823e425e9dd2e99985d
runc:
Version: 1.0.0
GitCommit: 84113ee56fc27af1b01b3181f31bbef700715301
docker-init:
Version: 0.19.0
GitCommit: de40ad8
+ kubectl get nodes
error: the server doesn't have a resource type "nodes"
Build step 'Execute shell' marked build as failure
Finished: FAILURE
```

Check the user: root

```
[root@master ~]# ls -la
total 316
dr-xr-x--- 14 root root 4096 Apr 28 12:53 .
dr-xr-xr-x 18 root root 257 Apr 26 06:49 ..
drwxr-xr-x 2 root root 39 Apr 27 09:03 .aws
-rw----- 1 root root 36373 Apr 28 12:53 .bash_history
-rw-r--r-- 1 root root 18 Oct 18 2017 .bash_logout
-rw-r--r-- 1 root root 176 Oct 18 2017 .bash_profile
-rw-r--r-- 1 root root 218 Apr 27 08:57 .bashrc
drwx----- 3 root root 19 Apr 27 11:44 .cache
-rw-r--r-- 1 root root 222270 Apr 26 07:16 calico.yaml
drwxr-xr-x 2 root root 45 Apr 26 10:58 code
-rw-r--r-- 1 root root 100 Oct 18 2017 .cshrc
drwxr-xr-x 4 root root 99 Apr 28 11:06 devsecopsjava
drwxr-xr-x 2 root root 42 Apr 27 04:59 docker
drwx----- 2 root root 25 Apr 28 10:27 .docker
-rw-r--r-- 1 root root 50 Apr 28 06:35 .gitconfig
-rwxr-xr-x 1 root root 2603 Apr 27 13:17 harsh.py
drwxr-xr-x 3 root root 54 Apr 28 06:43 JenkinsTest
drwxr-xr-x 3 root root 33 Apr 27 04:09 .kube
-rw-r--r-- 1 root root 89 Apr 27 13:27 mayank.py
drwxr-xr-x 4 root root 39 Apr 28 11:34 project-html-website
-rw-r----- 1 root root 4364 Apr 28 06:24 root@54.152.185.61
drwx----- 2 root root 48 Apr 27 09:56 .ssh
-rw-r--r-- 1 root root 129 Oct 18 2017 .tcshrc
drwxr-xr-x 3 root root 122 Apr 27 09:43 terracode
drwxr-xr-x 2 root root 58 Apr 27 09:07 .terraform.d
-rw----- 1 root root 12492 Apr 28 12:53 .viminfo
[root@master ~]#
```

Copy the config file and give permission:

Give your kube credentials to jenkins.

```
[root@master ~]# mkdir /var/lib/jenkins
[root@master ~]# mkdir /var/lib/jenkins/.kube
mkdir: cannot create directory '/var/lib/jenkins/.kube': File exists
[root@master ~]# cp .kube/config /var/lib/jenkins/.kube/
[root@master ~]# chmod 777 /var/lib/jenkins/ -R
[root@master ~]#
```

Build is success now:


**Jenkins**  
[Dashboard](#) → [My SAST Projects](#) → [Code Review](#) → #3

[Back to Project](#)

[Status](#)

[Changes](#)

**Console Output**

[View as plain text](#)

[Edit Build Information](#)

[Delete build #3](#)

[Git Build Data](#)

[Previous Build](#)

## Console Output

```

Started by user mayank
Running as SYSTEM
Building in workspace /var/lib/jenkins/workspace/Code Review
The recommended git tool is: NONE
No credentials specified
> git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/Code Review/.git # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url https://github.com/mdhak0316/cnuebapp # timeout=10
Fetching upstream changes from https://github.com/mdhak0316/cnuebapp
> git --version # timeout=10
> git --version # 'git version 2.32.0'
> git fetch --tags --force --progress -- https://github.com/mdhak0316/cnuebapp +refs/heads/*:refs/remotes/origin/*
> git rev-parse refs/remotes/origin/master^{commit} # timeout=10
Checking out Revision 10c4179afeb570bc4726ce807cf2a448f0e5fc (refs/remotes/origin/master)
> git config core.sparsecheckout # timeout=10
> git checkout -f 10c4179afeb570bc4726ce807cf2a448f0e5fc # timeout=10
Commit message: "Update README.md"
> git rev-list --no-walk 10c4179afeb570bc4726ce807cf2a448f0e5fc # timeout=10
[Code Review] $ /bin/sh -xe /tmp/jenkins5715582482235777683.sh
+ ls
Dockerfile
README.md
webapp
+ docker version
Client:
Version:      20.10.7
API version:  1.41
Go version:   go1.15.14
Git commit:   f0dfr398
Built:        Wed Nov 17 03:05:30 2021
OS/Arch:      linux/amd64
Context:      default

```

Next step is to test the code:

[Dashboard](#) → [My SAST Projects](#)

**Enter an item name**

[Static Testing](#) (Required field)

**Freestyle project**  
This is the central feature of Jenkins. Jenkins will build your project, or run tests, or perform other activities.

**Pipeline**  
Orchestrates long-running activities that can span multiple build agents.

**Multi-configuration project**  
Suitable for projects that need a large number of different configurations.

**Folder**  
Creates a container that stores nested items in it. Useful for grouping items as they are in different folders.

**Multibranch Pipeline**  
Creates a set of Pipeline projects according to detected branches in a repository.

**Organization Folder**  
Creates a set of multibranch project subfolders by scanning for repositories.

If you want to create a new item from other existing, you can use this

[Create from...](#)

**OK**  Add to current view

Build command:

```
docker run --rm gesellix/truffelehog https://github.com/mdhack0316/cnawebapp > status.txt
if [ `cat status.txt | wc -l` -gt 0 ]
then
    echo "There is Some Bug Plz"

```

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

#### Command

```
if [ `cat status.txt | wc -l` -gt 0 ]
then
    echo "There is Some Bug Please Fix IT and Try again" I
    exit 1

```

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

#### Build

##### Execute shell

?

##### Command

```
exit 1
else
    echo "Go to GO"
fi
```

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

What is fi -> closing of the if block

In previous block – give this:

With Ant ?

#### Build

##### Execute shell

?

##### Command

```
ls
docker version
kubectl get nodes
```

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

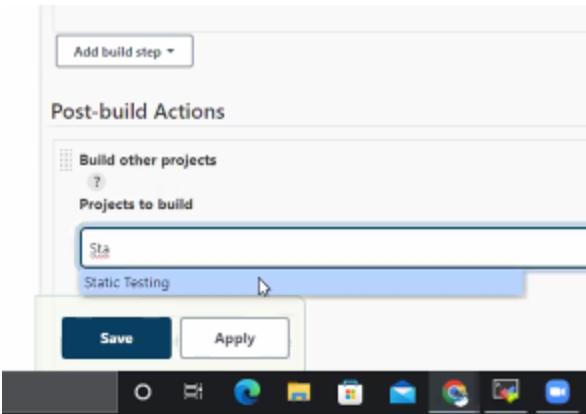
Add build step ▾

#### Post-build Actions

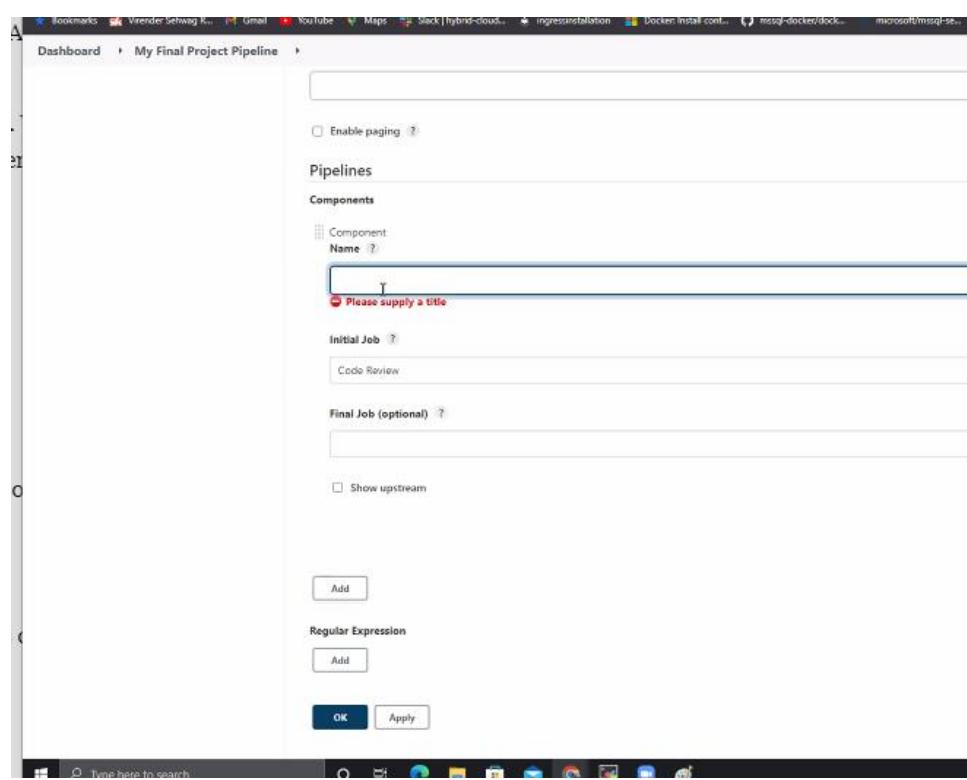
Add post-build action ▾

Save Apply

Add post build act



Create a delivery pipeline:



Make some change in the GitHub:

[mdhack0316 / cnawebapp](#) Public  
Forked from redashu/cnawebapp

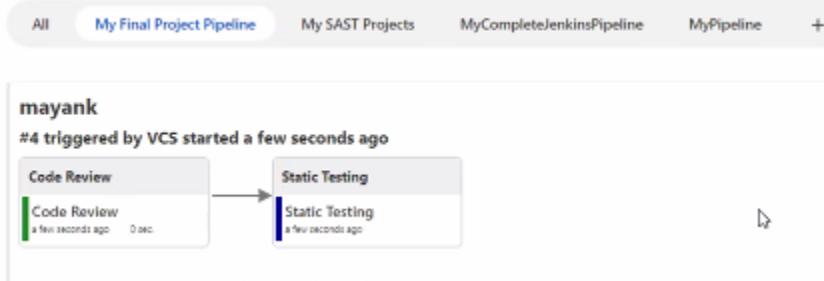
Code Pull requests Actions Projects Wiki Security Insights Settings

cnawebapp / README.md in master

Edit file Preview

```
1  # cnawebapp
2
3  ## add sample changes
4  ## 5 pipeline test
5  New test
```

Build will automatically trigger.



Failing – build: Problem: truffle name – check!

Dashboard > Static Testing > #1

Back to Project Status Changes Console Output Edit Build Information Delete build '#1'

## ✖ Build #1 (29-Apr-2022, 4:42:20 AM)

No changes.

Started by upstream project [Code Review](#) build number 4 originally caused by:

- Started by an SCM change

Change – give correct name:

**Build**

Execute shell

?

Command

```
docker run --rm gesellix/trufflehog https://github.com/edhack0316/cnakebapp > status.txt
if [ `cat status.txt | wc -l` -gt 0 ]
then
    echo "There is Some Bug Please Fix IT and Try again"
    exit 1
else
    echo "Go to GO"
fi
```

See the list of available environment variables

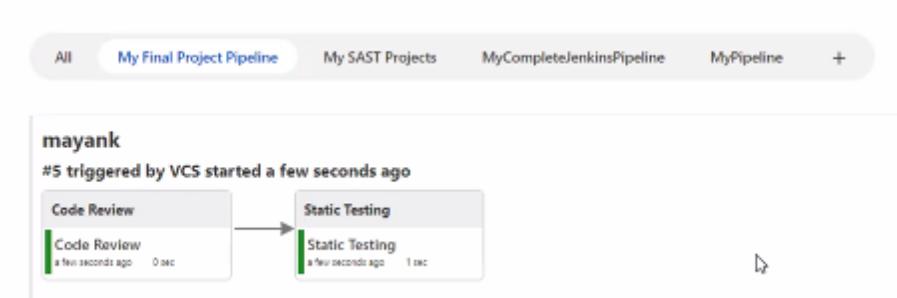
Add build step ▾

**Post-build Actions**

Add post-build action ▾

**Save** **Apply**

Make some changes in the GitHub and build will be trigger automatically:



Docker build image – next step:

Enter an item name

Building Docker Im|  
» Required field

**Freestyle project**  
This is the central feature of Jenkins. Jenkins will build your project.

**Pipeline**  
Orchestrates long-running activities that can span multiple builds.

**Multi-configuration project**  
Suitable for projects that need a large number of different configurations.

**Folder**  
Creates a container that stores nested items in it. Useful for grouping items that are in different folders.

**Multibranch Pipeline**  
Creates a set of Pipeline projects according to detected branches.

**Organization Folder**  
Creates a set of multibranch project subfolders by scanning files.

If you want to create a new item from other existing, you can use the "Copy from..." button.

Copy from...   
**OK** Cancel Auto-complete

The screenshot shows a Jenkins pipeline configuration page. At the top, there are tabs: General, Source Code Management, Build Triggers (which is selected), Build Environment, Build, and Post-build Actions. A blue arrow points from the 'Build Triggers' tab to the 'General' tab. Handwritten in blue across the top left is the word 'GIT'. The 'Build Triggers' section contains several options: Trigger builds remotely (e.g., from scripts), Build after other projects are built, Build periodically, GitHub hook trigger for GITScm polling, and Poll SCM. The 'Build Environment' section includes options like Delete workspace before build starts, Use secret text(s) or file(s), Abort the build if it's stuck, Add timestamps to the Console Output, Create Delivery Pipeline version, and Inspect build log for published Gradle build scans. The 'Build' section contains an 'Execute shell' step with the command 'docker build -t adhack0316/myFinal:v1 .' highlighted with a blue box. Below the command is a note: 'See the list of available environment variables'. At the bottom are 'Save' and 'Apply' buttons.

How is this build going to trigger now?

The screenshot shows a Jenkins dashboard under the 'My SAST Projects' view. On the left, there are sections for 'Lockable Resources', 'New View', 'Build Queue' (which is empty), and 'Build Executor Status' (which shows 1 Idle and 2 Idle executors). On the right, there are 'Job Filters' with options: Filter build executors, Recurse in subfolders, and a list of jobs: Apache Server, Building Docker Image, Code Review (which has a blue checkmark), Code Commit, DeployApp, Docker Website, Git Website, ImageCreation, Static Testing (which also has a blue checkmark), Testing, and testproject. There is also an option to 'Use a regular expression to include jobs into the view' and a 'Add Job Filter' button. A blue checkmark is placed next to the 'Code Review' checkbox in the 'Job Filters' section.

Dashboard > My SAST Projects > Static Testing >

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

With Ant ?

**Build**

Execute shell

?

Command

