

CICD-NODEJS-K8S-PIPELINE-APPLICATION

Prerequisites

- **Jenkins**: Installed and configured with access to GitHub, SonarQube, Docker, and Kubernetes.
- **GitHub Repository**: A repository for our Node.js application.
- **Docker**: Installed on the Jenkins server to build Docker images.
- **Kubernetes Cluster**: A functional Kubernetes cluster with at least one worker node and one master node.
- **SonarQube**: Installed and configured for Static Application Security Testing (SAST).
- **Docker Hub Account**: For storing the Docker images.

Setting Up Jenkins for CI/CD Pipeline

Installation of Java

```
sudo apt update
```

```
sudo apt install fontconfig openjdk-17-jre
```

```
java -version
```

Long Term Support release

```
sudo wget -O /usr/share/keyrings/jenkins-keyring.asc \
```

```
https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key
```

```
echo "deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc]" \
```

```
https://pkg.jenkins.io/debian-stable binary/ | sudo tee \
```

```
/etc/apt/sources.list.d/jenkins.list > /dev/null
```

```
sudo apt-get update
```

```
sudo apt-get install Jenkins
```

Launch an instance [Info](#)

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags [Info](#)

Name: Jenkins

Application and OS Images (Amazon Machine Image) [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below.

Search our full catalog including 1000s of application and OS images

Quick Start

Summary

Number of instances: 1

Software Image (AMI): Amazon Linux 2023 AMI 2023.6.2... [read more](#)
ami-08ce76ba3e392de7dc

Virtual server type (instance type): t2.micro

Firewall (security group): New security group

Storage (volumes): 1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or +2 micros in the Regions in which you launch)

Cancel **Launch instance** Preview code

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Quick Start

Search our full catalog including 1000s of application and OS images

Amazon Machine Image (AMI)

Ubuntu Server 24.04 LTS (HVM), SSD Volume Type
ami-0b20f552f63953f0e (64-bit (x86)) / ami-0dd19c76a9f10fdf9 (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs

Free tier eligible

Description: Ubuntu Server 24.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).

Canonical, Ubuntu, 24.04, amd64 noble image

Architecture: 64-bit (x86) AMI ID: ami-0b20f552f63953f0e Username: ubuntu Verified provider

Summary

Number of instances: 1

Software Image (AMI): Canonical, Ubuntu, 24.04, amd6... [read more](#)
ami-0b20f552f63953f0e

Virtual server type (instance type): t2.micro

Firewall (security group): New security group

Storage (volumes): 1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or +2 micros in the Regions in which you launch)

Cancel **Launch instance** Preview code

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Instance type [Info](#) | Get advice

Instance type: t2.medium

Family: t2 2 vCPU 4 GiB Memory Current generation: true
On-Demand Windows base pricing: 0.0788 USD per Hour
On-Demand RHEL base pricing: 0.0896 USD per Hour
On-Demand Linux base pricing: 0.0608 USD per Hour
On-Demand Ubuntu Pro base pricing: 0.0643 USD per Hour
On-Demand SUSE base pricing: 0.1608 USD per Hour

All generations Compare instance types

Additional costs apply for AMIs with pre-installed software

Key pair (login) [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required: cicd-project-pkpk [Create new key pair](#)

Network settings [Info](#)

Network: [Edit](#)

Summary

Number of instances: 1

Software Image (AMI): Canonical, Ubuntu, 24.04, amd6... [read more](#)
ami-0b20f552f63953f0e

Virtual server type (instance type): t2.medium

Firewall (security group): New security group

Storage (volumes): 1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or +2 micros in the Regions in which you launch)

Cancel **Launch instance**  McAfee | WebAdvisor Your download's being scanned. We'll let you know if there's an issue.

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

aws Services Search [Alt+S]

Description Info

launch-wizard-1 created 2024-11-07T13:41:41.603Z

Inbound Security Group Rules

- Security group rule 1 (TCP, 22, 0.0.0.0/0)

Type	Info	Protocol	Info	Port range	Info
ssh		TCP		22	
Source type	Info	Source	Info	Description - optional	Info
Anywhere		Anywhere		e.g. SSH for admin desktop	
0.0.0.0/0 <input type="button" value="X"/>					
- Security group rule 2 (TCP, 8080, 0.0.0.0/0)

Type	Info	Protocol	Info	Port range	Info
Custom TCP		TCP		8080	
Source type	Info	Source	Info	Description - optional	Info
Anywhere		Anywhere		e.g. SSH for admin desktop	
0.0.0.0/0 <input type="button" value="X"/>					

⚠️ Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting

Summary

Number of instances Info

1

Software Image (AMI)
Canonical, Ubuntu, 24.04, amd64... [read more](#)

ami-0b20f552f63953f0e

Virtual server type (instance type)
t2.medium

Firewall (security group)
New security group

Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or 12 months in the Regions in which it's available)

Cancel Preview code

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

CloudShell Feedback

aws Services Search [Alt+S]

Add security group rule

Configure storage Advanced

1x GiB gp3 Root volume (Not encrypted)

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage

Add new volume

The selected AMI contains more instance store volumes than the instance allows. Only the first 0 instance store volumes from the AMI will be accessible from the instance

Click refresh to view backup information

The tags that you assign determine whether the instance will be backed up by any Data Lifecycle Manager policies.

0 x File systems

Advanced details

Summary

Number of instances Info

1

Software Image (AMI)
Canonical, Ubuntu, 24.04, amd64... [read more](#)

ami-0b20f552f63953f0e

Virtual server type (instance type)
t2.medium

Firewall (security group)
New security group

Storage (volumes)
1 volume(s) - 30 GiB

Free tier: In your first year includes 750 hours of t2.micro (or 12 months in the Regions in which it's available)

Cancel Preview code

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

CloudShell Feedback

aws Services Search [Alt+S]

Instances (1/1)

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
Jenkins	i-029da3a48a6e57ee6	Running	t2.medium	Initializing	View alarms +	ap-northeast-1a	ec2-54-178-66-43.ap-northeast-1.compute.amazonaws.com

i-029da3a48a6e57ee6 (Jenkins)

Details Status and alarms Monitoring Security Networking Storage Tags

Instance summary

Instance ID <input type="button" value="i-029da3a48a6e57ee6"/>	Public IPv4 address <input type="button" value="54.178.66.43 open address"/>
IPv6 address -	Instance state <input type="button" value="Running"/>