```
docker run --rm gosellix/trufflehog https://github.com/ndhack0316/cnawebapp > status.txt
if [ `cat status.txt | wc -l` -gt 8 ]
then
    echo "There is Some Bug Please Fix IT and Try Again"
    exit 1
else
    echo "No Bug Found"
fi
```

Aggregate downstream test results

Archive the artifacts

Build other projects

**Publish JUnit test result report**

Record fingerprints of files to track usage

Git Publisher

E-mail Notification

Editable Email Notification

Set GitHub commit status (universal)

Set build status on GitHub commit [deprecated]

Trigger parameterized build on other projects

Delete workspace when build is done

Add post-build action ▾

**Save** **Apply**

Dashboard > My SAST Projects > Static Testing >

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

See the list of available environment variables

Add build step ▾

**Post-build Actions**

Build other projects

?

Projects to build

**Build**

Building Docker image

Trigger only if build is stable

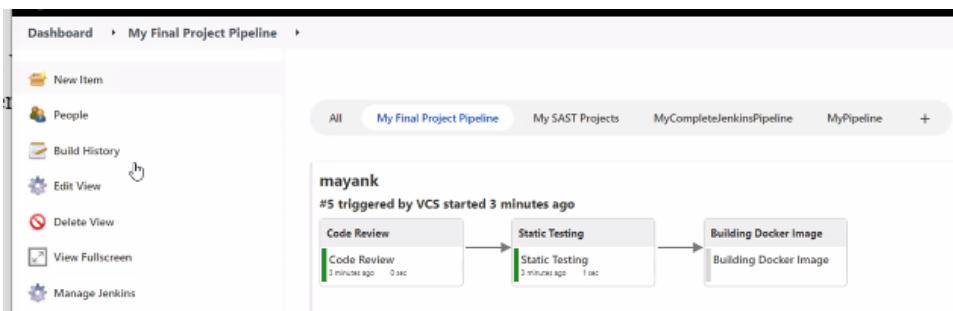
Trigger even if the build is unstable

Trigger even if the build fails

Add post-build action ▾

**Save** **Apply**

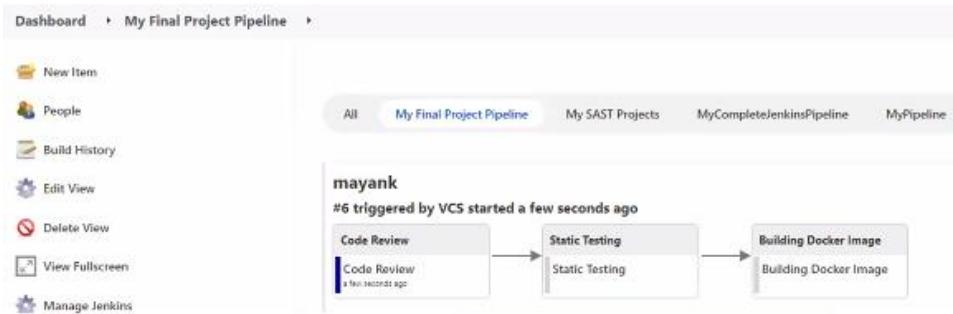




Make changes:

The screenshot shows a GitHub commit changes dialog. It includes fields for updating README.md and adding an optional extended description. Below these fields is a dropdown menu with two options: one selected (Commit directly to the master branch) and another (Create a new branch for this commit and start a pull request). At the bottom, there are 'Saving...' and 'Cancel' buttons.

Build started:



Check docker image: You will see the image:

```
[root@master ~]# docker images
REPOSITORY          TAG      IMAGE ID      CREATED     SIZE
mdhack0316/myfinal    v1      0d734b160056  5 seconds ago  142MB
newimage              latest   8dce3451f4e9  19 hours ago  144MB
mdhack0316/testwithcollins    latest   8dce3451f4e9  19 hours ago  144MB
jenkins_docker        latest   a09eacf4d706  20 hours ago  144MB
mayanktest            latest   373987aa2ad7  2 days ago   393MB
mdhack/collinsimage  latest   373987aa2ad7  2 days ago   393MB
nginx                latest   fa5269854a5e  8 days ago   142MB
httpd                latest   c30a46771695  8 days ago   144MB
calico/cni            v3.22.2  d6660bf471e1  2 weeks ago  236MB
calico/pod2daemon-flexvol  v3.22.2  fd1608dbbc19  2 weeks ago  198MB
calico/node           v3.22.2  8fa62c12256d  2 weeks ago  135MB
k8s.gcr.io/kube-apiserver  v1.23.6  df7b72818ad2  2 weeks ago  125MB
k8s.gcr.io/kube-controller-manager  v1.23.6  595f327f224a  2 weeks ago  53.5MB
k8s.gcr.io/kube-scheduler        v1.23.6  4c0375452406  2 weeks ago  112MB
k8s.gcr.io/kube-proxy          v1.23.6  25f8c7f3da61  5 months ago  293MB
k8s.gcr.io/etcetd            3.5.1-0   a4ca41631cc7  6 months ago  46.8MB
k8s.gcr.io/coredns/coredns    v1.8.6   6270bb605e12  8 months ago  683kB
k8s.gcr.io/pause             3.6      816d99f0bbe8  12 months ago  224MB
oraclelinux              8.3      e8ce7504414a  21 months ago  350MB
mdhack/myserver            latest   c6e3cd9aae36  3 years ago   84.8MB
quay.io/mayank123modi/simple-webapp  latest   2488ade2fa0c  4 years ago   97MB
[root@master ~]#
```

Now test whether it is going to give correct output or not:

Dashboard → My SAST Projects →

**Enter an item name**

Testing Code Locally

Required field\*

**Freestyle project**  
This is the central feature of Jenkins. Jenkins will build ;

**Pipeline**  
Orchestrates long-running activities that can span multiple stages.

**Multi-configuration project**  
Suitable for projects that need a large number of different configurations.

**Folder**  
Creates a container that stores nested items in it. Useful if your items are in different folders.

**Multibranch Pipeline**  
Creates a set of Pipeline projects according to detected branches.

**Organization Folder**  
Creates a set of multibranch project subfolders by scanning a directory.

If you want to create a new item from other existing, you can do so by clicking on 'Create from...' button.

**Create from...**

**OK**  Add to current view

Build command:

The screenshot shows the SonarQube dashboard for a project named 'Testing Code Locally'. The 'Source Code Management' tab is active. Under 'Source Code Management', 'None' is selected. In the 'Build Triggers' section, there are several options: 'Trigger builds remotely (e.g., from scripts)', 'Build after other projects are built', 'Build periodically', 'GitHub hook trigger for GITScm polling', and 'Poll SCM'. In the 'Build Environment' section, options include 'Delete workspace before build starts', 'Use secret text(s) or file(s)', 'Abort the build if it's stuck', 'Add timestamps to the Console Output', 'Create Delivery Pipeline version', 'Inspect build log for published Gradle build scans', and 'With Ant'. The 'Build' section contains a command box with the following content:  
docker run -itd --name mayanktest1 mdhack0316/myfinal:v1 |  
See the list of available environment variables  
At the bottom are 'Save' and 'Apply' buttons.

Curl – used to test it.

Now we will do manually:

```
gesellix/trufflehog      latest    2488ade2fa8c  4 years ago   97MB
[root@master ~]# docker run -itd --name mayanktest1 mdhack0316/myfinal:v1
21cd49ee6f0895da355bc8db6de1e7450d0565e22fb9256255cec6ae0580bd9f
[root@master ~]#
```

Curl that ip:

Find the ip address using: Docker inspect <container name>:

```
[root@master ~]# curl 172.17.0.2
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>cisco</title>
</head>
<body>
    <h1> Hello guys </h1>
    <h2> please follow the stpes </h2>
    <h3> Login page web are about to create </h3>

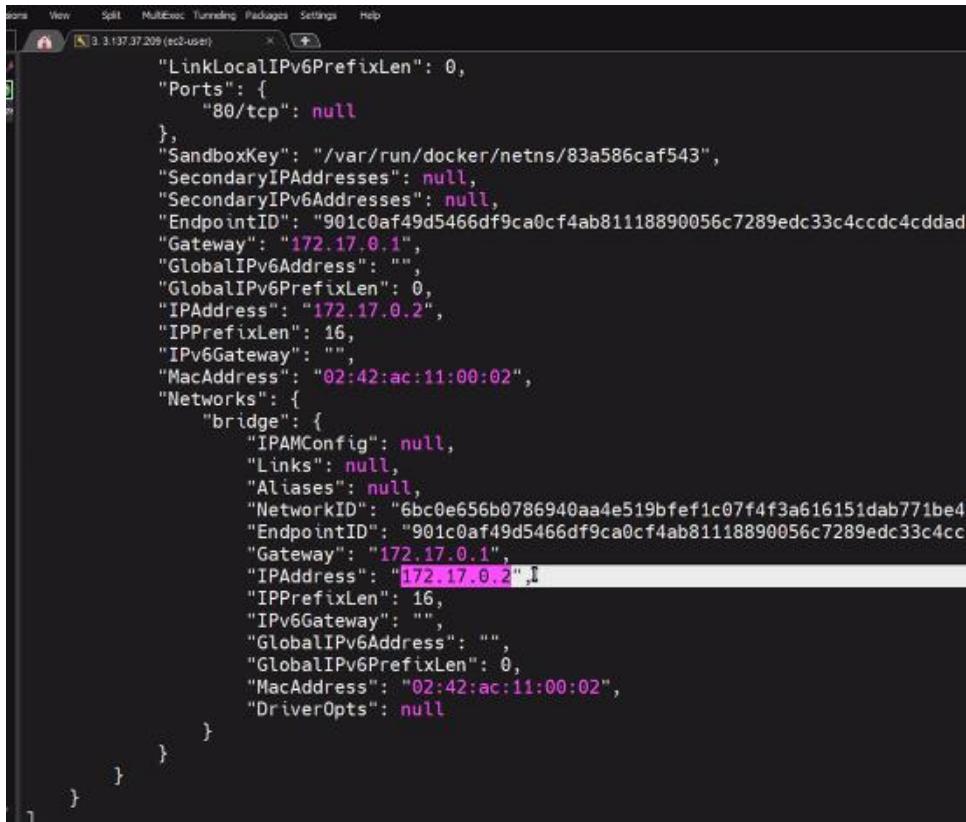
    <a href="blogs/index.html">click here </a>
    

</body>
</html>[root@master ~]#
```

```
[root@master ~]# curl 172.17.0.2 | grep -i hello
% Total    % Received % Xferd  Average Speed   Time   Time     Time Current
          Dload  Upload Total Spent   Left Speed
100  448  100  448    0      0  324k      0 --:--:-- --:--:-- 437k
<h1> Hello guys!</h1>
[root@master ~]#
```

```
[root@master ~]# curl 172.17.0.2 | grep -i skdfdf
% Total    % Received % Xferd  Average Speed   Time   Time     Time Current
          Dload  Upload Total Spent   Left Speed
100  448  100  448    0      0  338k      0 --:--:-- --:--:-- 437k
[root@master ~]#
```

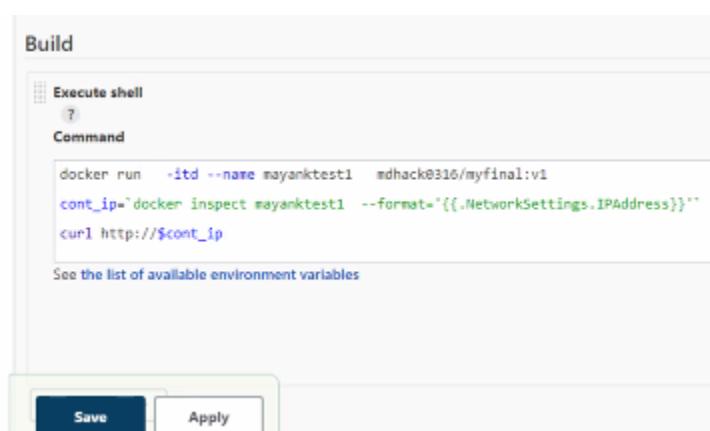
Docker inspect <container name>:



```
...  
"LinkLocalIPv6PrefixLen": 0,  
"Ports": {  
    "80/tcp": null  
},  
"SandboxKey": "/var/run/docker/netns/83a586caf543",  
"SecondaryIPAddresses": null,  
"SecondaryIPv6Addresses": null,  
"EndpointID": "901c0af49d5466df9ca0cf4ab8118890056c7289edc33c4ccdc4cdad",  
"Gateway": "172.17.0.1",  
"GlobalIPv6Address": "",  
"GlobalIPv6PrefixLen": 0,  
"IPAddress": "172.17.0.2",  
"IPPrefixLen": 16,  
"IPv6Gateway": "",  
"MacAddress": "02:42:ac:11:00:02",  
"Networks": {  
    "bridge": {  
        "IPAMConfig": null,  
        "Links": null,  
        "Aliases": null,  
        "NetworkID": "6bc0e656b0786940aa4e519bfef1c07f4f3a616151dab771be4",  
        "EndpointID": "901c0af49d5466df9ca0cf4ab8118890056c7289edc33c4cc",  
        "Gateway": "172.17.0.1",  
        "IPAddress": "172.17.0.2",  
        "IPPrefixLen": 16,  
        "IPv6Gateway": "",  
        "GlobalIPv6Address": "",  
        "GlobalIPv6PrefixLen": 0,  
        "MacAddress": "02:42:ac:11:00:02",  
        "DriverOpts": null  
    }  
}  
}  
}
```

Use this command to get only the ip:

```
[root@master ~]# docker inspect mayanktest1 --format='{{.NetworkSettings.IPAddress}}'  
172.17.0.2  
[root@master ~]#
```



Remove the container and provide the grep:

RM