Private IPv4 addresses

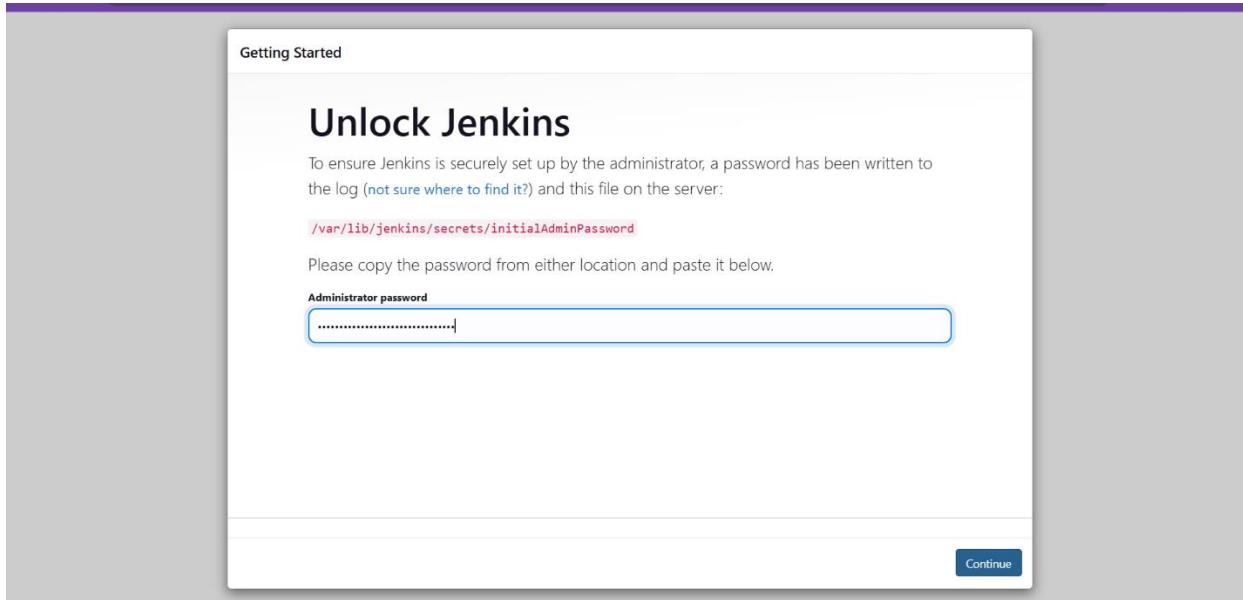
Public IPv4 DNS

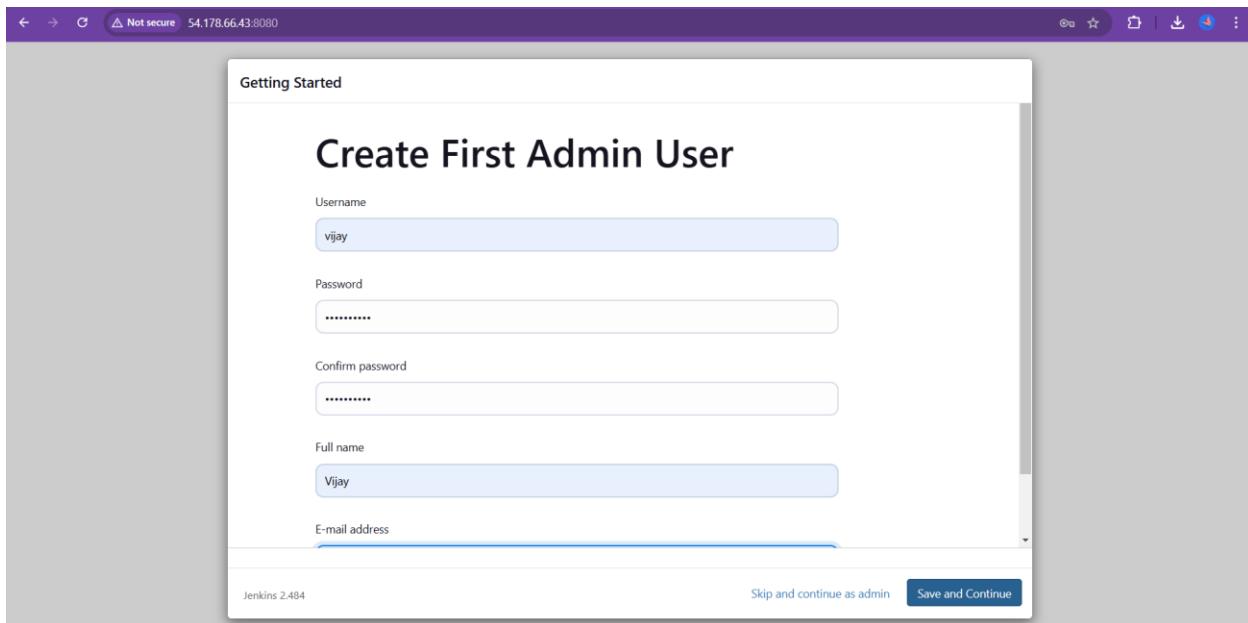
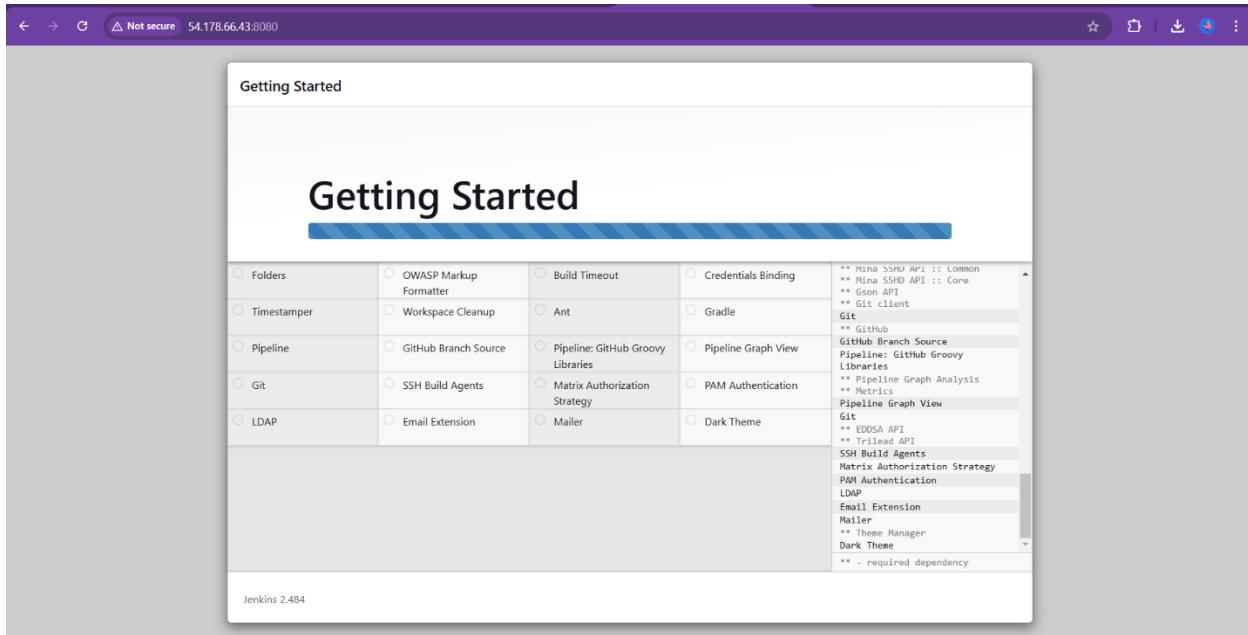
© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

CloudShell Feedback

```
root@ip-172-31-43-145:~  
login as: ubuntu  
Authenticating with public key "ciid-project-ppk"  
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-1016-aws x86_64)  
* Documentation: https://help.ubuntu.com  
* Management: https://landscape.canonical.com  
* Support: https://ubuntu.com/pro  
  
System information as of Thu Nov  7 13:44:47 UTC 2024  
  
System load: 1.09 Processes: 117  
Usage of '/': 5.5% of 28.02GB Users logged in: 0  
Memory usage: 5% IPv4 address for enX0: 172.31.43.145  
Swap usage: 0%  
  
Expanded Security Maintenance for Applications is not enabled.  
0 updates can be applied immediately.  
Enable ESM Apps to receive additional future security updates.  
See https://ubuntu.com/esm or run: sudo pro status  
  
The list of available updates is more than a week old.  
To check for new updates run: sudo apt update  
  
The programs included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*copyright.  
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
ubuntu@ip-172-31-43-145:~$ sudo -i  
root@ip-172-31-43-145:~#
```

```
root@ip-172-31-43-145:~  
root@ip-172-31-43-145:~$  
root@ip-172-31-43-145:~$  
root@ip-172-31-43-145:~$  
root@ip-172-31-43-145:~$ java -version  
openjdk version "17.0.12" 2024-07-16  
OpenJDK Runtime Environment (build 17.0.12+7-Ubuntu-ubuntu224.04)  
OpenJDK 64-Bit Server VM (build 17.0.12+7-Ubuntu-ubuntu224.04, mixed mode, sharing)  
root@ip-172-31-43-145:~$  
root@ip-172-31-43-145:~$  
root@ip-172-31-43-145:~$  
root@ip-172-31-43-145:~$  
root@ip-172-31-43-145:~$ jenkins --version  
3.48  
root@ip-172-31-43-145:~$  
root@ip-172-31-43-145:~$  
root@ip-172-31-43-145:~#
```



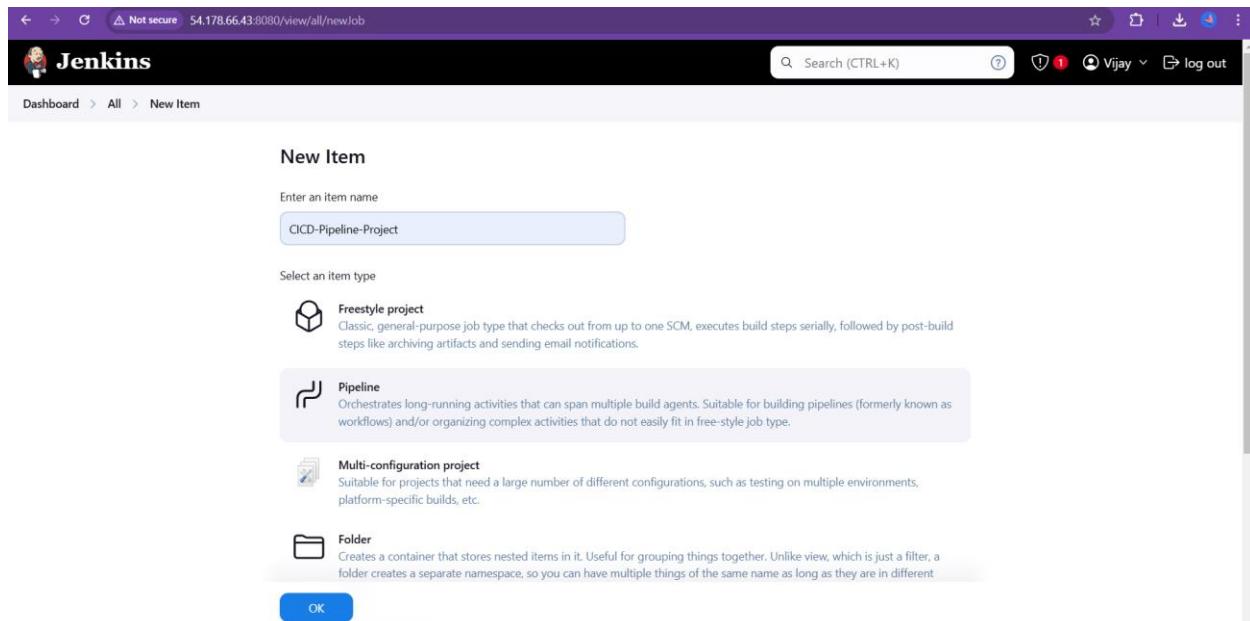


The screenshot shows the Jenkins 'Instance Configuration' page. At the top, it says 'Getting Started' and 'Instance Configuration'. Below that, there is a 'Jenkins URL' field containing 'http://54.178.66.43:8080/'. A note explains that this URL is used for absolute links to Jenkins resources and is required for proper operation. It also states that the value is proposed and not yet saved. At the bottom, it shows 'Jenkins 2.484' and two buttons: 'Not now' and 'Save and Finish'.

The screenshot shows the Jenkins Dashboard. At the top, it says 'Welcome to Jenkins!'. Below that, it says 'This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.' On the left, there are links for 'New Item', 'Build History', 'Manage Jenkins', and 'My Views'. There are also sections for 'Build Queue' (No builds in the queue) and 'Build Executor Status' (0/2). On the right, there are buttons for 'Create a job', 'Set up a distributed build', 'Set up an agent', 'Configure a cloud', and 'Learn more about distributed builds'.

Step 1: SCM Checkout (GitHub Integration)

1. Install Git Plugin in Jenkins (if not already installed).
 - o Navigate to Manage Jenkins > Manage Plugins > Available and search for Git Plugin.
 - o Install the plugin and restart Jenkins.
2. Create a New Jenkins Job:
 - o In Jenkins, create a new Pipeline Job (select Pipeline from the job type).
 - o Provide the URL of your GitHub repository and your authentication details (username/password or personal access token).



The screenshot shows a GitHub repository page for 'cicd-nodejs-k8s-pipeline'. The repository has 1 branch and 1 tag. The code section shows a single commit from 'VijayaVijayanS' titled 'Initial commit'. The commit includes files like 'Helm', 'Dockerfile', 'Jenkinsfile', 'Jenkinsfile-skip stage', 'README.md', 'buildspec.yml', 'docker-compose.yml', 'package.json', 'rocket-app.yml', and 'server.js'. The 'Code' dropdown menu is open, showing options for cloning the repository via HTTPS, SSH, or GitHub CLI, along with links to copy the URL to clipboard and open it in GitHub Desktop or download a ZIP file. The repository has 0 stars, 1 watching, and 0 forks. The 'About' section describes the project's focus on continuous integration/continuous deployment (CI/CD) for a Node.js application using Kubernetes (k8s) for deployment.