```
Execute shell
?
Command
docker run -itd --name mayanktest1 mdhack0316/myfinal:v1
cont_ip=$(docker inspect mayanktest1 --format="{{.NetworkSettings.IPAddress}}")
curl http://$cont_ip | grep hello
See the list of available environment variables
```

Save Apply

### Post action build: - add it

hboard → My SAST Projects → Building Docker Image →

General Source Code Management Build Triggers Build Environment Build

See the list of available environment variables

Add build step ▾

Post-build Actions

Build other projects ?

Projects to build

Testing Code Locally.

No such project 'Tes'. Did you mean 'Testing'?

Trigger only if build is stable

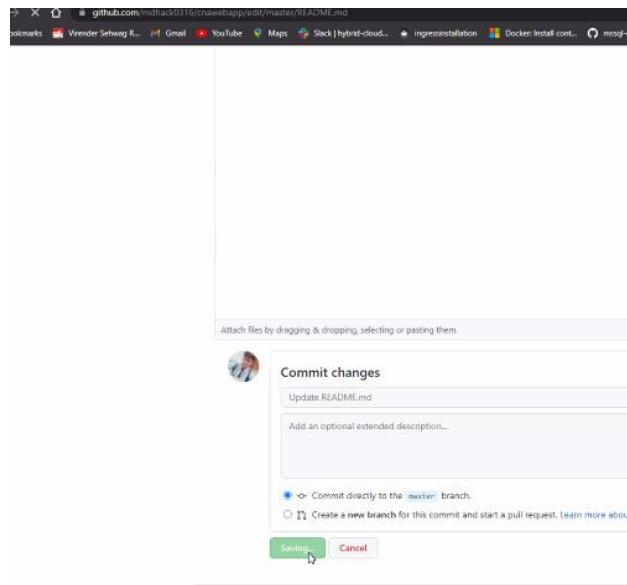
Trigger even if the build is unstable

Trigger even if the build fails

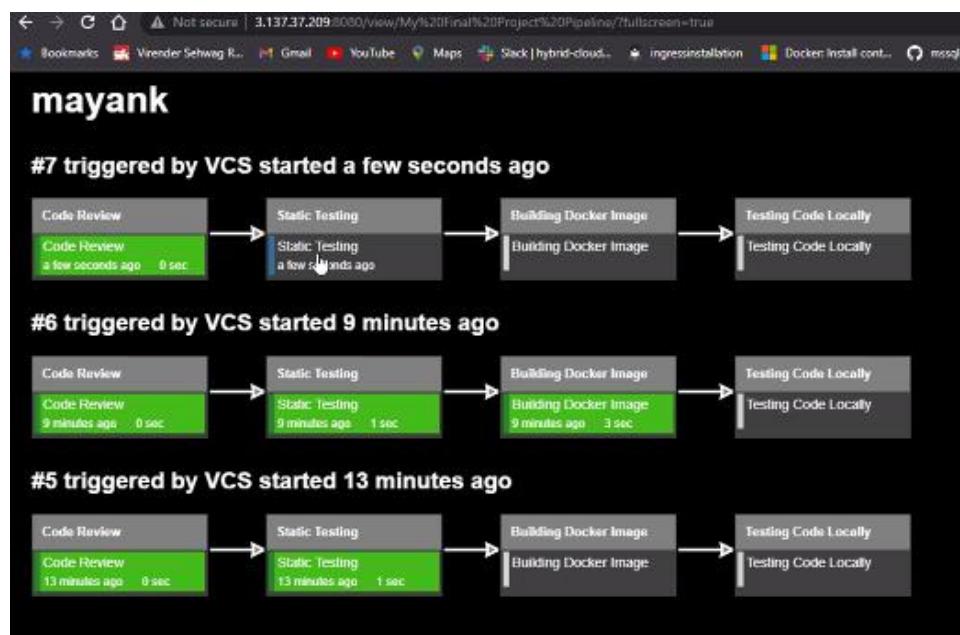
Add post-build action ▾

Save Apply

Make changes:



## Build in full screen: Check the build



Build failed:

The screenshot shows the Jenkins interface for a job named "Testing Code Locally". The "Console Output" tab is selected. The output shows a series of commands being run in a shell, including docker rm, docker run, docker inspect, and curl. The curl command fails with the error "(28) Failed writing body". The build step "Executes shell" is marked as failure, and the build is finished with a FAILURE status.

```

Started by upstream project "Building Docker Image" build number 2
originally caused by:
Started by upstream project "Static Testing" build number 4
originally caused by:
Started by upstream project "Code Review" build number 7
originally caused by:
Started by an SCM change
Running as SYSTEM
Building in workspace /var/lib/jenkins/workspace/Testing Code Locally
[Testing Code Locally] $ ./bin/sh -xe /tmp/jenkins4874638007375955149.sh
+ docker rm -f mayanktest1
mayanktest1
+ docker run -itd --name mayanktest1 mdhack0310/myfinal:v1
1b157b148afc3c69e64d14017180da4id7ac8c77ffa17a25f577803af356acac
++ docker inspect mayanktest1 --format='{{.NetworkSettings.IPAddress}}'
+ cont_ip=172.17.0.2
+ grep i hello
+ curl http://172.17.0.2
grep: Hello: No such file or directory
  % Total    % Received % Xferd  Average Speed   Time     Time     Time  Current
          Dload  Upload Total Spent   Spent    Left  Speed
0      0      0      0      0      0      0      0      0      0      0      0
100  448  100  448      0      0  294k      0      0      0      0      0      0
curl: (28) Failed writing body
Build step 'Executes shell' marked build as failure
Finished: FAILURE

```

Give -i:

The screenshot shows the Jenkins build configuration for the "Testing Code Locally" job. In the "Build" section, under the "Execute shell" step, the command is modified to include the "-i" option with the curl command. This is done to handle the failure of writing the body. The "Post-build Actions" section is also visible, along with "Save" and "Apply" buttons at the bottom.

```

docker rm -f mayanktest1
docker run -itd --name mayanktest1 mdhack0310/myfinal:v1
cont_ip=`docker inspect mayanktest1 --format='{{.NetworkSettings.IPAddress}}'`
curl http://$cont_ip | grep -i hello

```

Credentials task:

While creating a job, there also you can provide the credentials, how? Trainer will show

The screenshot shows the Jenkins Plugin Manager interface. At the top, there's a search bar with the text 'Search' and a user icon labeled 'mayank'. Below the search bar, the title 'Plugin Manager' is displayed. Underneath it, there are tabs for 'Updates', 'Available' (which is selected), 'Installed', and 'Advanced'. A search input field contains the text 'docker build'. The results list includes three items:

- CloudBees Docker Build and Publish 1.3** (Build Tools - Docker) - Released 1 yr 2 mo ago. Description: This plugin enables building Dockerfile based projects, as well as publishing of the built images/repos to the docker registry.
- Google Container Registry Auth 0.3** (Build Tools - Docker) - Released 6 yr 5 mo ago. Description: This plugin exposes a credential for use with the Docker Build Step plugin for authenticating with Google Container Registry as a service account.
- Aqua MicroScanner 1.0.8** (Build Tools - Docker) - Released 2 yr 7 mo ago. Description: Enables scanning of docker build for OS package vulnerabilities.

At the bottom of the list, there are buttons for 'Install without restart', 'Download now and install after restart', and 'Check now'.

Without using docker tag command, we can push the image. Install the above plugin.

Jenkins will restart:



Please wait while Jenkins is restarting ...

Your browser will reload automatically when Jenkins is ready.

Create an item:

**Enter an item name**

Required field

 **Freestyle project**  
This is the central feature of Jenkins. Jenkins will build y

 **Pipeline**  
Orchestrates long-running activities that can span multi

 **Multi-configuration project**  
Suitable for projects that need a large number of differen

 **Folder**  
Creates a container that stores nested items in it. Useful as they are in different folders.

 **Multibranch Pipeline**  
Creates a set of Pipeline projects according to detected

 **Organization Folder**  
Creates a set of multibranch project subfolders by scan

If you want to create a new item from other existing, you

 *Copy from*

Add to current view

Provide the Git URL:

Source Code Management

None

Git [?](#)

[Repositories](#) [?](#)

Repository URL [?](#)

⚠ Please enter Git repository.

Credentials [?](#)

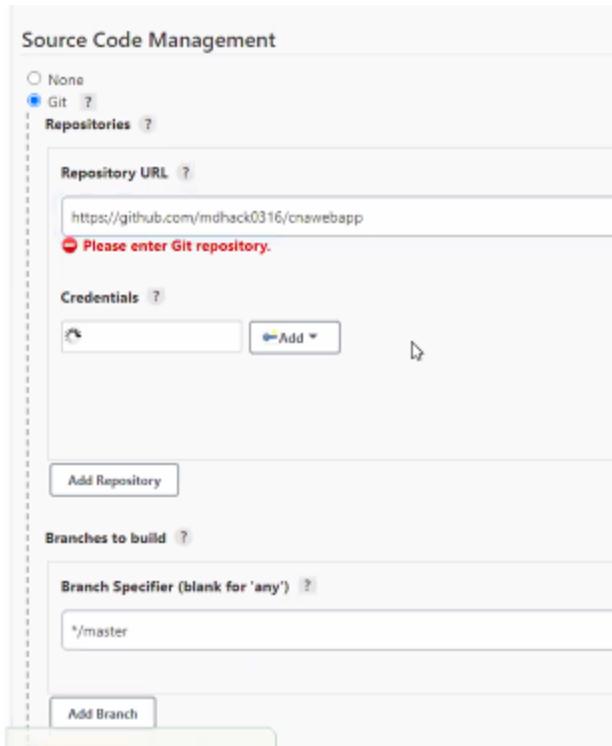
[Add](#) [?](#)

[Add Repository](#)

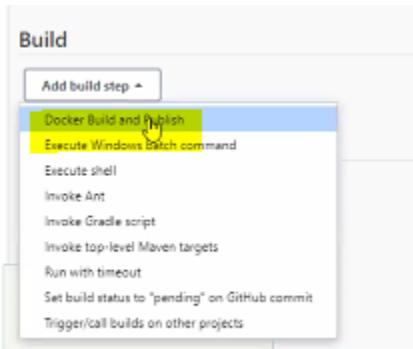
Branches to build [?](#)

Branch Specifier (blank for 'any') [?](#)

[Add Branch](#)



You will see the docker build and Publish option:



Provide the image name and tag properly:

With Ant ?

### Build

Docker Build and Publish

Repository Name ?

mdhack0316/myfire

Tag

v1

Docker Host URI ?

Server credentials

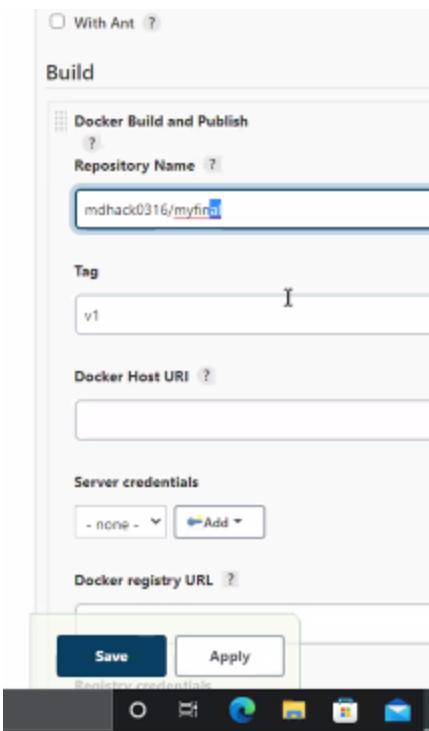
- none -

Docker registry URL ?

Save

Booted credentials

O



Add sever credentials:

### Jenkins Credentials Provider: Jenkins

Add Credentials

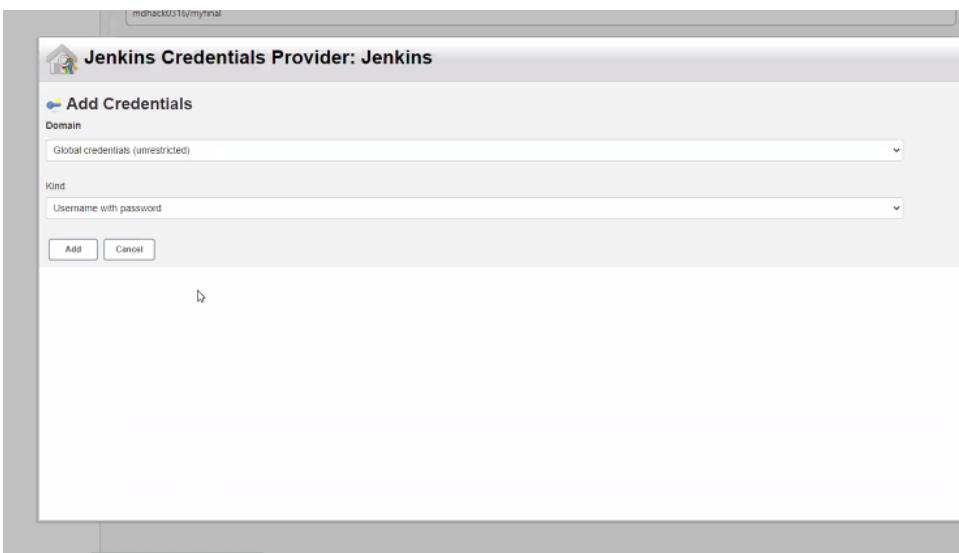
Domain

Global credentials (unrestricted)

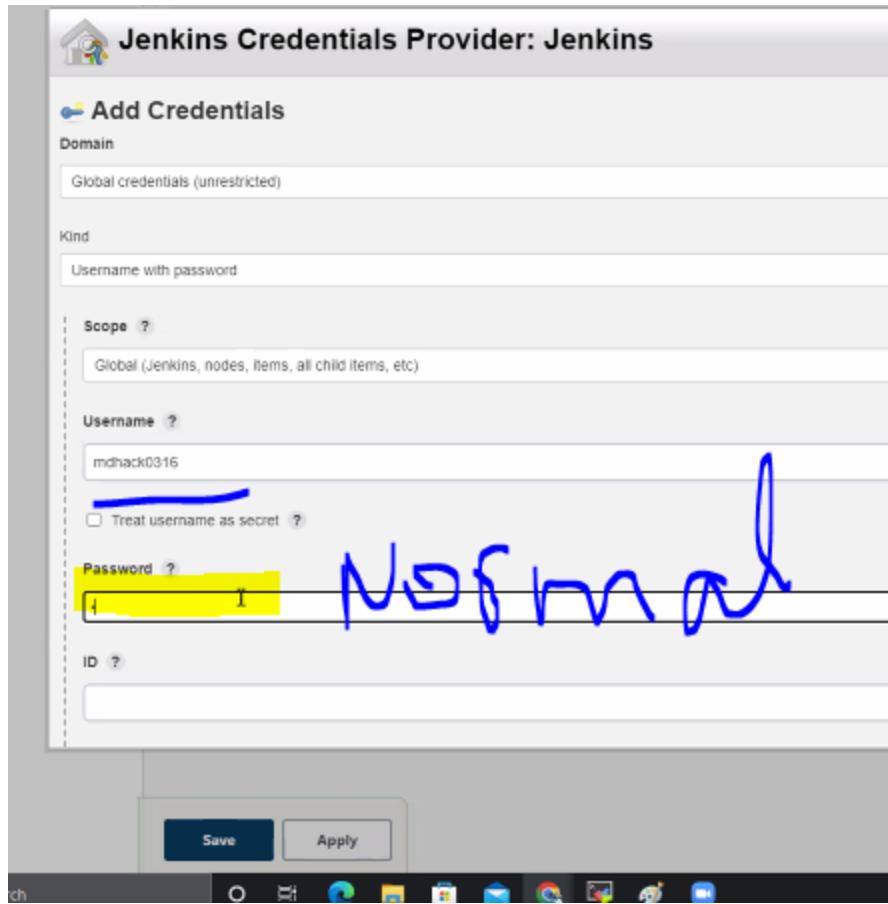
Kind

Username with password

Add Cancel



Add docker username:



Provide id, description and docker URL is not necessary:

You will see your credentials now:

With Ant ?

### Build

Docker Build and Publish

Repository Name ?  
mdhack0316/myfinal

Tag  
v1

Docker Host URI ?

Server credentials

- none -

Docker registry URL ?

Registry credentials

- none -   
- none -  
mdhack0316/\*\*\*\*\*  
mdhack0316/\*\*\*\*\*



Add the credentials while creating the new job:

Dashboard → My SAST Projects → Pushing Image to DockerHub →

General Source Code Management Build Triggers Build Environment **Build**

Docker Build and Publish

Repository Name ?  
mdhack0316/myfinal

Tag  
v1

Docker Host URI ?  
[ ]

Server credentials  
- none -

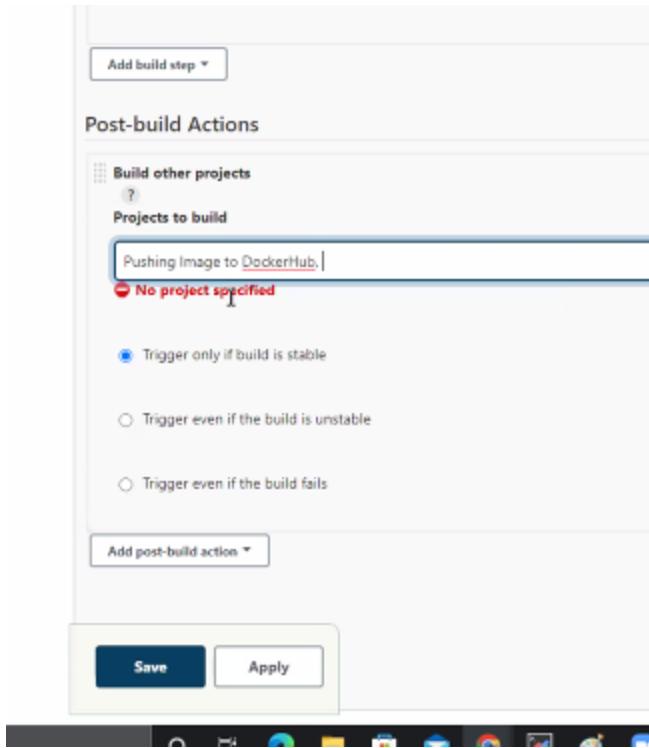
Docker registry URL ?  
[ ]

Registry credentials  
mdhack0316/\*\*\*\*\*

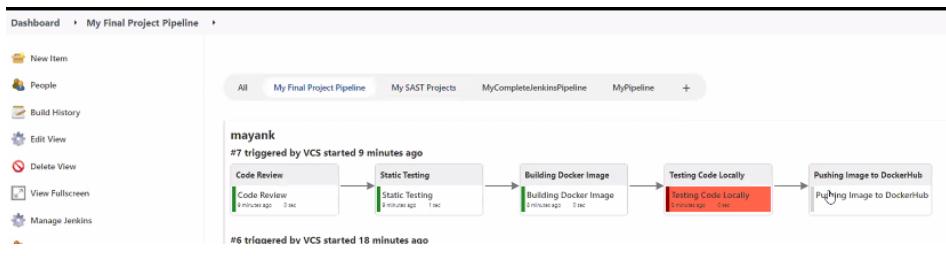
Add build step ▾

Post-build Actions

Add post build action is previous block:



Now, you can see the last step is added!



Edit the code:

```

1  # cnawebapp
2
3  ## add sample changes
4  ## 5 pipeline test
5  ## New test
6  ## New code
7  # CHANGING
8  # Something doing
9

```

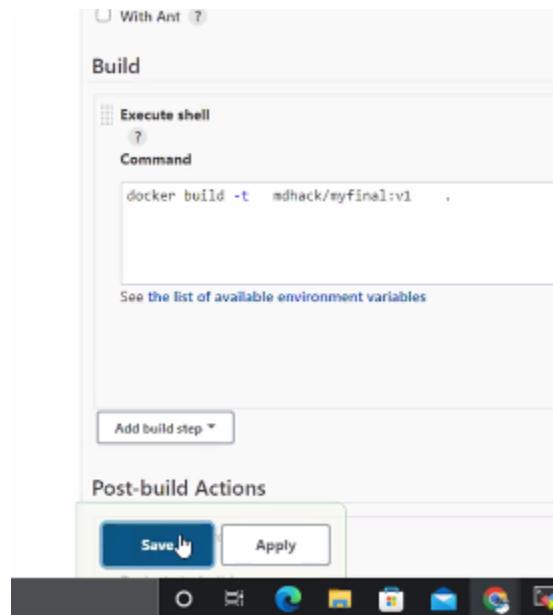
Now automatically the build will run:

The screenshot shows the Jenkins console output for a build named '#1 mdhack0316/myfinal:v1'. The output details the build process, including cloning from GitHub, fetching upstream changes, and pushing to DockerHub. A blue line highlights the error message 'unauthorized: incorrect username or password' at the end of the log.

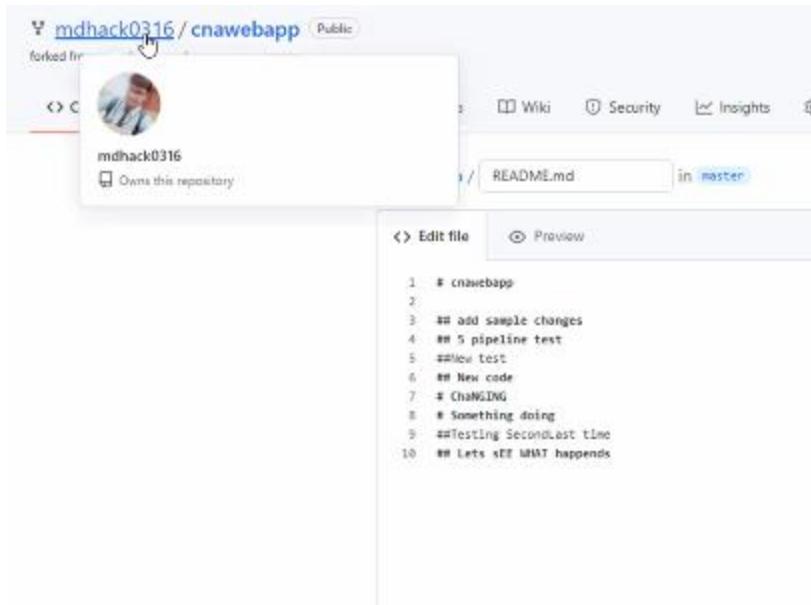
```
Started by upstream project "Testing Code Locally" build number 2
originally caused by:
Started by upstream project "Building Docker Image" build number 3
originally caused by:
Started by upstream project "Static Testing" build number 5
originally caused by:
Started by upstream project "Code Review" build number 8
originally caused by:
Started by an SCM change
Running as SYSTEM
Building in workspace /var/lib/jenkins/workspace/Pushing Image to DockerHub
The recommended git tool is: NONE
No credentials specified
Cloning the remote Git repository
Cloning repository https://github.com/mdhack0316/cnawebapp
> git init /var/lib/jenkins/workspace/Pushing Image to DockerHub # timeout=10
Fetching upstream changes from https://github.com/mdhack0316/cnawebapp
> git --version # timeout=10
> git --version # git version 2.32.0"
> git fetch -t --tags --progress -- https://github.com/mdhack0316/cnawebapp +refs/heads/*:refs/remotes/origin/* # timeout=10
> git config remote.origin.url https://github.com/mdhack0316/cnawebapp # timeout=10
> git config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
Avoid second fetch
> git rev-parse refs/remotes/origin/master^{commit} # timeout=10
Checking out Revision 8946524f1118572c33980800c1b6acd9c55ed94 (refs/remotes/origin/master)
> git config core.sparsecheckout # timeout=10
> git checkout -f 8946524f1118572c33980800c1b6acd9c55ed94 # timeout=10
Commit message: "Update README.md"
First time build. Skipping changelog.
[Pushing Image to DockerHub] $ docker build -t mdhack0316/myfinal:v1 --pull=true "/var/lib/jenkins/workspace/Pushing Image to DockerHub"
WARNING! Support for the legacy ~/.dockercfg configuration file and file-format is deprecated and will be removed in an upcoming release
Sending build context to Docker daemon 200.5kB

Step 1/4 : FROM nginx
Head "https://registry-1.docker.io/v2/library/nginx/manifests/latest": unauthorized: incorrect username or password
Build step 'Docker Build and Publish' marked build as failure
Finished: FAILURE
```

Make sure the image name starts with the username of the docker Hub:



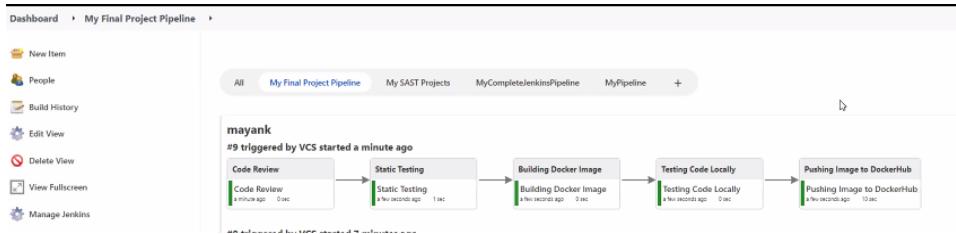
Make changes:



The screenshot shows a GitHub repository page for 'mdhack0316/cnawebapp'. The README.md file contains the following content:

```
1  # cnawebapp
2
3  ## add sample changes
4  ## 5 pipeline test
5  ##New test
6  ## New code
7  ## ChalGZNG
8  ## Something doing
9  ##Testing SecondLast time
10 ## Lets see WHAT happens
```

Now, build will automatically start:



Now, deploy it into the Kubernetes cluster:

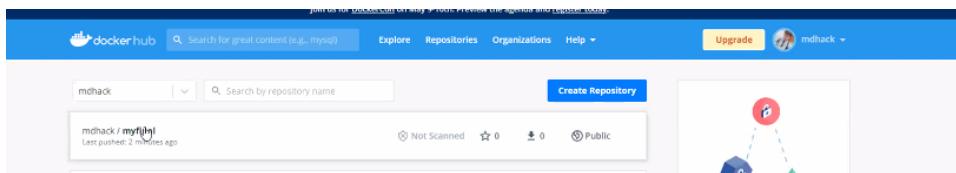
Create a new item:

The screenshot shows the Jenkins interface for creating a new project. The title bar says "Dashboard > My SAST Projects >". A modal window titled "Enter an item name" is open, with the text "Kubernetes Deploy" entered into the input field. Below the input field is a note: "Required field". The modal lists several project types with icons: "Freestyle project", "Pipeline", "Multi-configuration project", "Folder", "Multibranch Pipeline", and "Organization Folder". At the bottom of the modal, there is a note: "If you want to create a new item from other existing, you can use the 'Copy from...' button." Below the modal is a toolbar with icons for "With Ant", "Build", "Execute shell", "Command", and a search bar. The command section contains the command: "kubectl create deployment webapp --image=ndhack/myfinal:v1".

Build commands"

The screenshot shows the Jenkins build configuration for the "Execute shell" step. The title bar says "With Ant" and "Build". The "Command" field contains the command: "kubectl create deployment webapp --image=ndhack/myfinal:v1". Below the command field is a note: "See the list of available environment variables". At the bottom of the configuration are "Save" and "Apply" buttons.

Check the docker hub: you will see the image:



Are you able to access it from outside world? No

Command to expose it.

Build commands:

Build

Execute shell

?

Command

```
kubectl create deployment webapp --image=mdhack/myfinal:v1
kubectl expose deployment webapp --port 80 --type NodePort |
```

See the list of available environment variables

Save Apply

Build

Execute shell

?

Command

```
kubectl delete deployment Webapp
kubectl create deployment webapp --image=mdhack/myfinal:v1
[kubectl delete]
kubectl expose deployment webapp --port 80 --type NodePort
```

See the list of available environment variables

Save Apply

Build

Execute shell

?

Command

```
kubectl create deployment webapp --image=mdhack/myfinal:v1
[kubectl delete service]
kubectl expose deployment webapp --port 80 --type NodePort
```

See the list of available environment variables

Do it locally:

```
[root@master ~]# kubectl create deployment webapp --image=mdhack/myfinal:v1
deployment.apps/webapp created
[root@master ~]# kubectl expose deployment webapp --port 80 --type NodePort
service/webapp exposed
[root@master ~]#
[root@master ~]# kubectl ■
```

```
[root@master ~]# kubectl create deployment webapp --image=mdhack/myfinal:v1
deployment.apps/webapp created
[root@master ~]# kubectl expose deployment webapp --port 80 --type NodePort
service/webapp exposed
[root@master ~]#
[root@master ~]# kubectl get svc
error: the server doesn't have a resource type "svc"
[root@master ~]# kubectl get svc
NAME      TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
webapp   NodePort  10.103.75.68    <none>        80:31843/TCP  4s
[root@master ~]# ■
```

How to grab this service port number: by using awk command or something

```
error: the server doesn't have a resource type "svc"
[root@master ~]# kubectl get svc
NAME      TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
webapp   NodePort  10.103.75.68    <none>        80:31843/TCP  4s
[root@master ~]# kubectl get svc | grep PORT
NAME      TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
[root@master ~]# kubectl get svc | grep tcp
[root@master ~]# kubectl get svc | grep -i tcp
webapp   NodePort  10.103.75.68    <none>        80:31843/TCP  38s
[root@master ~]# ■
```

Describe command:

```
[root@master ~]# kubectl describe svc webapp
Name:                  webapp
Namespace:              mayank
Labels:                app=webapp
Annotations:            <none>
Selector:              app=webapp
Type:                  NodePort
IP Family Policy:      SingleStack
IP Families:           IPv4
IP:                    10.103.75.68
IPs:                  10.103.75.68
Port:                  <unset>  80/TCP
TargetPort:             80/TCP
NodePort:               <unset>  31843/TCP
Endpoints:              192.168.0.178:80
Session Affinity:       None
External Traffic Policy: Cluster
Events:                <none>
[root@master ~]# ■
```

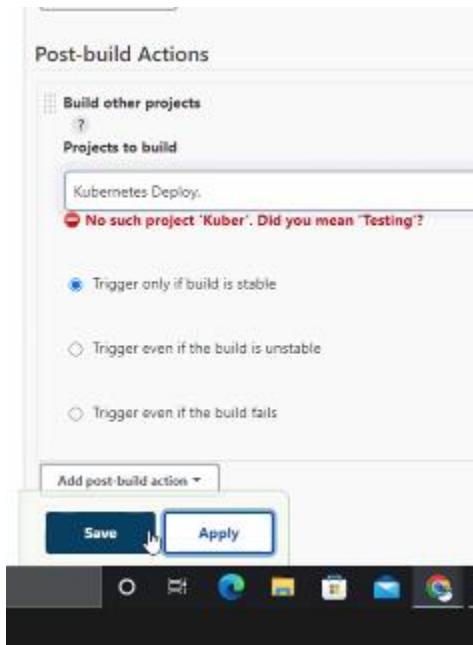
To get the port number:

```
[root@master ~]# kubectl describe svc webapp | grep -w -i nodeport
Type:           NodePort
NodePort:       <unset>  31843/TCP
[root@master ~]#
```

Delete all:

```
[root@master ~]# kubectl delete all --all --force
[warning]: Immediate deletion does not wait for confirmation that the running resource has been terminated. The resource may continue to run on the cluster indefinitely.
pod "webapp-78f7b99db9-85nq5" force deleted
service "webapp" force deleted
deployment.apps "webapp" force deleted
[root@master ~]#
```

Add post action in pushing docker image:



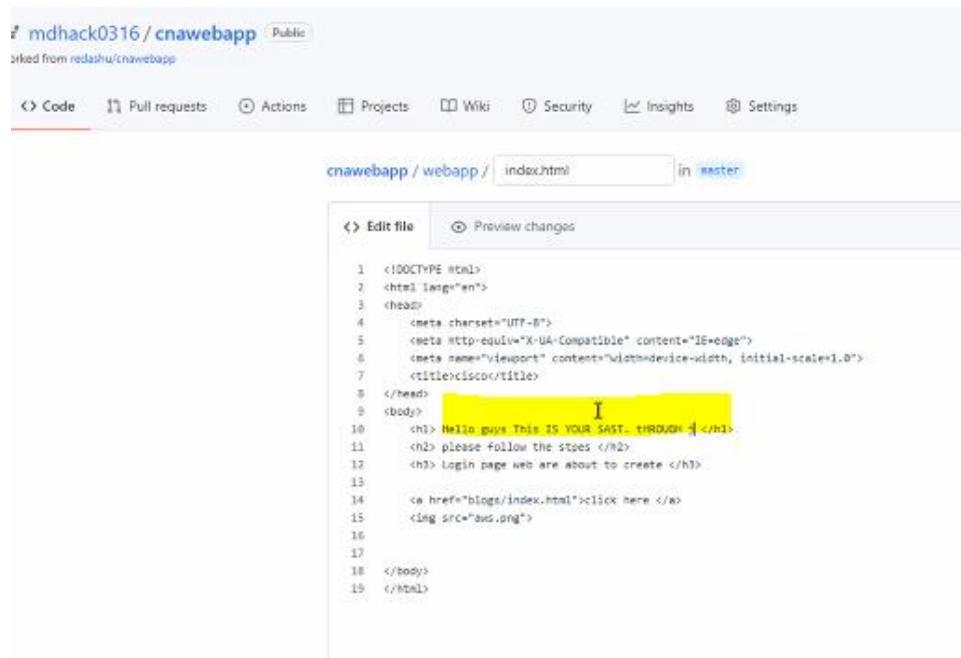
See the SAST Projects view:

S	W	Name	Last Success	Last Failure
✓	⌚	Building Docker Image	8 min 15 sec #4	N/A
✓	⌚	Code Review	8 min 35 sec #9	47 min #2
⌚	⌚	Kubernetes Deploy	N/A	N/A
✓	☁️	Pushing Image to DockerHub	7 min 55 sec #2 mdhack/myfinalsv1 mdhack/myfinallatest	13 min #1 mdhack
✓	⌚	Static Testing	8 min 25 sec #6	39 min #1
✓	👤	Testing Code Locally	8 min 5 sec #3	24 min #1

Icon: S M L

Icon legend

### Change the code:



```

# mdhack0316 / cnawebapp Public
forked from redash/cnawebapp

Code Pull requests Actions Projects Wiki Security Insights Settings

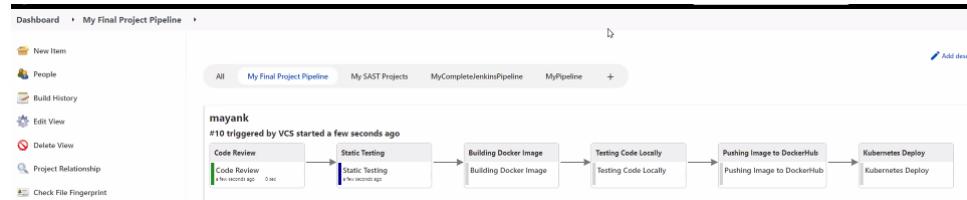
cnawebapp / webapp / index.html in master

Edit file Preview changes

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Visor</title>
8 </head>
9 <body>
10  <h1>Hello guys This IS YOUR SITE. ENJOY!</h1>
11  <h2>please follow the steps </h2>
12  <h3>Login page Web are about to create </h3>
13
14  <a href="blogs/index.html">click here </a>
15  
16
17
18 </body>
19 </html>

```

### Build will be triggered automatically:



Problem: Problem is in the delete command

The screenshot shows the Jenkins interface for a build named "Kubernetes Deploy" #1. The left sidebar has links for Back to Project, Status, Changes, and Console Output, with "Console Output" being the active tab. The main content area is titled "Console Output" with a red circle and an exclamation mark icon. It displays the following log output:

```
Started by upstream project "Pushing Image to DockerHub" build number 3
originally caused by:
Started by upstream project "Testing Code Locally" build number 4
originally caused by:
Started by upstream project "Building Docker Image" build number 5
originally caused by:
Started by upstream project "Static Testing" build number 7
originally caused by:
Started by upstream project "Code Review" build number 10
originally caused by:
Started by an SCM change
Running as SYSTEM
Building in workspace /var/lib/jenkins/workspace/Kubernetes Deploy
[Kubernetes Deploy] $ /bin/sh -xe /tmp/jenkins17894329248262072664.sh
+ kubectl delete deployment webapp
Error from server (NotFound): deployments.apps "webapp" not found
Build step 'Execute shell' marked build as failure
Finished: FAILURE
```

Change the build commands in the Kubernetes deploy block:

The screenshot shows the Jenkins build configuration for the "Kubernetes Deploy" job. Under the "Build" section, there is one step listed:

- Execute shell**

The "Command" field contains the following Jenkinsfile code:

```
kubectl create deployment webapp --image=ndhack/myfinal:v1
kubectl expose deployment webapp --port 80 --type NodePort
```

Below the command field, there is a link: "See the list of available environment variables". At the bottom of the build section, there is a "Save" button and an "Apply" button.

Edit the delivery pipeline view and enable these things:

Dashboard > My Final Project Pipeline >

2 idle

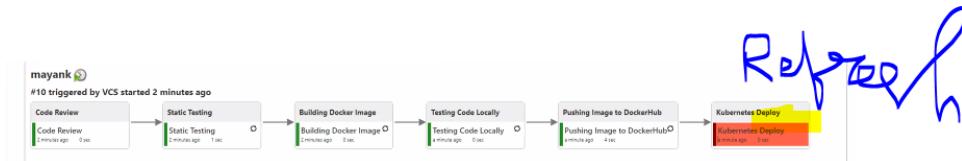
**Max number of pipelines**

**Update interval**

Enable start of new pipeline build
  Enable manual triggers
  Enable rebuild
  Allow cancelling pipeline builds
  Show avatars
  Show commit messages
  Show absolute date and time
  Show job description
  Show job promotions
  Show test results
  Show static analysis results
  Show total build time
  Use relative links for easier navigation

**OK** **Apply**

See the build now: Click on refresh on that block to make only that run!



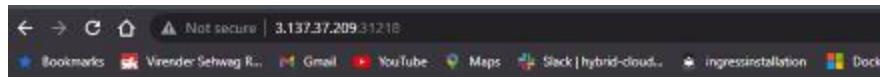
Get the port number details:

```

** resources found in mayank namespace
[root@master ~]# kubectl get svc
NAME      TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
webapp   NodePort   10.111.71.68   <none>        80:31218/TCP   1s
[root@master ~]# dock

```

Check the website:



Hello guys

please follow the steps

Login page web are about to create



[click here](#)

Updated code is not seen.

```
[root@master ~]# curl 172.17.0.2
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>cisco</title>
</head>
<body>
  <h1> Hello guys </h1>
  <h2> please follow the steps </h2>
  <h3> Login page web are about to create </h3>

  <a href="blogs/index.html">click here </a>
  

</body>
</html>[root@master ~]#
```

Check manually!

```
[root@master ~]# cd /var/lib/jenkins/workspace/Code
Code_Commit/ Code_Review/
[root@master ~]# cd /var/lib/jenkins/workspace/Code
.aws/           devsecopsjava/      JenkinsTest/        .ssh/
.cache/          docker/           .kube/            terracode/
code/           .docker/          project-html-website/ .terraform.d/
[root@master ~]# cd /var/lib/jenkins/workspace/Code
```

```

[root@master ~]# cd /var/lib/jenkins/workspace/Code
Code_Commit/ Code Review/
[root@master ~]# cd /var/lib/jenkins/workspace/Code
.aws/           devsecopsjava/ JenkinsTest/      .ssh/
.cache/          docker/        .kube/       terracode/
code/            .docker/      project-html-website/ .terraform.d/
[root@master ~]# cd /var/lib/jenkins/workspace/Code
Code_Commit/ Code Review/
[root@master ~]# cd /var/lib/jenkins/work
-bash: cd: /var/lib/jenkins/work: No such file or directory
[root@master ~]# ls
calico.yaml  devsecopsjava harsh.py  mayank.py      root@54.152.185.61
code          docker      JenkinsTest  project-html-website terracode
[root@master ~]# cd /var/lib/jenkins/workspace/
[root@master workspace]# ls
Apache Server  Code Review  Git Website  Pushing Image to DockerHub  TestingCode  UploadImage
Building Docker Image DeployApp  ImageCreation  Static Testing  Testing Code Locally
Code Commit  Docker Website Kubernetes Deploy  Testing  testproject
[root@master workspace]# rm -rf Code_Commit/
[root@master workspace]# cd Code\ Review/
[root@master Code Review]# ls
Dockerfile README.md webapp
[root@master Code Review]# cd webapp/
[root@master webapp]# ls
aws.png blogs index.html
[root@master webapp]# 

[root@master webapp]# cat index.html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>cisco</title>
</head>
<body>
  <h1> Hello guys This IS YOUR SAST. tHROUGH jENKINS </h1>
  <h2> please follow the stpes </h2>
  <h3> Login page web are about to create </h3>

  <a href="blogs/index.html">click here </a>
  

</body>
</html>
[root@master webapp]# 

```

Try to build manually now and check:

Get the port details:

```

deployment.apps webapp force deleted
[root@master workspace]# kubectl get svc
NAME      TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
webapp   NodePort  10.102.158.63  <none>        80:31033/TCP  5s
[root@master workspace]# 

```

Remove the image:

```

[4.3.197.37.209 (ec2-user)]# docker rmi -f 0d734b160056
Error response from daemon: conflict: unable to delete 0d734b160056 (cannot be forced) - image
inner 3949e2ccb3c1
[root@master workspace]# docker rm -f mayanktest1
mayanktest1
[root@master workspace]# docker rmi -f 0d734b160056
Untagged: mdhack/myfinal@sha256:adf96a3452ef32ef4259d57b703c276ca565776b50fa2edd0e19b482136ffe8
Untagged: mdhack0316/myfinal:v1
Deleted: sha256:0d734b160056e1f088b4bbc6cecb0dbe5692334de4a095692c9a27517fa51c66
Deleted: sha256:c182b3d6268315c577b9ce1123f12fabd67eb1addb8206dce0f1db40b45ee61e
Deleted: sha256:282d54091b05ec0038f688556ecbd4ae0425658f7215180c15ed2c10d3db061c
Deleted: sha256:b93be56141ba4cb0f9dc79dd618471a321a481f921081fd1dbf74bc44474ec6f
[root@master workspace]# docker images
REPOSITORY          TAG      IMAGE ID      CREATED     SIZE
newimage            latest   8dce3451f4e9  20 hours ago  144MB
mdhack0316/testwithcollins    latest   8dce3451f4e9  20 hours ago  144MB
jenkins_docker      latest   a09eacf4d706  20 hours ago  144MB
mayanktest          latest   373987aa2ad7  2 days ago   393MB
mdhack/collinsimage    latest   373987aa2ad7  2 days ago   393MB
nginx              latest   fa5269854a5e  8 days ago   142MB
httpd              latest   c30a46771695  8 days ago   144MB
calico/cni          v3.22.2  be7dfc21ba2e  2 weeks ago  236MB
calico/pod2daemon-flexvol    v3.22.2  d6660bf471e1  2 weeks ago  19.7MB
calico/node          v3.22.2  fd1608dbbc19  2 weeks ago  198MB
k8s.gcr.io/kube-apiserver    v1.23.6  8fa62c12256d  2 weeks ago  135MB
k8s.gcr.io/kube-controller-manager    v1.23.6  df7b72818ad2  2 weeks ago  125MB
k8s.gcr.io/kube-proxy          v1.23.6  4c0375452406  2 weeks ago  112MB
k8s.gcr.io/kube-scheduler        v1.23.6  595f327f224a  2 weeks ago  53.5MB
k8s.gcr.io/etcdd            3.5.1-0   25f8c7f3da61  5 months ago  293MB
k8s.gcr.io/coredns/coredns       v1.8.6   a4ca41631cc7  6 months ago  46.8MB
k8s.gcr.io/pause             3.6      6270bb605e12  8 months ago  683kB
oraclelinux           8.3      816d99f0bbe8  12 months ago  224MB
mdhack/myserver          latest   e8ce7504414a  21 months ago  350MB
quay.io/mayank123modi/simple-webapp    latest   c6e3cd9aae36  3 years ago   84.8MB
gesellix/trufflhog         latest   2488ade2fa8c  4 years ago   97MB

```

Give latest:

Build

Execute shell

Command

```
docker build -t mdhack/myfinal:latest
```

See the list of available environment variables

Add build step \*

Post-build Actions

Save Apply

Or else normally:

General Source Code Management Build Triggers Build

**Execute shell**

?

**Command**