The screenshot shows the Jenkins interface for configuring a pipeline. The top navigation bar indicates the URL as 54.178.66.43:8080/job/CICD-Pipeline-Project/configure. The left sidebar has tabs for General, Build Triggers, Advanced Project Options, and Pipeline, with Pipeline being the active tab. The main content area is titled 'Configure' and shows the 'Definition' section set to 'Pipeline script'. A code editor displays the Groovy pipeline script:

```
1 * pipeline {
2     agent any
3     stages {
4         stage('SCM checkout') {
5             steps {
6                 script {
7                     git url: 'https://github.com/VijayaVijayanS/cicd-nodejs-k8s-pipeline.git'
8                 }
9             }
10        }
11    }
12 }
```

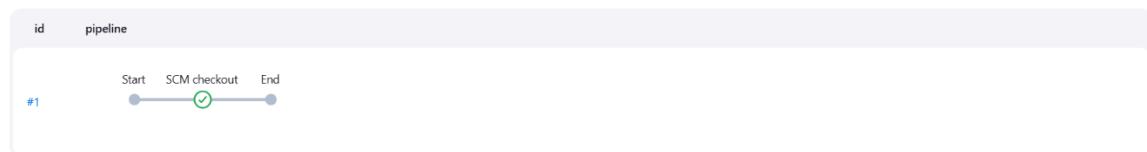
Below the code editor is a checkbox labeled 'Use Groovy Sandbox' which is checked. At the bottom are 'Save' and 'Apply' buttons, and a green footer bar indicating the configuration has been 'Saved'.

← → C Not secure 54.178.66.43:8080/job/CICD-Pipeline-Project/multi-pipeline-graph/ Jenkins

Dashboard > CICD-Pipeline-Project > Stages

Build CICD-Pipeline-Project

Build Configure



Jenkins 2.484

← → C Not secure 54.178.66.43:8080/job/CICD-Pipeline-Project/1/console Jenkins

Dashboard > CICD-Pipeline-Project > #1

Replay Pipeline Steps Workspaces

```
Fetching upstream changes from https://github.com/VijayaVijayans/cicd-nodejs-k8s-pipeline.git
> git --version # timeout=10
> git --version # 'git version 2.43.0'
> git fetch --tags --force --progress -- https://github.com/VijayaVijayans/cicd-nodejs-k8s-pipeline.git +refs/heads/*:refs/remotes/origin/*
timeout=10
> git config remote.origin.url https://github.com/VijayaVijayans/cicd-nodejs-k8s-pipeline.git # 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 02f8327750e154ebfd3e4a54175e38eedf37dd5a (refs/remotes/origin/master)
> git config core.sparsecheckout # timeout=10
> git checkout -f 02f8327750e154ebfd3e4a54175e38eedf37dd5a # timeout=10
> git branch -a -v --no-abbrev # timeout=10
> git checkout -b master 02f8327750e154ebfd3e4a54175e38eedf37dd5a # timeout=10
Commit message: "Initial commit"
First time build. Skipping changelog.
[Pipeline] 
[Pipeline] // script
[Pipeline] 
[Pipeline] // stage
[Pipeline] 
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```

REST API Jenkins 2.484

Step 2: SonarQube Analysis (SAST)

Step 1 : Installing Docker Compose

```
sudo curl -L "https://github.com/docker/compose/releases/download/1.29.2/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
```

```
sudo chmod +x /usr/local/bin/docker-compose
```

```
docker-compose --version
```

Step 2 : Setting Up a docker-compose.yml File

```
vi docker-compose.yaml
```

Docker Compose file:

```
version: "3"
```

```
services:
```

```
    sonarqube:
```

```
        image: sonarqube:community
```

```
        container_name: sonarqube
```

```
        depends_on:
```

```
            - db
```

```
environment:
```

```
    SONAR_JDBC_URL: jdbc:postgresql://db:5432/sonar
```

```
    SONAR_JDBC_USERNAME: sonar
```

```
    SONAR_JDBC_PASSWORD: sonar
```

```
volumes:
```

```
    - sonarqube_data:/opt/sonarqube/data
```

- sonarqube_extensions:/opt/sonarqube/extensions
- sonarqube_logs:/opt/sonarqube/logs

ports:

- "9000:9000"

restart: always

db:

image: postgres:12

container_name: postgres12

environment:

POSTGRES_USER: sonar

POSTGRES_PASSWORD: sonar

volumes:

- postgresql:/var/lib/postgresql
- postgresql_data:/var/lib/postgresql/data

restart: always

volumes:

sonarqube_data:

sonarqube_extensions:

sonarqube_logs:

postgresql:

postgresql_data:

Launch an instance

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags

Name: SonarQube & DB

Application and OS Images (Amazon Machine Image)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below.

Search our full catalog including 1000s of application and OS images

Recents | Quick Start

Summary

Number of instances: 1

Software Image (AMI): Amazon Linux 2023.6.2...read more
ami-08ce76bea592de7dc

Virtual server type (instance type): t2.micro

Firewall (security group): New security group

Storage (volumes): 1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or +2 micro in the Regions in which

Cancel | Launch instance | Preview code

Amazon Machine Image (AMI)

Ubuntu Server 22.04 LTS (HVM), SSD Volume Type

Free tier eligible

Ubuntu Server 22.04 LTS (HVM), SSD General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).

Description

Canonical, Ubuntu, 22.04 LTS, amd64 jammy image build on 2024-09-27

Architecture: 64-bit (x86) | AMI ID: ami-0ac6b9b2908f5e20d | Username: ubuntu | Verified provider

Summary

Number of instances: 1

Software Image (AMI): Canonical, Ubuntu, 22.04 LTS, ...read more
ami-0ac6b9b2908f5e20d

Virtual server type (instance type): t2.micro

Firewall (security group): New security group

Storage (volumes): 1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or +2 micro in the Regions in which

Cancel | Launch instance | Preview code

Instance type

t2.medium

Family: t2. 2 vCPU 4 GiB Memory Current generation: true
On-Demand Windows base pricing: 0.0788 USD per Hour
On-Demand RHEL base pricing: 0.0896 USD per Hour
On-Demand Linux base pricing: 0.0608 USD per Hour
On-Demand Ubuntu Pro base pricing: 0.0643 USD per Hour
On-Demand SUSE base pricing: 0.1608 USD per Hour

All generations | Compare instance types

Additional costs apply for AMIs with pre-installed software

Key pair (login)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required: cicd-project-ppk

Create new key pair

Network settings

Network: Default

Summary

Number of instances: 1

Software Image (AMI): Canonical, Ubuntu, 22.04 LTS, ...read more
ami-0ac6b9b2908f5e20d

Virtual server type (instance type): t2.medium

Firewall (security group): New security group

Storage (volumes): 1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or +2 micro in the Regions in which

Cancel | Launch instance | Preview code

Screenshot of the AWS IAM Security Groups configuration page.

Security group rule 1 (TCP, 22, 0.0.0.0/0)

- Type: ssh
- Protocol: TCP
- Port range: 22
- Source type: Anywhere
- Description: e.g. SSH for admin desktop

Security group rule 2 (TCP, 9000, 0.0.0.0/0)

- Type: Custom TCP
- Protocol: TCP
- Port range: 9000
- Source type: Anywhere
- Description: e.g. SSH for admin desktop

Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Add security group rule

Summary

- Number of instances: 1
- Software Image (AMI): Canonical, Ubuntu, 22.04 LTS, ... (ami-0ac6b9b2908f5e20d)
- Virtual server type (instance type): t2.medium
- Firewall (security group): New security group
- Storage (volumes): 1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or 12 months in this Duration in which)

Launch instance

Screenshot of the AWS IAM Security Groups configuration page.

Add security group rule

Configure storage

- Root volume (Not encrypted): 1x 30 GiB gp2

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage

Add new volume

The selected AMI contains more instance store volumes than the instance allows. Only the first 0 instance store volumes from the AMI will be accessible from the instance

Click refresh to view backup information

The tags that you assign determine whether the instance will be backed up by any Data Lifecycle Manager policies.

Advanced details

Summary

- Number of instances: 1
- Software Image (AMI): Canonical, Ubuntu, 22.04 LTS, ... (ami-0ac6b9b2908f5e20d)
- Virtual server type (instance type): t2.medium
- Firewall (security group): New security group
- Storage (volumes): 1 volume(s) - 30 GiB

Free tier: In your first year includes 750 hours of t2.micro (or 12 months in this Duration in which)

Launch instance

Screenshot of the AWS EC2 Instances management page.

Instances (1/2)

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
SonarQube & DB	i-056545535c487b39f	Running	t2.medium	Initializing	View alarms +	ap-northeast-1a	ec2-18-183-182-211.ap-northeast-1.compute.amazonaws.com
Jenkins	i-029da3a48a6e57ee6	Running	t2.medium	2/2 checks passed	View alarms +	ap-northeast-1a	ec2-54-197-142-149.ap-northeast-1.compute.amazonaws.com

i-056545535c487b39f (SonarQube & DB)

Details

- Instance ID: i-056545535c487b39f
- Public IPv4 address: 18.183.182.211 | [Open address](#)
- Private IPv4 addresses: 172.31.32.92
- Public IPv4 DNS: ec2-18-183-182-211.ap-northeast-1.compute.amazonaws.com

Actions

Launch instances

```
root@ip-172-31-32-92:~  
login as: ubuntu  
Authenticating with public key "cicd-project-ppk"  
Welcome to Ubuntu 22.04.5 LTS (GNU/Linux 6.8.0-1015-aws x86_64)  
* Documentation: https://help.ubuntu.com  
* Management: https://landscape.canonical.com  
* Support: https://ubuntu.com/pro  
  
System information as of Thu Nov 7 14:25:45 UTC 2024  
  
System load: 0.33 Processes: 118  
Usage of /: 5.5% of 28.89GB Users logged in: 0  
Memory usage: 5% IPv4 address for eth0: 172.31.32.92  
Swap usage: 0%  
  
Expanded Security Maintenance for Applications is not enabled.  
0 updates can be applied immediately.  
Enable ESM Apps to receive additional future security updates.  
See https://ubuntu.com/esm or run: sudo pro status  
  
The list of available updates is more than a week old.  
To check for new updates run: sudo apt update  
  
The programs included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*copyright*.  
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
ubuntu@ip-172-31-32-92:~$ sudo -i  
root@ip-172-31-32-92:~#
```

```
root@ip-172-31-32-92:~  
root@ip-172-31-32-92:~#  
root@ip-172-31-32-92:~# docker  
Command 'docker' not found, but can be installed with:  
apt install podman-docker # version 3.4.4+ds1-ubuntu1.22.04.2, or  
apt install dockerio # version 24.0.7-0ubuntu2-22.04.1  
root@ip-172-31-32-92:~# apt install docker.io  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
The following additional packages will be installed:  
bridge-utils containerd dns-root-data dnsmasq-base pigz runc ubuntu-fan  
Suggested packages:  
ifupdown aufs-tools cgroupsfs-mount | cgroup-lite debootstrap docker-darling rinse zfs-fuse | zfsutils  
The following NEW packages will be installed:  
bridge-util containerd dns-root-data dnsmasq-base docker.io pigz runc ubuntu-fan  
0 upgraded, 8 newly installed, 0 to remove and 16 not upgraded.  
Need to get 75.5 MB of archives.  
After this operation, 284 kB of additional disk space will be used.  
Do you want to continue? [Y/n] Y  
Get:1 http://ap-northeast-1.ec2.archive/ubuntu jammy/universe amd64 pigz amd64 2.6-1 [63.6 kB]  
Get:2 http://ap-northeast-1.ec2.archive/ubuntu.com/ubuntu jammy/main amd64 bridge-utils amd64 1.7-lubuntu3 [34.4 kB]  
Get:3 http://ap-northeast-1.ec2.archive/ubuntu.com/ubuntu jammy-updates/main amd64 runc amd64 1.1.12-0ubuntu2-22.04.1 [8405 kB]  
Get:4 http://ap-northeast-1.ec2.archive/ubuntu.com/ubuntu jammy-updates/main amd64 containerd amd64 1.7.12-0ubuntu2-22.04.1 [37.8 kB]  
Get:5 http://ap-northeast-1.ec2.archive/ubuntu.com/ubuntu jammy-updates/main amd64 dns-root-data all 2023112702~ubuntu0.22.04.1 [5136 B]  
Get:6 http://ap-northeast-1.ec2.archive/ubuntu.com/ubuntu jammy-updates/main amd64 dnsmasq-base amd64 2.90-0ubuntu0.22.04.1 [374 kB]  
Get:7 http://ap-northeast-1.ec2.archive/ubuntu.com/ubuntu jammy-updates/universe amd64 docker.io amd64 24.0.7-0ubuntu2-22.04.1 [28.8 MB]  
Get:8 http://ap-northeast-1.ec2.archive/ubuntu.com/ubuntu jammy/universe amd64 ubuntu-fan all 0.12.16 [35.2 kB]  
Fetched 75.5 MB in 1s (72.5 MB/s)
```

```
root@ip-172-31-32-92:~  
root@ip-172-31-32-92:~#  
root@ip-172-31-32-92:~# sudo curl -L "https://github.com/docker/compose/releases/download/1.29.2/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose  
% Total % Received % Xferd Average Speed Time Time Current  
Dload Upload Total Spent Left Speed  
0 0 0 0 0 0 0 ---:--- ---:--- ---:--- 0  
100 12.1M 100 12.1M 0 0 8652k 0 0:00:01 0:00:01 ---:--- 14.5M  
root@ip-172-31-32-92:~#  
root@ip-172-31-32-92:~#  
root@ip-172-31-32-92:~# sudo chmod +x /usr/local/bin/docker-compose  
root@ip-172-31-32-92:~#  
root@ip-172-31-32-92:~#  
root@ip-172-31-32-92:~#  
root@ip-172-31-32-92:~# docker-compose --version  
docker-compose version 1.29.2, build 5becea4c  
root@ip-172-31-32-92:~#  
root@ip-172-31-32-92:~#  
root@ip-172-31-32-92:~# vi docker-compose.yaml  
root@ip-172-31-32-92:~#  
root@ip-172-31-32-92:~#  
root@ip-172-31-32-92:~#
```

```

root@ip-172-31-32-92:~#
root@ip-172-31-32-92:~# docker-compose --version
docker-compose version 1.29.2, build 5becea4c
root@ip-172-31-32-92:~#
root@ip-172-31-32-92:~# vi docker-compose.yaml
root@ip-172-31-32-92:~#
root@ip-172-31-32-92:~# docker-compose up -d
Creating network "root_default" with the default driver
Creating volume "root_sonarqube_data" with default driver
Creating volume "root_sonarqube_extensions" with default driver
Creating volume "root_sonarqube_logs" with default driver
Creating volume "root_postgresql" with default driver
Creating volume "root_postgresql_data" with default driver
Pulling db (postgres:12)...
12: Pulling from library/postgres
a490a496ba95: Pull complete
65de630e0c02: Pull complete
733f3a2a0a08: Pull complete
e26ca1e19fb8: Pull complete
9f88a44502eb: Pull complete
15379595632b7: Pull complete
881a8d18d015: Pull complete
8ad9c6393a05: Pull complete
8e958e82aa0a: Pull complete
a78f60ee2598: Pull complete
ec1566b1a9: Pull complete
7fe768ae11c8: Pull complete
d527a1181af2: Pull complete
90ed12f9090a: Pull complete
Digest: sha256:37b91ef09491bd36cdae4ca95a95a0e4b992847fa7d9990e99b271c049dd3d
Status: Downloaded newer image for postgres:12
Pulling sonarqube (sonarqube:community)...
community: Pulling from library/sonarqube
6414378b6477: Pull complete
17da818a0a08: Pull complete
d1258e90d61: Pull complete
f44133ca2b7f: Pull complete
143733ae87a4: Pull complete
524c194d4b54: Pull complete
8d0039a7530a: Pull complete
414fb70e0a08: Pull complete
Digest: sha256:0942acd35f051ce44f8edaf45ac93f7c9079017d60d99f614663e60cef5efe9
Status: Downloaded newer image for sonarqube:community
Creating postgres12 ... done
Creating sonarqube ... done
root@ip-172-31-32-92:~#

```

```

root@ip-172-31-32-92:~#
root@ip-172-31-32-92:~# docker images
REPOSITORY          TAG        IMAGE ID      CREATED       SIZE
sonarqube           community  e3295fe6246a  5 weeks ago   1.16B
postgres            12         180d00dc8fca  3 months ago  419MB
root@ip-172-31-32-92:~#
root@ip-172-31-32-92:~#
root@ip-172-31-32-92:~#
root@ip-172-31-32-92:~# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED          STATUS          PORTS          NAMES
7556a37e5f91        sonarqube:community "/opt/sonarqube/docker-entrypoint.sh"   22 seconds ago   Up 8 seconds   0.0.0.0:9000->9000/tcp, ::1:9000->9000/tcp   sonarqube
4c1d943la321        postgres:12          "docker-entrypoint.s..."   23 seconds ago   Up 21 seconds   5432/tcp          postgres12
root@ip-172-31-32-92:~#
root@ip-172-31-32-92:~#
root@ip-172-31-32-92:~#
root@ip-172-31-32-92:~# sysctl vm.max_map_count=52488
vm.max_map_count = 52488
root@ip-172-31-32-92:~#
root@ip-172-31-32-92:~# sysctl -w vm.max_map_count=52488
vm.max_map_count = 52488
root@ip-172-31-32-92:~#

```

The screenshot shows a web browser window with the following details:

- URL:** `18.183.182.211:9000/sessions/new?return_to=%2F`
- Content:** A SonarQube login form. The form has two fields: "Login" (containing "admin") and "Password" (containing "*****"). Below the form are two buttons: "Go back" and "Log in".

Not secure 18.183.182.211:9000/account/reset_password

Update your password

This account should not use the default password.

Enter a new password
All fields marked with * are required

Old Password *

Password *

Confirm Password *

Update

Not secure 18.183.182.211:9000/projects/create

sonarqube

Projects Issues Rules Quality Profiles Quality Gates Administration More

How do you want to create your project?

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.

Import from Azure DevOps 

Import from Bitbucket Cloud 

Import from Bitbucket Server 

Import from GitHub 

Import from GitLab 

Create a local project

Step 3 :

- 1. Install SonarQube Plugin in Jenkins:**
 - Navigate to Manage Jenkins > Manage Plugins > Available and search for **SonarQube Scanner for Jenkins**.
 - Install the plugin and restart Jenkins.
- 2. Configure SonarQube in Jenkins:**
 - Go to Manage Jenkins > Configure System.
 - Find the **SonarQube Servers** section and add the URL of your SonarQube server.
 - Provide the necessary authentication token (create one in SonarQube if not already created).
- 3. Add SonarQube Scanner in Jenkins Job:**
 - In your Jenkins pipeline, add the following step for SonarQube analysis:

```
stage('SonarQube-SAST') {  
    steps {  
        script {  
            def scannerHome = tool 'sonarscanner6'  
            withSonarQubeEnv('sonar-pro') {  
                sh "${scannerHome}/bin/sonar-scanner -Dsonar.projectKey=rocket-nodejs"  
            }  
        }  
    }  
}
```

[Not secure](https://54.178.66.43:8080/) 54.178.66.43:8080

Jenkins

Dashboard >

+ New Item Add description

All +

S	W	Name ↴	Last Success	Last Failure	Last Duration
		CICD-Pipeline-Project	27 min #1	N/A	4 sec

Icon: S M L

Build Queue: No builds in the queue.

Build Executor Status: 0/2

REST API Jenkins 2.484

54.178.66.43:8080/job/CICD-Pipeline-Project/

Jenkins

Dashboard > Manage Jenkins

+ New Item Search settings

Set up agent Set up cloud Dismiss

Manage Jenkins

Building on the built-in node can be a security issue. You should set up distributed builds. See [the documentation](#).

System Configuration

- System** Configure global settings and paths.
- Tools** Configure tools, their locations and automatic installers.
- Nodes** Add, remove, control and monitor the various nodes that Jenkins runs jobs on.
- Clouds** Add, remove, and configure cloud instances to provision agents on-demand.
- Plugins** Add, remove, disable or enable plugins that can extend the functionality of Jenkins.
- Appearance** Configure the look and feel of Jenkins

Security

- Security** Secure Jenkins; define who is allowed to access/use the system.
- Credentials** Configure credentials
- Credential Providers** Configure the credential providers and types

REST API Jenkins 2.484

[54.178.66.43:8080/manage/pluginManager/available](https://54.178.66.43:8080/manage/pluginManager/)

Jenkins

Dashboard > Manage Jenkins > Plugins

Updates Available plugins Installed plugins Advanced settings Download progress

Plugins

Search: sonarqube Install ↻

Install	Name ↴	Released
<input checked="" type="checkbox"/>	SonarQube Scanner 2.17.2	8 mo 22 days ago
<input type="checkbox"/>	External Site/Tool Integrations	
<input type="checkbox"/>	Build Reports	
This plugin allows an easy integration of SonarQube, the open source platform for Continuous Inspection of code quality.		
<input type="checkbox"/>	Sonar Gerrit 388.v9b_f1cb_e42306	5 mo 6 days ago
<input type="checkbox"/>	External Site/Tool Integrations	
This plugin allows to submit issues from SonarQube to Gerrit as comments directly.		
<input type="checkbox"/>	SonarQube Generic Coverage 1.0	5 yr 3 mo ago
TODO		

REST API Jenkins 2.484

← → ⚡ Not secure 54.178.66.43:8080/manage/pluginManager/updates/

Dashboard > Manage Jenkins > Plugins

Plugins

- Updates
- Available plugins
- Installed plugins
- Advanced settings
- Download progress

	Metrics	Status
	Pipeline Graph View	Success
	Git	Success
	EDDSA API	Success
	Trilead API	Success
	SSH Build Agents	Success
	Matrix Authorization Strategy	Success
	PAM Authentication	Success
	LDAP	Success
	Email Extension	Success
	Mailer	Success
	Theme Manager	Success
	Dark Theme	Success
	Loading plugin extensions	Success
	SonarCube Scanner	Success
	Loading plugin extensions	Success

→ [Go back to the top page](#)
 (you can start using the installed plugins right away)

→ Restart Jenkins when installation is complete and no jobs are running

54.178.66.43:8080 REST API Jenkins 2.484

← → ⚡ Not secure 18.183.182.211:9000/account/security

sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration More

A Administrator

Profile Security Notifications Projects

Security

If you want to enforce security by not providing credentials of a real SonarQube user to run your code scan or to invoke web services, you can provide a User Token as a replacement of the user login. This will increase the security of your installation by not letting your analysis user's password going through your network.

Generate Tokens

Name	Type	Expires in
Enter Token Name	Select Token Type	30 days
<button>Generate</button>		

New token "sonarqube" has been created. Make sure you copy it now, you won't be able to see it again!

sqa_9d5e486495472ed7d7c84c40dbf96e7d07c78985 [Copy](#)

Name	Type	Project	Last use	Created	Expiration
sonarqube	Global		Never	November 7, 2024	December 7, 2024
					Revoke

← → ⚡ Not secure 54.178.66.43:8080/manage/

Dashboard > Manage Jenkins

My Views

System Configuration

- Build Queue
- Build Executor Status
- System
- Nodes
- Tools
- Clouds
- Plugins
- Appearance

Security

- Security
- Credentials
- Users
- Credential Providers

Status Information

- System Information
- Custom 1
- Load Statistics

54.178.66.43:8080/manage/credentials

Not secure 54.178.66.43:8080/manage/credentials/store/system/

Jenkins

Dashboard > Manage Jenkins > Credentials > System >

System

+ Add domain

Domain ↓

Description



Global credentials (unrestricted) ▾

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

Icon: S M L

REST API Jenkins 2.484

54.178.66.43:8080/manage/credentials/store/system/domain/

Not secure 54.178.66.43:8080/manage/credentials/store/system/domain/ /

Jenkins

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

Global credentials (unrestricted)

+ Add Credentials

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

ID

Name

Kind

Description

This credential domain is empty. How about [adding some credentials?](#)

Icon: S M L

REST API Jenkins 2.484

54.178.66.43:8080/manage/credentials/store/system/domain/_/newCredentials

Not secure 54.178.66.43:8080/manage/credentials/store/system/domain/_/credential/sonar-pro/update

Jenkins

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

Update

Update credentials

Delete

Move

Scope ?

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

Change Password

Secret

Concealed

ID ?

sonar-pro

Description ?

SonarPWD

Save

REST API Jenkins 2.484

Not secure 54.178.66.43:8080/manage/credentials/store/system/domain/_/ Jenkins

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

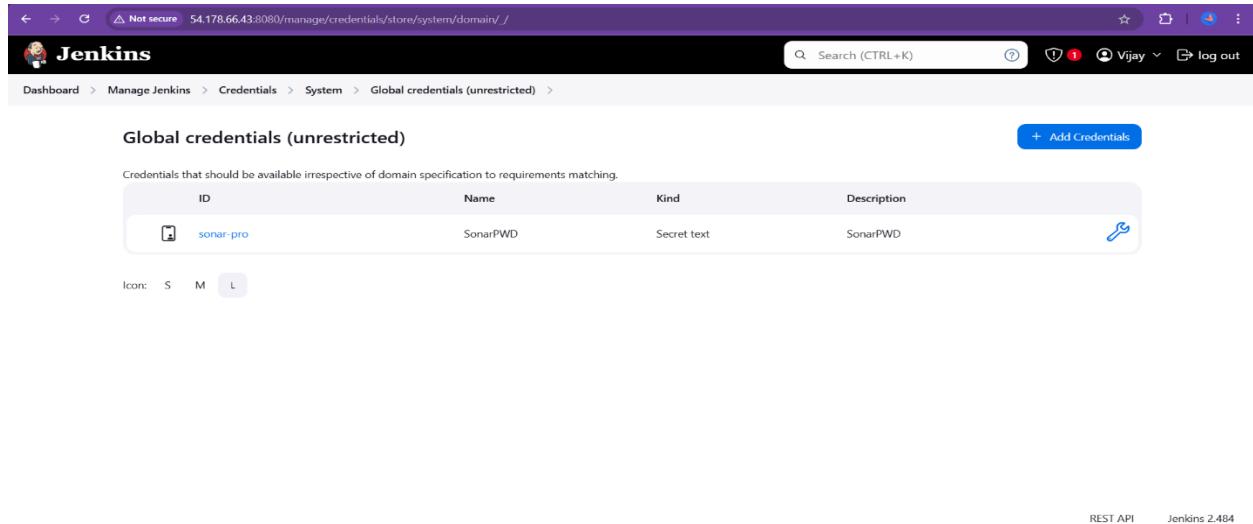
Global credentials (unrestricted)

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

ID	Name	Kind	Description
	sonar-pro	Secret text	SonarPWD

Icon: S M L

REST API Jenkins 2.484



Not secure 54.178.66.43:8080/manage/ Jenkins

Dashboard > Manage Jenkins

Manage Jenkins

Building on the built-in node can be a security issue. You should set up distributed builds. See [the documentation](#).

[Set up agent](#) [Set up cloud](#) [Dismiss](#)

System Configuration

- Build Queue: No builds in the queue.
- System: Configure global settings and paths.
- Tools: Configure tools, their locations and automatic installers.
- Plugins: Add, remove, disable or enable plugins that can extend the functionality of Jenkins.

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

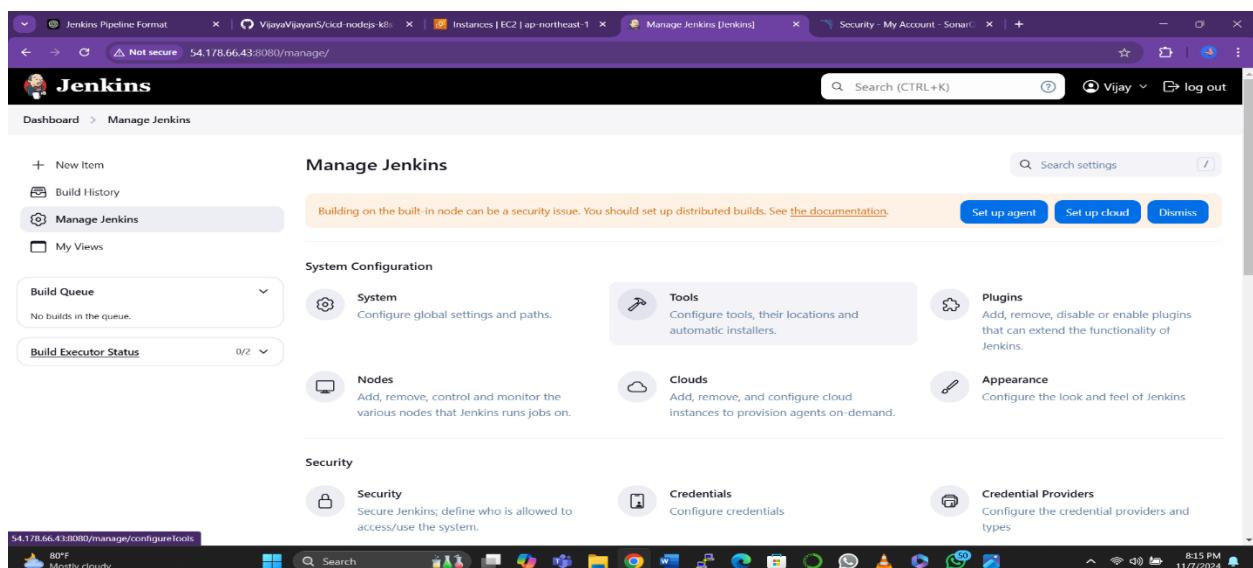
Clouds: Add, remove, and configure cloud instances to provision agents on-demand.

Appearance: Configure the look and feel of Jenkins.

Security

- Security: Secure Jenkins; define who is allowed to access/use the system.
- Credentials: Configure credentials.
- Credential Providers: Configure the credential providers and types.

54.178.66.43:8080/manage/configuretools



Not secure 54.178.66.43:8080/ Jenkins

Dashboard >

[New Item](#) [Add description](#)

[Build History](#)

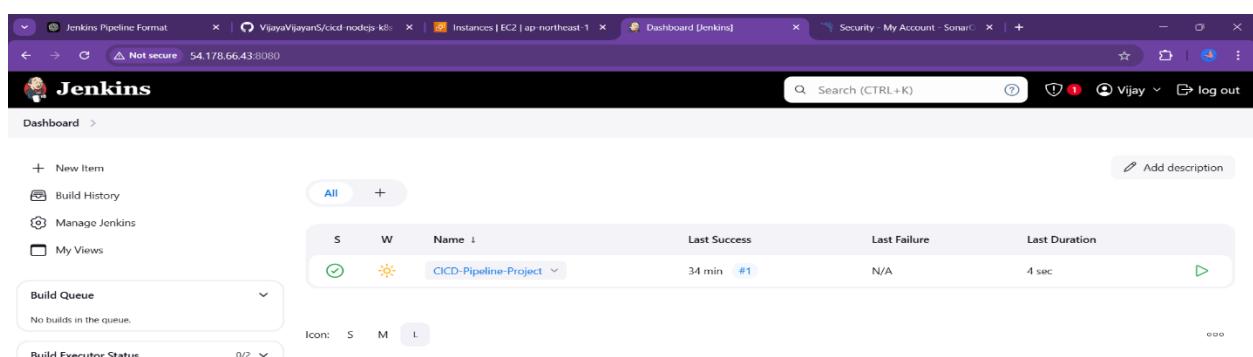
[Manage Jenkins](#)

[My Views](#)

Build Queue: No builds in the queue.

Build Executor Status: 0/2

Icon: S M L



54.178.66.43:8080/job/CICD-Pipeline-Project/ Jenkins

81°F Partly cloudy

Search

REST API Jenkins 2.484



Not secure 54.178.66.43:8080/job/CICD-Pipeline-Project/configure

Dashboard > CICD-Pipeline-Project > Configuration

Configure

Definition Pipeline script

General Build Triggers Advanced Project Options Pipeline

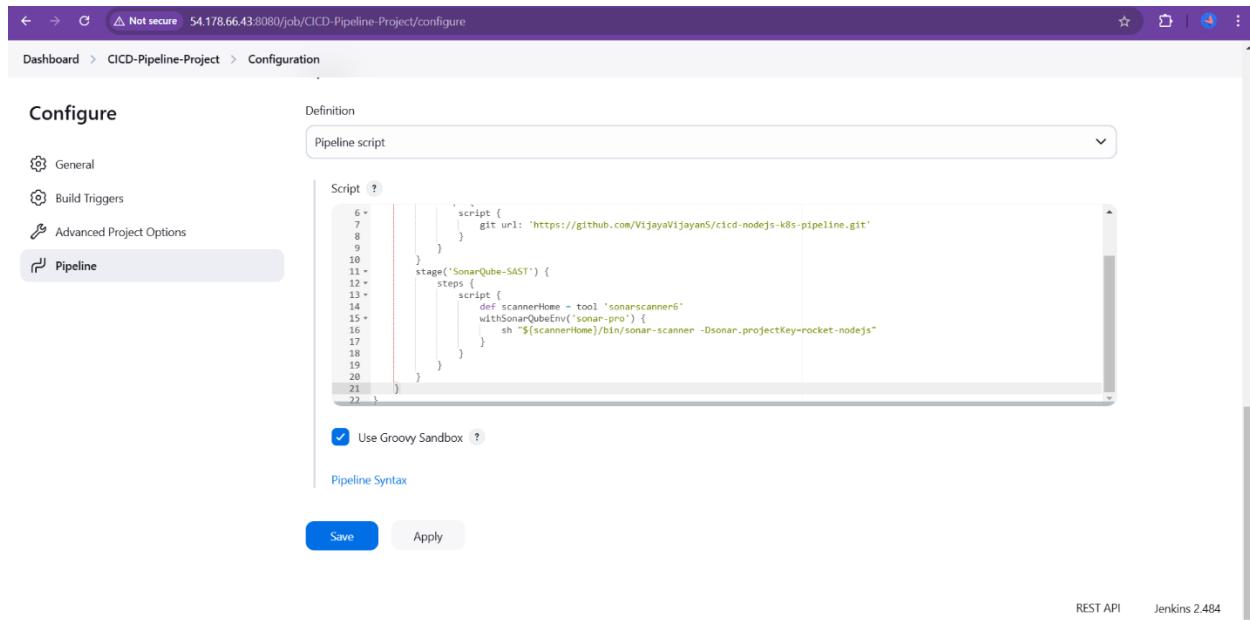
```
6 *          script {  
7     git url: 'https://github.com/VijayaVijayanS/cicd-nodejs-k8s-pipeline.git'  
8 }  
9 }  
10 }  
11 stage('SonarQube-SAST') {  
12     steps {  
13         script {  
14             def scannerHome = tool 'sonarscanner6'  
15             withSonarQubeEnv('sonar-pro') {  
16                 sh "${scannerHome}/bin/sonar-scanner -Dsonar.projectKey=rocket-nodejs"  
17             }  
18         }  
19     }  
20 }  
21 }  
22 }
```

Use Groovy Sandbox

Pipeline Syntax

Save Apply

REST API Jenkins 2.484



Not secure 54.178.66.43:8080/manage/configureTools/

Dashboard > Manage Jenkins > Tools

SonarQube Scanner installations Edited

Add SonarQube Scanner

SonarQube Scanner

Name: sonarscanner6

Install automatically

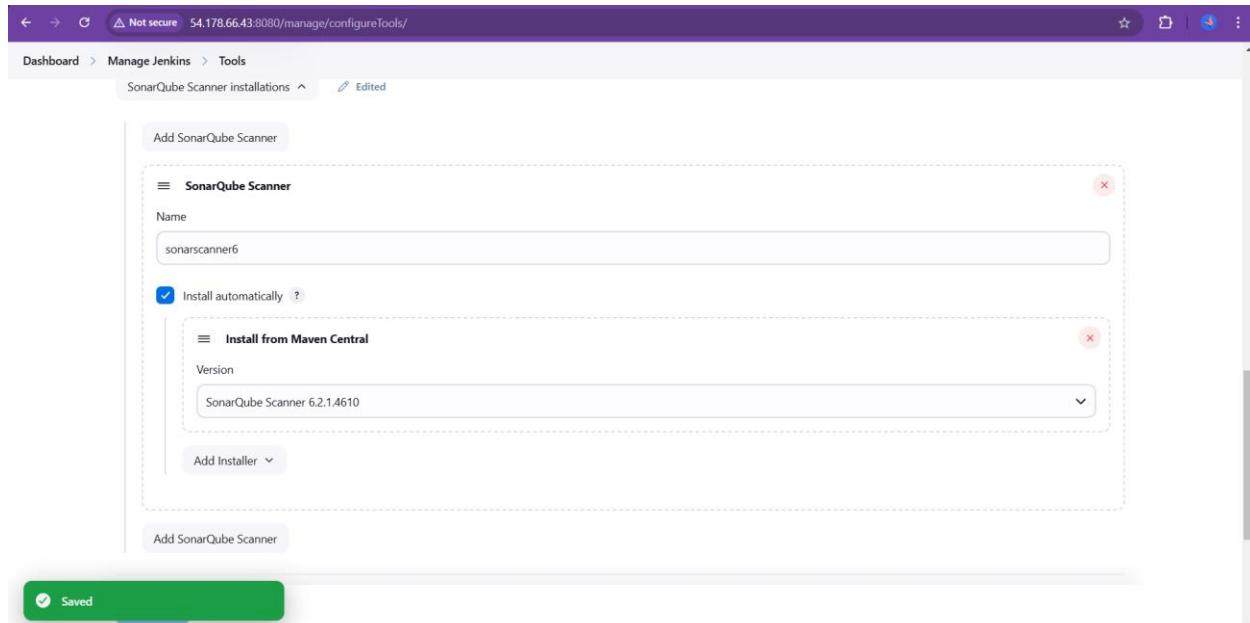
Install from Maven Central

Version: SonarQube Scanner 6.2.1.4610

Add Installer

Added SonarQube Scanner

Saved



Not secure 54.178.66.43:8080/manage/configure

Dashboard > Manage Jenkins > System >

- Environment variables
- SonarQube installations
- List of SonarQube installations

Name: sonar-pro

Server URL: http://localhost:9000
Default is http://localhost:9000

http://18.183.182.211:9000/

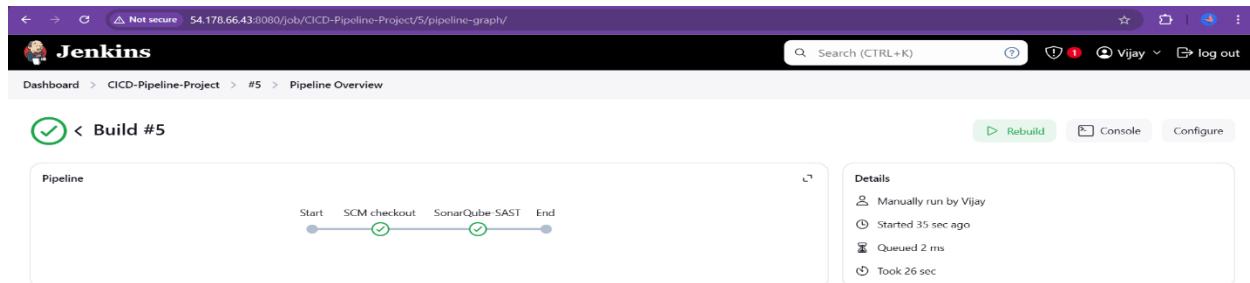
Server authentication token: SonarQube authentication token. Mandatory when anonymous access is disabled.

SonarPWD

+ Add

Advanced ▾

Saved



Jenkins 2.484

Not secure 54.178.66.43:8080/job/CICD-Pipeline-Project/5/console

Dashboard > CICD-Pipeline-Project > #5