```
docker build -t mdhack/myfinal .
```

See the list of available environment variables

Add build step ▾

**Post-build Actions**

Build other projects

?

**Projects to build**

Testing Code Locally

Trigger only if build is stable

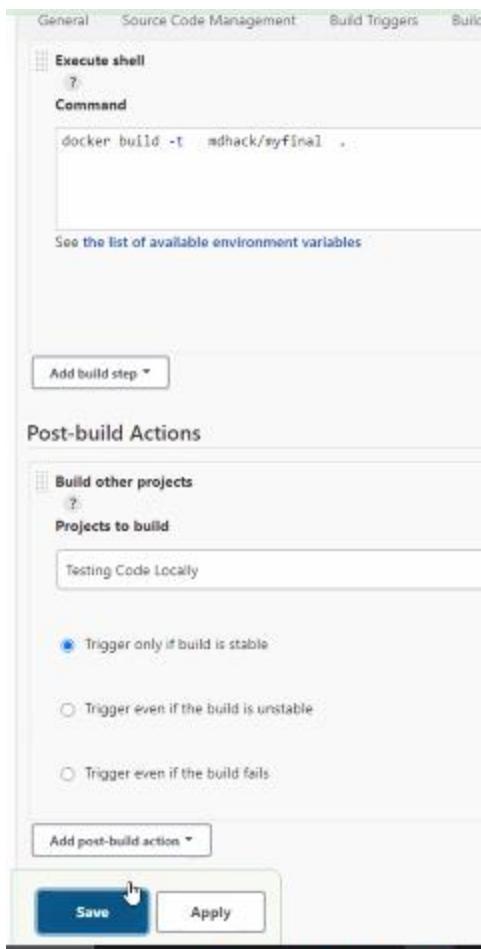
Trigger even if the build is unstable

Trigger even if the build fails

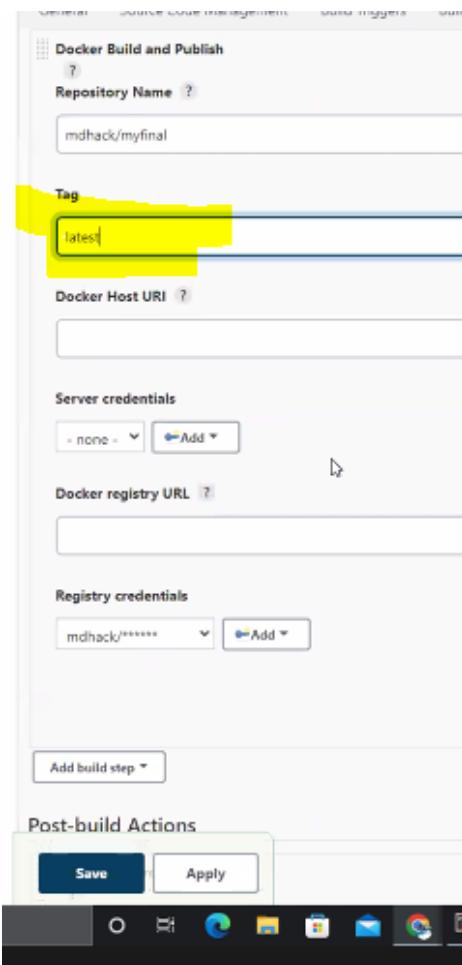
Add post-build action ▾

 Save

Apply



Change the tag:



Testing code locally – change the image name:

With Ant ?

### Build

- Execute shell
  - ?
  - Command

```
docker rm -f mayanktest1
docker run -itd --name mayanktest1 mdhack/myfinal | I
cont_ip=$(docker inspect mayanktest1 --format='{{.NetworkSettings.IPAddress}}')
curl http://$cont_ip | grep -i hello
```

See the list of available environment variables

Add build step ▾

### Post-build Actions

- Build other projects
  - ?
  - Projects to build
- Pushing Image to DockerHub

**Save** **Apply**

Curl and check – you can see the latest content:

```
curl: (7) failed to connect to 172.17.0.2 port 80 after 1450 ms. No route to host
[root@master workspace]# curl 172.17.0.2
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>cisco</title>
</head>
<body>
  <h1> Hello guys This IS YOUR SAST. tHROUGH jENKINS </h1>
  <h2> please follow the stpes </h2>
  <h3> Login page web are about to create </h3>

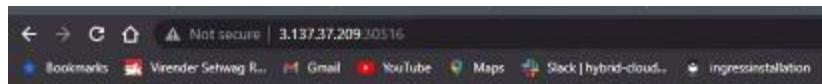
  <a href="blogs/index.html">click here </a>
  
  ...
</body>
</html>
[root@master workspace]#
```

Get the port details:

```
[root@master workspace]# kubectl get svc
NAME      TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)      AGE
webapp   NodePort   10.102.0.85  <none>        80:30516/TCP   1s
[root@master workspace]#
```

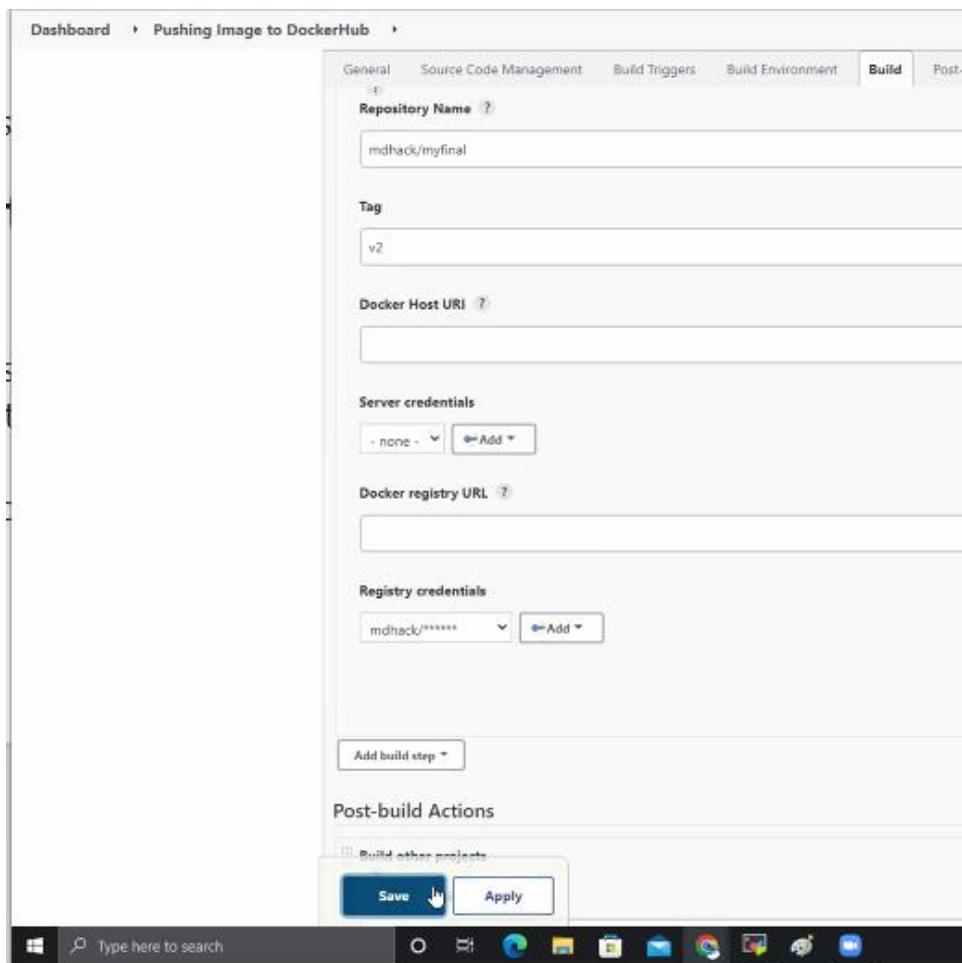
Check the website: Still, we didn't get the latest content

3.137.37.209



[click here](#)

Change the version and check:



Change the tag name in the deployment as well

**Build**

**Execute shell**

Command

```
kubectl create deployment webapp --image=ndhack/myfinal:1.0
kubectl expose deployment webapp --port 80 --type NodePort
```

See the list of available environment variables

Add build step ▾

**Post-build Actions**

Add post-build action ▾

**Save** **Apply**

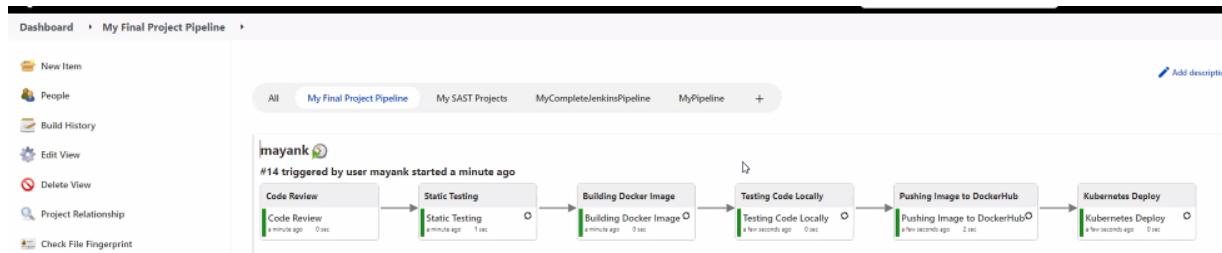
Delete the service:

```
[root@master workspace]# kubectl get all
NAME           TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)    AGE
service/webapp  NodePort   10.98.167.2 <none>        80:32077/TCP 24s
[root@master workspace]# kubectl delete svc webapp
service "webapp" deleted
[root@master workspace]# kubectl get svc
No resources found in mayank namespace.
[root@master workspace]#
```

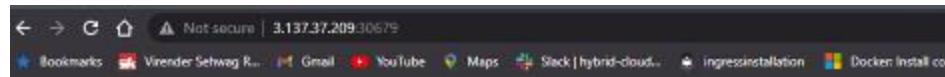
Get the IP address:

```
Every 2.0s: kubectl get svc
NAME      TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)    AGE
webapp   NodePort   10.97.33.189 <none>        80:30679/TCP 0s
```

Build now:



Check website:



Hello guys **This IS YOUR SAST. tHROUGH jENKINS**

please follow the stpes

Login page web are about to create



[click here](#)

### Problem:

**While building the use latest, while pushing use the tag/ latest**

Go with http response also: 404 – not found error:

A terminal window showing the output of a curl command. The command is 'curl 172.17.0.2/mayank.html'. The output is an HTML document with a title '404 Not Found' and a body containing '404 Not Found' and 'nginx/1.21.6'. The terminal prompt '[root@master workspace]#' is visible at the bottom.

Go with http response also: 200 means successful

```

[vishali@vishali ~]# curl -I 172.17.0.2
HTTP/1.1 200 OK
Server: nginx/1.21.6
Date: Fri, 29 Apr 2022 05:50:43 GMT
Content-Type: text/html
Content-Length: 484
Last-Modified: Fri, 29 Apr 2022 05:30:19 GMT
Connection: keep-alive
ETag: "626b77eb-1e4"
Accept-Ranges: bytes

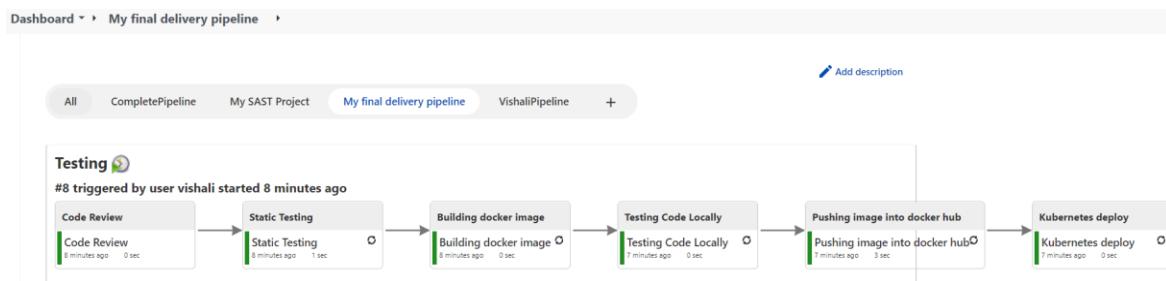
[vishali@vishali ~]#

```

TASK: Perform the above task

Fork: <https://github.com/mdhack0316/cnawebapp>

Build:



Docker Hub:

The screenshot shows the Docker Hub repository 'vishali007/myfinal'. The repository has 0 private repositories. The general tab is selected. It shows:

- Advanced Image Management: View preview
- vishali007 / myfinal: This repository does not have a description.
- Last pushed: a few seconds ago.
- Docker commands: docker push vishali007/myfinal:tagname

Get the port:

```

[vishali@vishali ~]# kubectl get svc
NAME      TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
kubernetes  ClusterIP  10.96.0.1    <none>        443/TCP      34m
webapp     NodePort   10.111.28.6  <none>        80:31411/TCP  33s

```

Check the website: Try with the Mayank IP Address. 3.137.37.209

Hello guys This IS YOUR SAST. tHROUGH jENKINS

please follow the stpes

Login page web are about to create



[click here](#)

<http://3.137.37.209:31411/>

Change the content and try:

← → ⌂ ▲ Not secure | 3.137.37.209:30910

## Hello Vishali!!

please follow the stpes

Login page web are about to create



[click here](#)

---

Image scanning:

```
[root@vishali ~]# trivy image vishali007/myfinal | head -10
2022-04-29T07:01:55.274Z      INFO  Detected OS: debian
2022-04-29T07:01:55.274Z      INFO  Detecting Debian vulnerabilities...
2022-04-29T07:01:55.296Z      INFO  Number of PL dependency files: 1
2022-04-29T07:01:55.296Z      INFO  Detecting jar vulnerabilities...

vishali007/myfinal (debian 11.3)
=====
Total: 130 (UNKNOWN: 0, LOW: 19, MEDIUM: 58, HIGH: 48, CRITICAL: 5)

+-----+-----+-----+
[root@vishali ~]#
```

If the deployment is in default namespace, this use this command:

```
[root@vishali ~]# kubectl get all -n default
NAME                                         READY   STATUS    RESTARTS   AGE
pod/kaustubhwebapp-8c5558df9-cscsh         1/1     Running   0          6m28s
pod/ramanandwebapp-7d56784ccc-2dlhk        1/1     Running   0          7m3s

NAME           TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)
AGE
service/kaustubhwebapp   NodePort   10.109.108.2   <none>       80:31355/TCP
6m28s
service/kubernetes      ClusterIP  10.96.0.1      <none>       443/TCP
7m22s
service/ramanandwebapp  NodePort   10.97.92.19   <none>       80:32267/TCP
7m3s

NAME           READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/kaustubhwebapp   1/1      1           1           6m28s
deployment.apps/ramanandwebapp  1/1      1           1           7m3s

NAME           DESIRED  CURRENT  READY   AGE
replicaset.apps/kaustubhwebapp-8c5558df9  1        1           1       6m28s
replicaset.apps/ramanandwebapp-7d56784ccc  1        1           1       7m3s
[root@vishali ~]#
```

Get service:

```
Run "sudo yum update" to apply all updates.
[root@vishali ~]# kubectl get nodes
NAME     STATUS   ROLES      AGE   VERSION
master   Ready    control-plane,master  3d2h  v1.23.6
node1    Ready    <none>      3d1h  v1.23.6
node2    Ready    <none>      3d1h  v1.23.6
node3    Ready    <none>      3d1h  v1.23.6
node4    Ready    <none>      3d1h  v1.23.6
node5    Ready    <none>      3d1h  v1.23.6
[root@vishali ~]# kubectl get service
NAME      TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
kaustubhwebapp  NodePort   10.109.45.254   <none>       80:30160/TCP  52m
kaustubhwebapp1 NodePort   10.96.118.111   <none>       80:31652/TCP  38m
kubernetes   ClusterIP  10.96.0.1      <none>       443/TCP      94m
sanwebapp3   NodePort   10.107.110.133   <none>       80:30541/TCP  92m
sanwebapp5   NodePort   10.107.252.118   <none>       80:31146/TCP  77m
ukwebapp     NodePort   10.107.160.23    <none>       80:30306/TCP  74m
vishaliwebapp NodePort   10.97.95.71    <none>       80:32230/TCP  94m
webapp       NodePort   10.111.99.148   <none>       80:31528/TCP  78m
[root@vishali ~]#
```

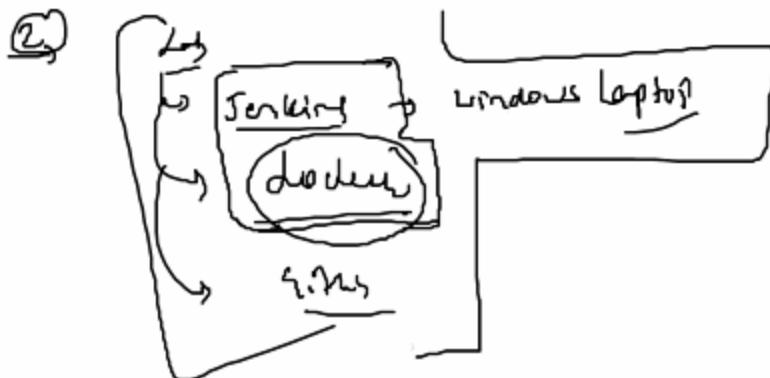
---

In Dynamic Analysis security testing, you will use trivy for image scanning

DAST:

7PM X

→ Backup

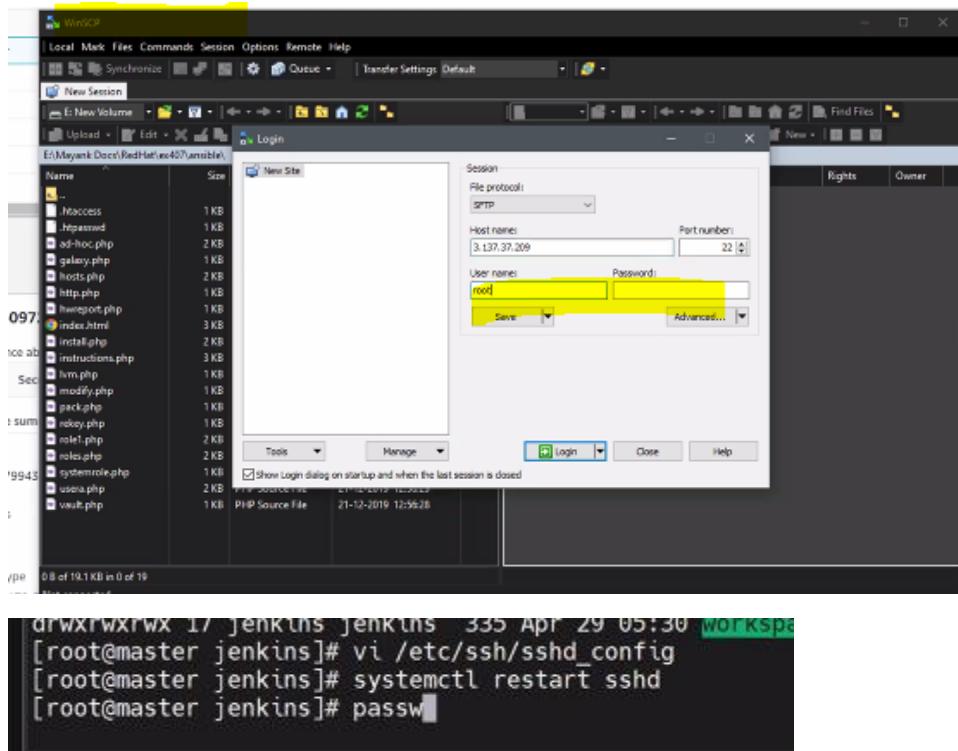


How to take the backup:

Location:

```
[root@master ~]# cd /var/lib/jenkins/
[root@master jenkins]# ls
...
com.cloudbees.hudson.plugins.folder.config.AbstractFolderConfiguration.xml
config.xml
credentials-configuration.xml
credentials.xml
fingerprints
github-plugin-configuration.xml
hudson.model.UpdateCenter.xml
hudson.plugins.build_timeout.operations.BuildStepOperation.xml
hudson.plugins.emailext.ExtendedEmailPublisher.xml
hudson.plugins.git.GitSCM.xml
hudson.plugins.git.GitTool.xml
hudson.plugins.timestamper.TimestamperConfig.xml
hudson.tasks.Mailer.xml
hudson.tasks.Shell.xml
hudson.triggers.SCMtrigger.xml
identity.key.enc
io.jenkins.plugins.junit.storage.JUnitResultStorageConfiguration.xml
jenkins.fingerprints.GlobalFingerprintConfiguration.xml
```

WINCP is the tool – help us to take the backup:



```
drwxrwxrwx 17 jenkins jenkins 335 Apr 29 05:30 workspace
[root@master jenkins]# vi /etc/ssh/sshd_config
[root@master jenkins]# systemctl restart sshd
[root@master jenkins]# passwd
```

Edit the sshd\_config

```
# but this is overridden so installations will only check .ss
AuthorizedKeysFile .ssh/authorized_keys

#AuthorizedPrincipalsFile none

# For this to work you will also need host keys in /etc/ssh/
#HostbasedAuthentication no
# Change to yes if you don't trust ~/.ssh/known_hosts for
# HostbasedAuthentication
#IgnoreUserKnownHosts no
# Don't read the user's ~/.rhosts and ~/.shosts files
#IgnoreRhosts yes

# To disable tunneled clear text passwords, change to no here
#PasswordAuthentication yes
#PermitEmptyPasswords no
PasswordAuthentication yes

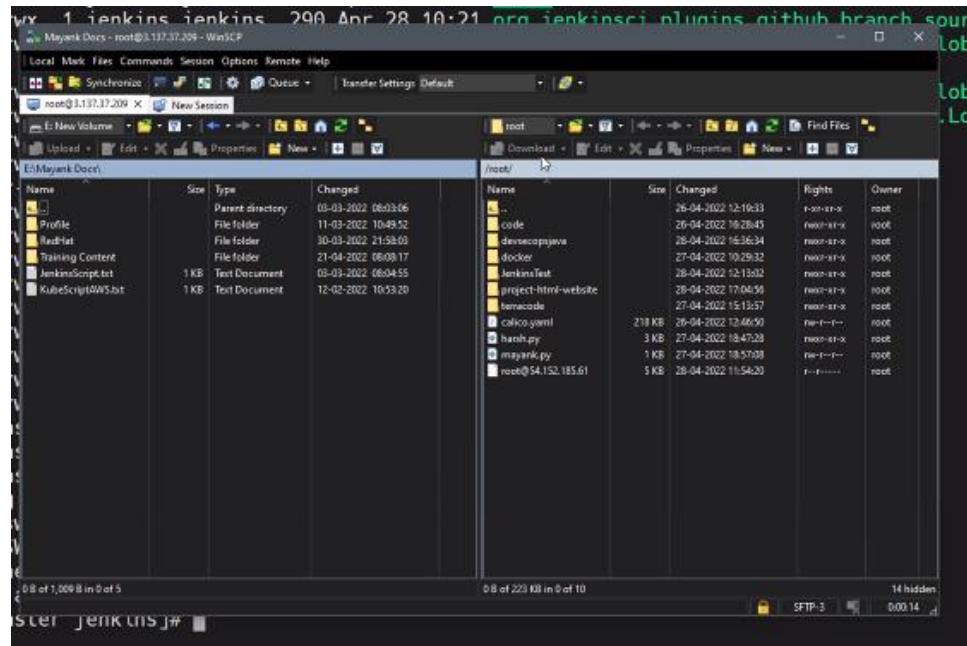
# Change to no to disable s/key passwords
#ChallengeResponseAuthentication yes
ChallengeResponseAuthentication no

# Kerberos options
#KerberosAuthentication no
#KerberosOrLocalPasswd yes
#KerberosTicketCleanup yes
#KerberosGetAFSToken no
#KerberosUseKuserok yes
```

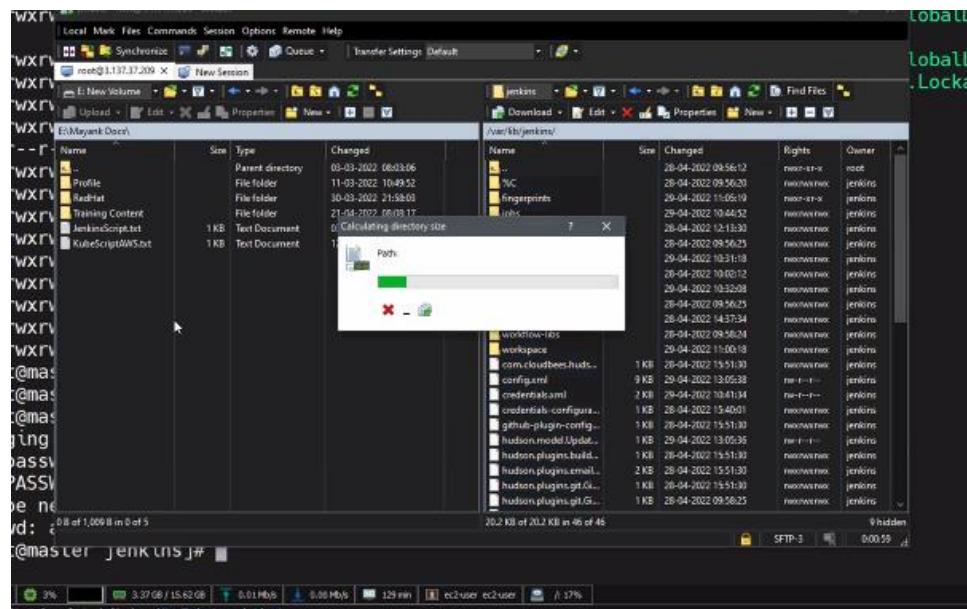
Restart the sshd:

```
drwxrwxrwx 17 jenkins jenkins 335 Apr 29 05:30 workspace
[root@master jenkins]# vi /etc/ssh/sshd_config
[root@master jenkins]# systemctl restart sshd
[root@master jenkins]# passwd
```

Go to the destination in window machine:



Select desired path in the linux machine:



Size: more space more time.



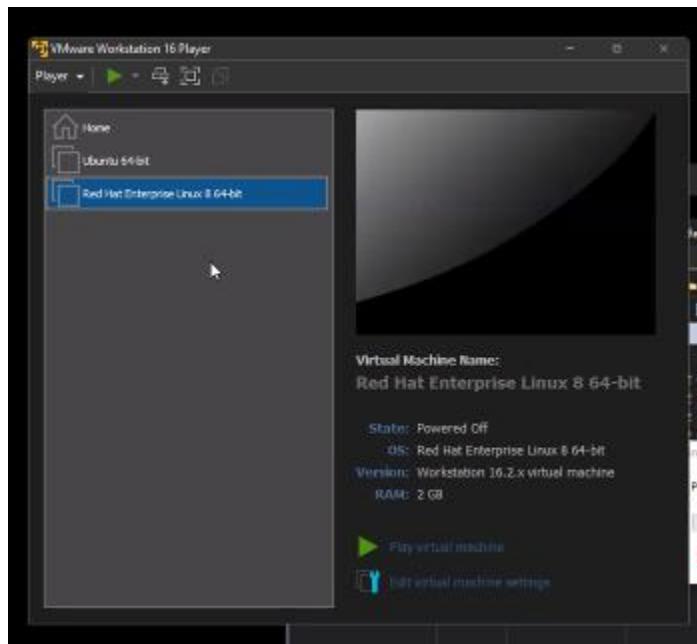
```
[root@master jenkins]# du -sch
364M
364M
[root@ma:
```

When you install the Jenkins in your machine, keep these files in that location. You will get all these in Jenkins.

Use this machine:



Trainer suggestion to use this software: vMware Workstation:



The below one consumes more RAM: Oracle VM



## Homework:

The OpenSSF CVE Benchmark consists of code and metadata for over 200 real life CVEs, as well as tooling to analyze the vulnerable codebases using a variety of static analysis security testing (SAST) tools and generate reports to evaluate those tools.



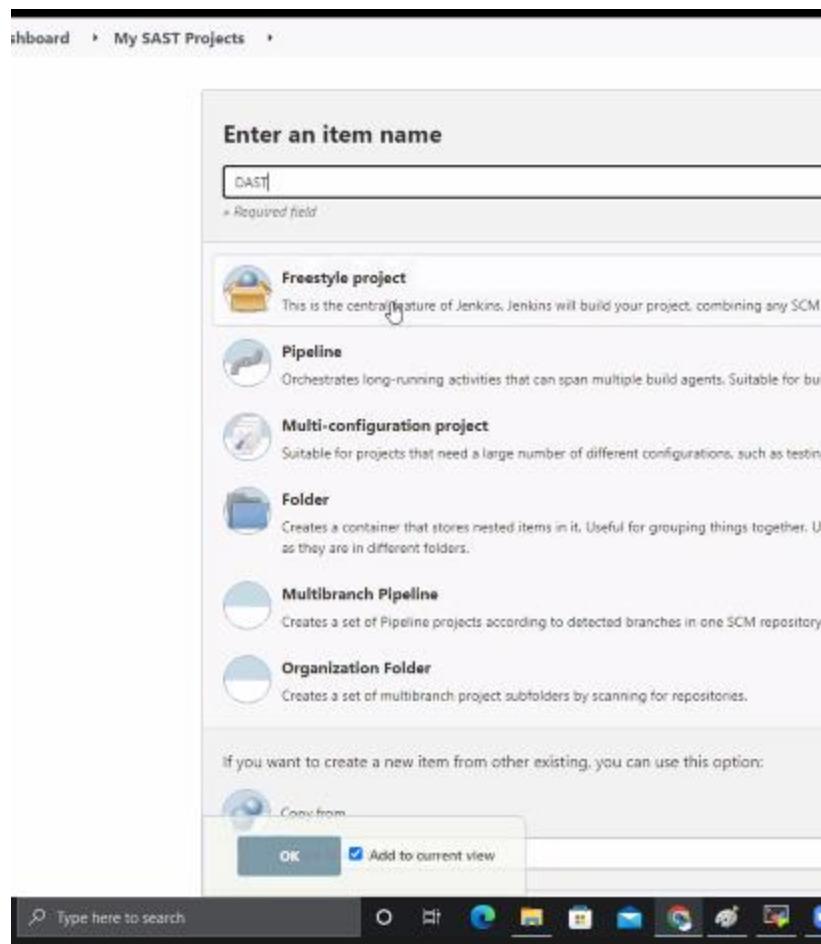
<https://github.com/ossf-cve-benchmark/ossf-cve-benchmark> ::

[Welcome to the OpenSSF CVE Benchmark project! - GitHub](#)

## DAST

```
code      docker      JenkinsTest  project-trainee-web
[root@master ~]# kubectl get deployments.apps
NAME      READY  UP-TO-DATE  AVAILABLE  AGE
webapp   1/1    1          1          4h11m
[root@master ~]# kubectl delete all --all --force
warning: Immediate deletion does not wait for confirmation.
ce may continue to run on the cluster indefinitely.
pod "webapp-7584f45968-hfl7s" force deleted
service "webapp" force deleted
deployment.apps "webapp" force deleted
[root@master ~]#
```

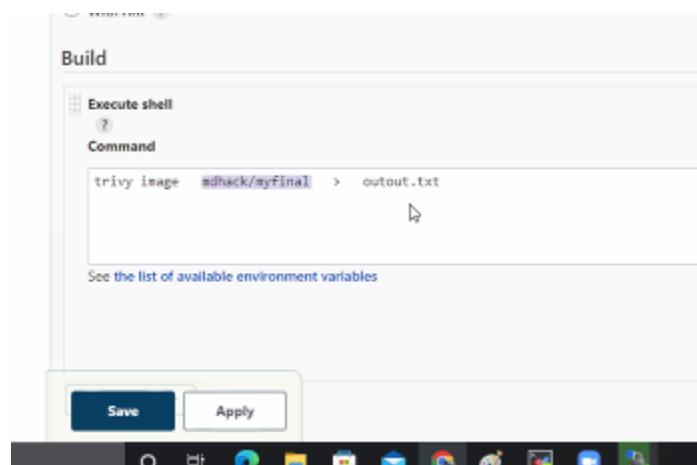
Create the item:



Check the space:

```
[root@master ~]# free -m
total        used        free      shared  buff/cache   available
Mem:       15997         3416       10790          3        1791       12335
Swap:            0           0           0
[root@master ~]#
```

Image scanning:



Definitely you will see the vulnerability

Command may be old / need to update it.

Here, we are not going to use any conditions.

The screenshot shows the 'Post-build Actions' configuration page in Jenkins. At the top, there is a button labeled 'Add build step ▾'. Below it, the 'Post-build Actions' section is titled 'Build other projects'. Under this, there is a 'Projects to build' field containing 'Kubernetes Deploy'. A red error message below the field states: 'No such project 'ku''. There are three radio buttons for triggering the build: 'Trigger only if build is stable' (selected), 'Trigger even if the build is unstable', and 'Trigger even if the build fails'. At the bottom of the configuration area, there is a 'Save' button with a mouse cursor icon hovering over it, and an 'Apply' button.

Dashboard > My SAST Projects > Pushing Image to DockerHub

General Source Code Management Build Triggers Build Environment Built-in

**Registry credentials**

mdback/\*\*\*\*\*

Add build step

**Post-build Actions**

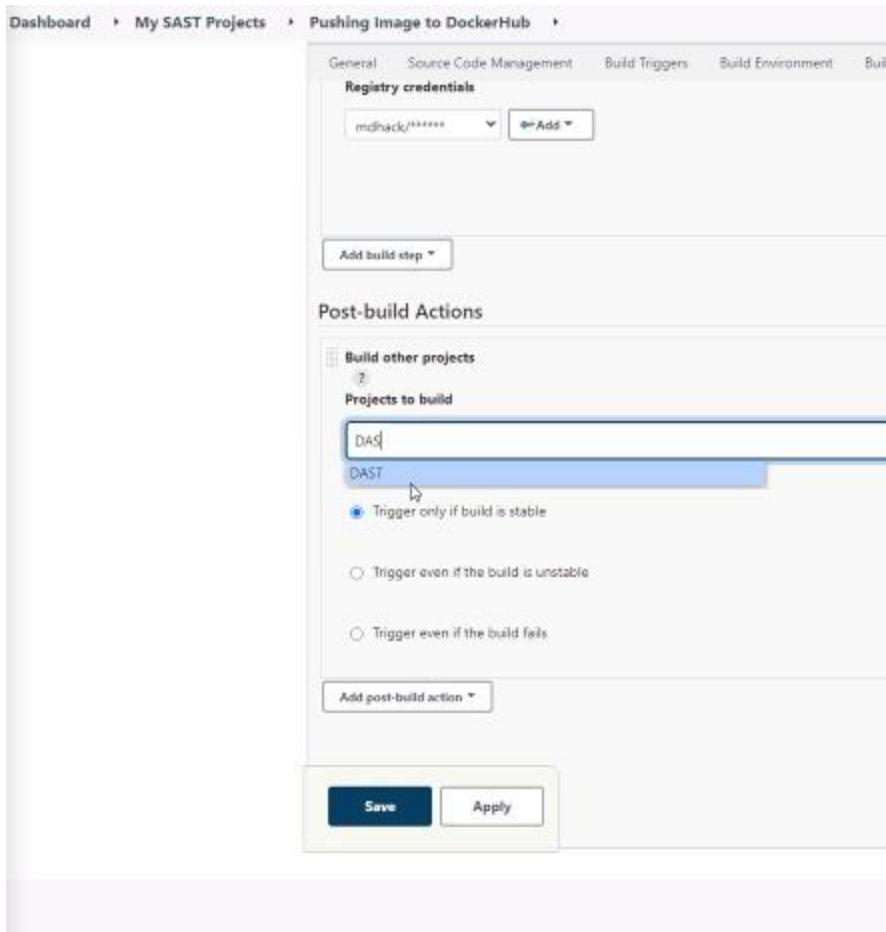
Build other projects

Trigger only if build is stable

Trigger even if the build is unstable

Trigger even if the build fails.

Add post-build action



## Plugin Manager

Updates Available Installed Advanced

Install Name:

Released

User Interface Build Tools Other Post-Build Actions

This plugin renders upstream and downstream connected jobs that typically form a build pipeline. In addition, it offers the ability to define manual triggers for jobs that require intervention prior to execution, e.g. an approval process outside of Jenkins.

Warning: This plugin version may not be safe to use. Please review the following security notices:

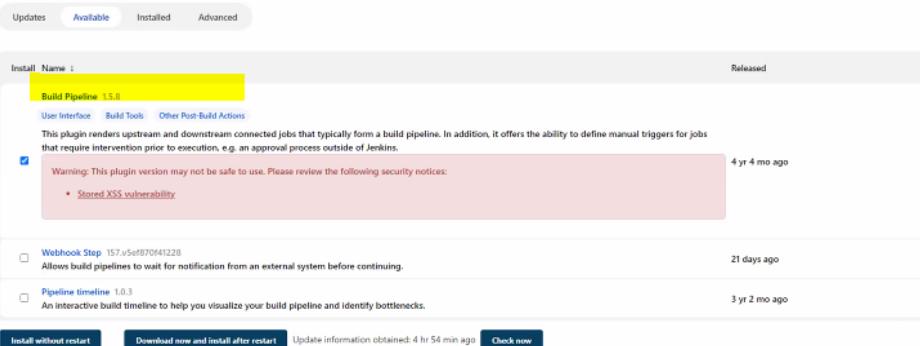
- Stored XSS vulnerability

4 yr 4 mo ago

Webhook Step 157.vSeF870441228 Allows build pipelines to wait for notification from an external system before continuing. 21 days ago

Pipeline timeline 1.0.3 An interactive build timeline to help you visualize your build pipeline and identify bottlenecks. 3 yr 2 mo ago

Update information obtained: 4 hr 54 min ago



Jenkins

Dashboard > New view

Name: FINAL

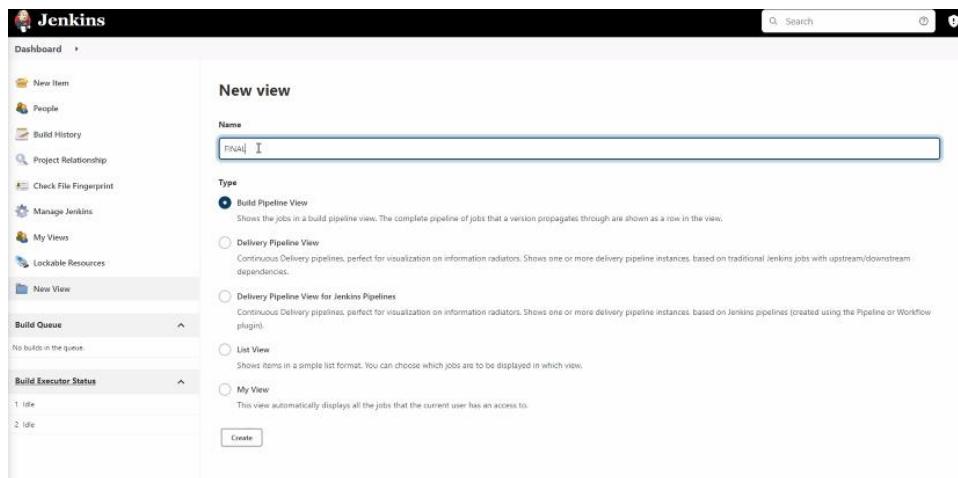
Type:

- Build Pipeline View
- Delivery Pipeline View
- Delivery Pipeline View for Jenkins Pipelines
- List View
- My View

Build Queue: No builds in the queue.

Build Executor Status: 1 idle, 2 idle.

Create



Dashboard > FINAL >

Always allow manual trigger on pipeline steps:  Yes  No

Display Options

No Of Displayed Builds: 1

Row Headers: Just the pipeline number

Show just the build pipeline number

Column Headers: No header

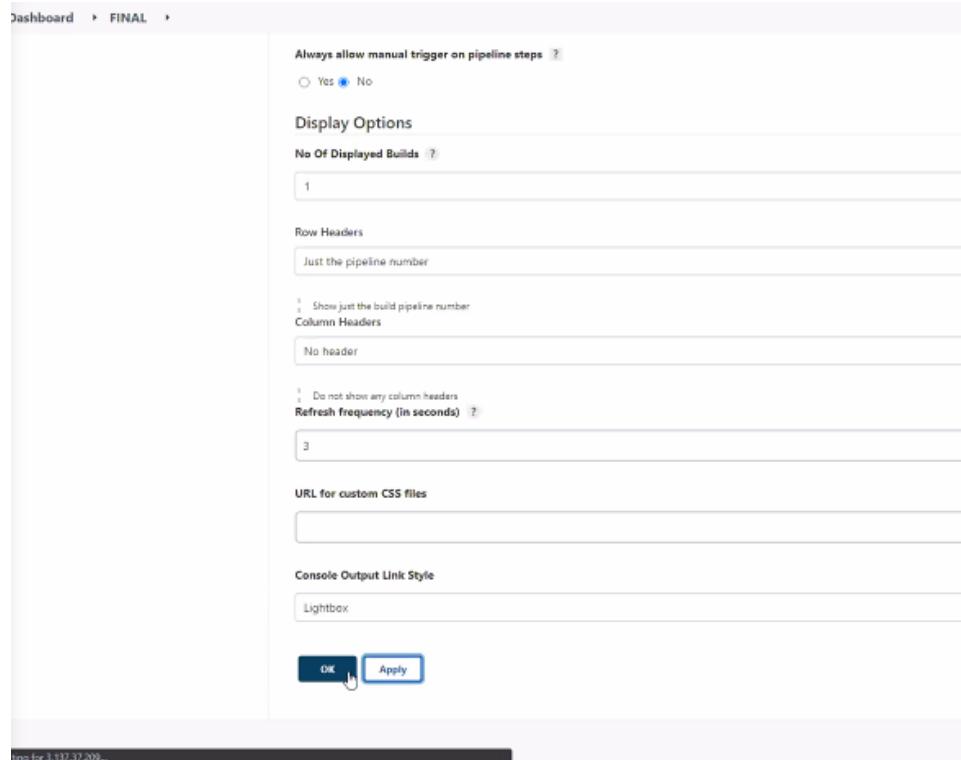
Do not show any column headers

Refresh frequency (in seconds): 3

URL for custom CSS files

Console Output Link Style: Lightbox

OK



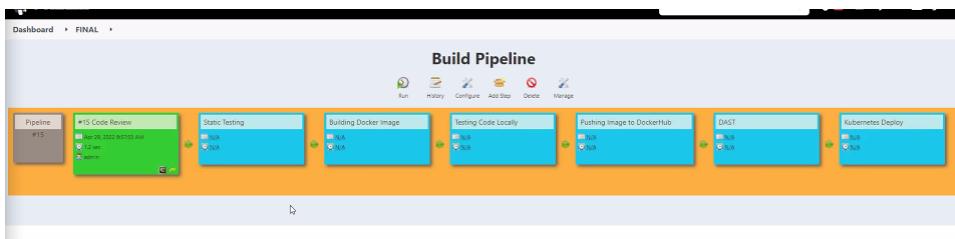
Build:

Dashboard > FINAL > Build Pipeline

Pipeline #114 Code Review #115 Static Testing #116 Building Docker Image #117 Testing Code Locally #118 pushing image to DockerHub #119 DAST #120 Kubernetes Deploy



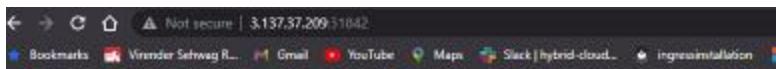
Running:



Get port details:

Kubectl get svc

Check website:



Please follow the steps

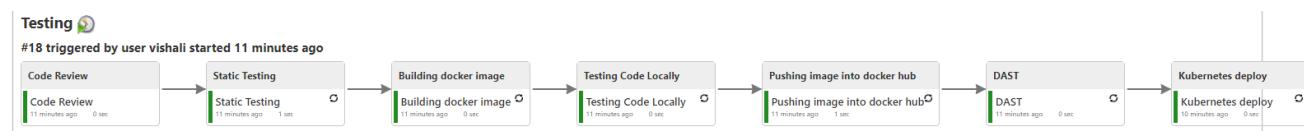
Login page web are about to create



[Click here.](#)

## TASK: Perform the above process

**Build flow:**



Get port details:

```
[root@vishali ~]# kubectl get svc -n default
NAME         TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)        AGE
kubernetes   ClusterIP 10.96.0.1    <none>        443/TCP       2m31s
vishaliwebapp   NodePort  10.99.25.204  <none>        80:31777/TCP  2m40s
webappshweta   NodePort  10.107.11.88  <none>        80:31779/TCP  56s
```

<http://3.137.37.209:31777>

Check the website:

We must use the IP address of Kubernetes master cluster.



**Hello Vishali!!**

**please follow the stpes**

Login page web are about to create



[click here](#)

---