```

14:59:09.552 INFO SCM Publisher 7/7 source files have been analyzed (done) | time=184ms
14:59:09.556 INFO CPD Executor Calculating CPD for 1 file
14:59:09.563 INFO CPD Executor CPD calculation finished (done) | time=6ms
14:59:09.569 INFO SCM revision ID '02f8327750e154ebfd3e4a54175e38eedf37dd5a'
14:59:09.668 INFO Analysis report generated in 91ms, dir size=230.5 kB
14:59:09.680 INFO Analysis report compressed in 21ms, zip size=32.7 kB
14:59:10.043 INFO Analysis report uploaded in 362ms
14:59:10.043 INFO ANALYSIS SUCCESSFUL, you can find the results at: http://18.183.182.211:9000/dashboard?id=rocket-nodejs
14:59:10.043 INFO Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
14:59:10.043 INFO More about the report processing at http://18.183.182.211:9000/api/ce/task?id=228ffb7a-ea7d-4d39-810b-b258c8e3b2ad
14:59:10.655 INFO Analysis total time: 18.199 s
14:59:10.656 INFO SonarScanner Engine completed successfully
14:59:10.995 INFO EXECUTION SUCCESS
14:59:10.996 INFO Total time: 24.264s
[Pipeline] 
[Pipeline] // withSonarQubeEnv
[Pipeline] 
[Pipeline] // script
[Pipeline] 
[Pipeline] // stage
[Pipeline] 
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS

```

Not secure 18.183.182.211:9000/projects

sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration More ? A Create Project

My Favorites All

Search for projects... Perspective Overall Status Sort by Name 1 project(s)

Filters

Quality Gate

- Passed 1
- Failed 0

Security

- A ≥ 0 info issues 0
- B ≥ 1 minor issue 0
- C ≥ 1 major issue 1
- D ≥ 1 critical issue 0
- E ≥ 1 blocker issue 0

Reliability

rocket-nodejs PUBLIC

Last analysis: 1 minute ago - 141 Lines of Code - YAML, JavaScript, ...

Security Reliability Maintainability Hotspots Reviewed Coverage Duplications

Passed

1 of 1 shown

SonarQube™ technology is powered by SonarSource SA

Community Edition v10.7 (96327) ACTIVE LGPL v3 Community Documentation Plugins Web API

Not secure 18.183.182.211:9000/dashboard?Id=rocket-nodejs&codeScope=overall

sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration More ? A

rocket-nodejs / main

Overview Issues Security Hotspots Measures Code Activity Project Settings Project Information

main 141 Lines of Code · Version not provided

Quality Gate Passed Last analysis 1 minute ago

New Code Overall Code

Security	Reliability	Maintainability
6 Open issues	6 Open issues	12 Open issues
Accepted issues 0	Coverage 0.0%	Duplications 0.0%
Valid issues that were not fixed	On 9 lines to cover.	On 166 lines.

Security Hotspots

Step 3: npm Build

1. Install Node.js Plugin in Jenkins:
 - o Go to Manage Jenkins > Manage Plugins > Available and search for NodeJS Plugin.
 - o Install the plugin and restart Jenkins.
2. Configure Node.js in Jenkins:
 - o Go to Manage Jenkins > Global Tool Configuration.
 - o Under NodeJS, add a new NodeJS installation. Make sure the Node.js and npm are installed on the Jenkins agent where this job will run.
3. Configure Build Step for npm:
 - o In the pipeline script, add the following:

```
stage('Build') {  
    steps {  
        script {  
            sh 'npm install'  
        }  
    }  
}
```

4. Installing Node.js with Apt from the Default Repositories

- sudo apt update
- sudo apt install nodejs
- node -v
- sudo apt install npm

```

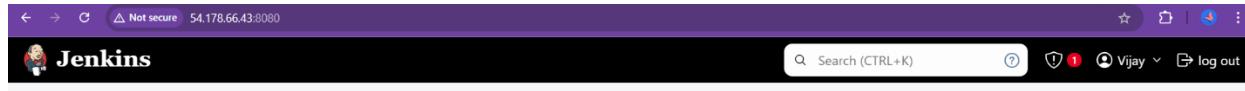
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# sudo apt install nodejs
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libcares2 libnode109 node-acorn node-busboy node-cjs-module-lexer node-undici node-xtend nodejs-doc
Suggested packages:
  npm
The following NEW packages will be installed:
  libcares2 libnode109 node-acorn node-busboy node-cjs-module-lexer node-undici node-xtend nodejs nodejs-doc
0 upgraded, 9 newly installed, 0 to remove and 30 not upgraded.
Need to get 16.1 MB of archives.
After this operation, 70.4 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libcares2 amd64 1.27.0-1.0ubuntu1 [73.7 kB]
Get:2 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu/universe amd64 node-xtend all 4.0.2-3 [3902 B]
Get:3 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu/universe amd64 nodejs amd64 10.19.1+dfsg-6ubuntu5 [306 kB]
Get:4 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu/universe amd64 node-acorn all 0.8.1+ds+cs25.17.7-2 [115 kB]
Get:5 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu/universe amd64 node-cjs-module-lexer all 1.2.3+dfsg-1 [32.1 kB]
Get:6 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu/universe amd64 node-busboy all 1.6.0+cs2.6.0-2 [17.3 kB]
Get:7 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu/noble/universe amd64 libnode109 amd64 10.19.1+dfsg-6ubuntu5 [325 kB]
Get:8 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu/noble/universe amd64 libnode109 amd64 10.19.1+dfsg-6ubuntu5 [11.6 kB]
Get:9 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu/noble/universe amd64 nodejs-doc all 10.19.1+dfsg-6ubuntu5 [3552 kB]
Fetched 16.1 MB in 0s (80.8 MB/s)
Selecting previously unselected package libcares2:amd64.
(Reading database ... 82587 files and directories currently installed.)
Preparing to unpack .../0-libcares2_1.27.0-1.0ubuntu1_amd64.deb ...
Unpacking libcares2:amd64 (1.27.0-1.0ubuntu1) ...
Selecting previously unselected package node-xtend.
Preparing to unpack .../1-node-xtend_4.0.2-3_all.deb ...
Unpacking node-xtend (4.0.2-3) ...
Selecting previously unselected package nodejs.
Preparing to unpack .../2-nodejs_10.19.1+dfsg-6ubuntu5_amd64.deb ...
Unpacking nodejs (10.19.1+dfsg-6ubuntu5) ...
Selecting previously unselected package node-acorn.
Preparing to unpack .../3-node-acorn_0.8.1+ds+cs25.17.7-2_all.deb ...
Unpacking node-acorn (0.8.1+ds+cs25.17.7-2) ...
Selecting previously unselected package node-cjs-module-lexer.
Preparing to unpack .../4-node-cjs-module-lexer_1.2.3+dfsg-1_all.deb ...
Unpacking node-cjs-module-lexer (1.2.3+dfsg-1) ...
Selecting previously unselected package node-busboy.
Preparing to unpack .../5-node-busboy_1.6.0+cs2.6.0-2_all.deb ...
Unpacking node-busboy (1.6.0+cs2.6.0-2) ...
Selecting previously unselected package node-undici.
Preparing to unpack .../6-node-undici_5.26.3+dfsg1+cs23.10.12-2_all.deb ...
Unpacking node-undici (5.26.3+dfsg1+cs23.10.12-2) ...
Selecting previously unselected package libnode109:amd64.
Preparing to unpack .../7-libnode109_10.19.1+dfsg-6ubuntu5_amd64.deb ...
Unpacking libnode109:amd64 (10.19.1+dfsg-6ubuntu5) ...

```

```

root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# root@ip-172-31-43-145:~# root@ip-172-31-43-145:~# node -v
v10.19.1
root@ip-172-31-43-145:~# 

```



Dashboard >

+ New Item Add description

All +

S	W	Name ↓	Last Success	Last Failure	Last Duration
		CICD-Pipeline-Project	1 hr 1 min #5	1 hr 3 min #4	26 sec

Build Queue No builds in the queue.

Icon: S M L

Build Executor Status 0/2

REST API Jenkins 2.484

Not secure 54.178.66.43:8080/manage/pluginManager/available

Jenkins

Dashboard > Manage Jenkins > Plugins

Plugins

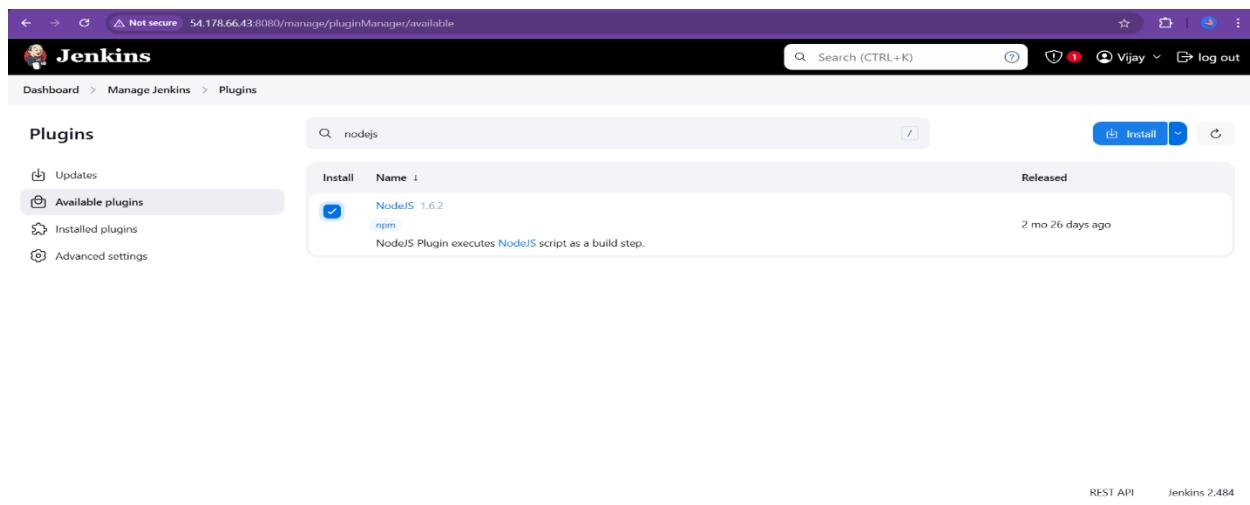
Updates Available plugins Installed plugins Advanced settings

Search: nodejs

Install Name ↴ Released

NodeJS 1.6.2 npm 2 mo 26 days ago
NodeJS Plugin executes NodeJS script as a build step.

REST API Jenkins 2.484



Not secure 54.178.66.43:8080/manage/pluginManager/updates/

Jenkins

Dashboard > Manage Jenkins > Plugins

Plugins

Updates Available plugins Installed plugins Advanced settings

Download progress

Preparation

- Checking internet connectivity Success
- Checking update center connectivity Success
- Success

Config File Provider

NodeJS Success

Loading plugin extensions Success

→ [Go back to the top page](#)
(you can start using the installed plugins right away)

→ Restart Jenkins when installation is complete and no jobs are running



54.178.66.43:8080

Not secure 54.178.66.43:8080/job/CICD-Pipeline-Project/configure

Jenkins

Dashboard > CICD-Pipeline-Project > Configuration

Configure

General Build Triggers Advanced Project Options Pipeline

Definition

Pipeline script

Script ?

```
13 *           script {
14   def scannerHome = tool 'sonarscanner6'
15   withSonarQubeEnv('sonar-pro') {
16     sh "${scannerHome}/bin/sonar-scanner -Dsonar.projectKey=rocket-nodejs"
17   }
18 }
19
20 }
21 }
22 stage('Build') {
23   steps {
24     script {
25       sh 'npm install'
26     }
27   }
28 }
29 }
```

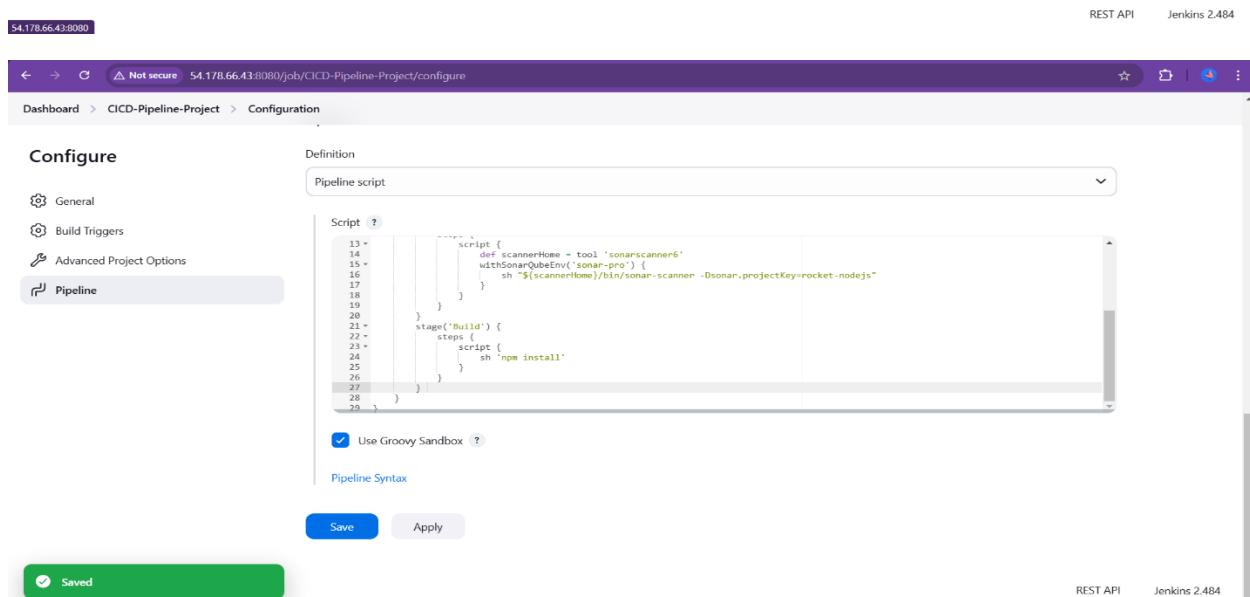
Use Groovy Sandbox ?

Pipeline Syntax

Save Apply

Saved

REST API Jenkins 2.484





Jenkins 2.484

A screenshot of a web browser showing the Jenkins Console output for Build #6. The URL is 54.178.66.43:8080/job/CICD-Pipeline-Project/6/console. The page header includes the Jenkins logo, a search bar, and user information for 'Vijay'. Below the header, the breadcrumb navigation shows 'Dashboard > CICD-Pipeline-Project > #6'. The console output window displays the following log entries:

```
[Pipeline] // script
[Pipeline]
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Build)
[Pipeline] script
[Pipeline] {
[Pipeline] sh
+ npm install

added 65 packages, and audited 66 packages in 2s

13 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
[Pipeline] }
[Pipeline] // script
[Pipeline]
[Pipeline] // stage
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```

REST API Jenkins 2.484

Step 4: Build Docker Images

1. Dockerfile Setup: Ensure the Node.js application has a valid Dockerfile in the root of the repository.

```
# Use an official Node.js runtime as a parent image
FROM node:14

# Set the working directory
WORKDIR /usr/src/app

# Copy package.json and package-lock.json
COPY package*.json .

# Install app dependencies
RUN npm install

# Copy the app source code
COPY .

# Expose the port the app runs on
EXPOSE 80

# Command to run the app
CMD ["node", "server.js"]
```

2. **Configure Docker Build in Jenkins:**

- In the pipeline script, add a step to build the Docker image:

```
stage('Docker Build Image') {
    steps {
        script {
            sh 'docker build -t vijay63833/helm-rockets:v1.1 .'
            sh 'docker images'
        }
    }
}
```

```
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# docker
Command 'docker' not found, but can be installed with:
snap install docker      # version 24.0.5, or
apt install docker.io    # version 24.0.7-Ubuntu4.1
apt install podman-docker # version 4.9.3+ds1-lubuntu0.2
See 'snap info docker' for additional versions.
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# apt install docker.io
```

```
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# chmod 777 /var/run/docker.sock
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~#
```

The screenshot shows the Jenkins dashboard at the URL 54.178.66.43:8080. The top navigation bar includes links for 'Dashboard', 'Build History', 'Manage Jenkins', and 'My Views'. A search bar and a user profile for 'Vijay' are also present. The main content area displays a table for the 'CICD-Pipeline-Project'. The table has columns for Status (S), Workstation (W), Name (CICD-Pipeline-Project), Last Success (12 min), Last Failure (1 hr 18 min), and Last Duration (24 sec). Below the table, there are sections for 'Build Queue' (empty) and 'Build Executor Status' (0/2).

S	W	Name	Last Success	Last Failure	Last Duration
		CICD-Pipeline-Project	12 min #6	1 hr 18 min #4	24 sec

Not secure 54.178.66.43:8080/job/CICD-Pipeline-Project/configure

Dashboard > CICD-Pipeline-Project > Configuration

Configure

Definition

Pipeline script

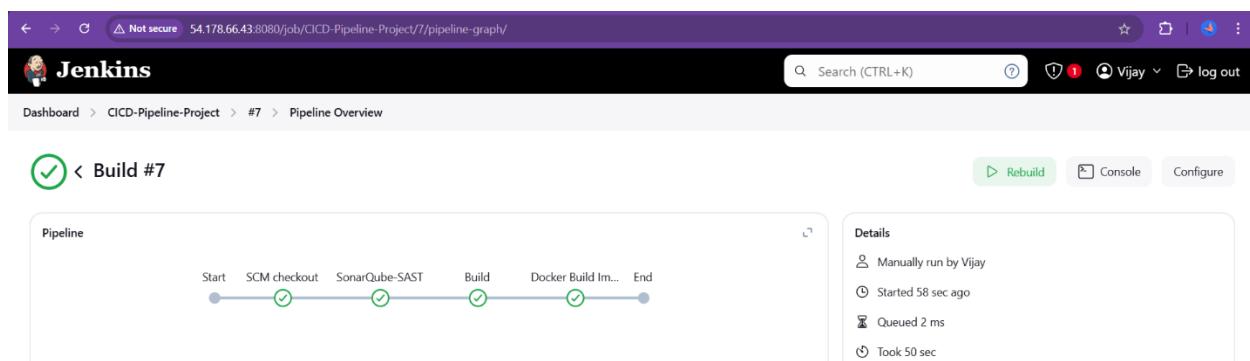
```
Script ?  
18 | }  
19 | }  
20 | }  
21 *   stage('Build') {  
22 *     steps {  
23 *       script {  
24 *         sh 'npm install'  
25 *       }  
26 *     }  
27 *   }  
28 *   stage('Docker Build Image') {  
29 *     steps {  
30 *       script {  
31 *         sh 'docker build -t vijay63833/helm-rockets:v1.1.'  
32 *         sh 'docker images'  
33 *       }  
34 *     }  
35 *   }
```

Use Groovy Sandbox ?

Pipeline Syntax

Save Apply

REST API Jenkins 2.484



← → ⚡ Not secure 54.178.66.43:8080/job/CICD-Pipeline-Project/7/console

Dashboard > CICD-Pipeline-Project > #7

```
...> b68cfa6fe47d
Step 6/7 : EXPOSE 80
--> Running in 08cc1490d7be
Removing intermediate container 08cc1490d7be
--> b3d5d15ea9
Step 7/7 : CMD ["node", "server.js"]
--> Running in 837faf0e047c0
Removing intermediate container 837faf0e047c0
--> fc3467639665
Successfully built fc3467639665
Successfully tagged dockerhub1010/helm-rockets:v1.1
[Pipeline] sh
+ docker images
REPOSITORY          TAG      IMAGE ID      CREATED        SIZE
dockerhub1010/helm-rockets  v1.1   fc3467639665  1 second ago  919MB
node                14     1d12470fa662  19 months ago  912MB
[Pipeline] )
[Pipeline] // script
[Pipeline] )
[Pipeline] // stage
[Pipeline] )
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```

REST API Jenkins 2.484

```
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# chmod 777 /var/run/docker.sock
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# docker images
REPOSITORY          TAG      IMAGE ID      CREATED        SIZE
dockerhub1010/helm-rockets  v1.1   fc3467639665  36 seconds ago  919MB
node                14     1d12470fa662  19 months ago  912MB
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# docker ps
CONTAINER ID  IMAGE      COMMAND      CREATED      STATUS      PORTS      NAMES
root@ip-172-31-43-145:~#
```

Step 5: Push Docker Images to Docker Hub

1. Configure Docker Hub Credentials in Jenkins:

- Go to Manage Jenkins > Manage Credentials > Global credentials (unrestricted) > Add Credentials.
- Choose **Username with password** and enter your Docker Hub credentials.

2. Push Image to Docker Hub:

- In the pipeline script, add the following step to log in to Docker Hub and push the image:

```
stage('Docker Push') {  
    steps {  
        script {  
            withCredentials([string(credentialsId: 'dockerPass', variable: 'dockerPassword')]) {  
                sh "echo ${dockerPassword} | docker login -u vijay63833 --password-stdin"  
                sh 'docker push vijay63833/helm-rockets:v1.1'  
            }  
        }  
    }  
}
```

The screenshot shows the Jenkins dashboard at the URL 54.178.66.43:8080. The top navigation bar includes links for 'Not secure', '54.178.66.43:8080', 'Dashboard', 'Search (CTRL+K)', 'Vijay', and 'log out'. The main content area displays the 'CICD-Pipeline-Project' view, which is the last successful build. It shows the build status (green), name ('CICD-Pipeline-Project'), last success time ('1 min 52 sec #7'), last failure time ('1 hr 24 min #4'), and last duration ('50 sec'). On the left sidebar, there are links for 'New Item', 'Build History', 'Manage Jenkins', 'My Views', 'Build Queue' (empty), and 'Build Executor Status' (0/2). A bottom footer bar includes links for 'REST API' and 'Jenkins 2.484'.

Not secure 54.178.66.43:8080/manage/

Dashboard > Manage Jenkins

Managed files
e.g. settings.xml for maven, central managed scripts, custom files, ...

Security

- Security**
Secure Jenkins; define who is allowed to access/use the system.
- Credentials**
Configure credentials
- Credential Providers**
Configure the credential providers and types

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

Status Information

- System Information**
Displays various environmental information to assist trouble-shooting.
- System Log**
System log captures output from java.util.logging output related to Jenkins.
- Load Statistics**
Check your resource utilization and see if you need more computers for your builds.

About Jenkins
See the version and license information.

54.178.66.43:8080/manage/credentials

Not secure 54.178.66.43:8080/manage/credentials/

Jenkins

Search (CTRL+K) Vijay log out

Dashboard > Manage Jenkins > Credentials

Credentials

T	P	Store ↓	Domain	ID	Name
		System	(global)	sonar-pro	SonarPWD

Stores scoped to Jenkins

P	Store ↓	Domains
		(global)

Icon: S M L

Not secure 54.178.66.43:8080/manage/credentials/store/system/

Jenkins

Dashboard > Manage Jenkins > Credentials > System >

System

+ Add domain

Domain ↴

Description



Global credentials (unrestricted) ▾

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

Icon: S M L

REST API Jenkins 2.484

54.178.66.43:8080/manage/credentials/store/system/domain/_/

Jenkins

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

Global credentials (unrestricted)

+ Add Credentials

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

ID

Name

Kind

Description



sonar-pro

SonarPWD

Secret text

SonarPWD



Icon: S M L

REST API Jenkins 2.484

54.178.66.43:8080/manage/credentials/store/system/domain/_/newCredentials

Jenkins

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

New credentials

Kind

Secret text

Scope ?

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

Secret

.....

ID ?

dockerPass

Description ?

dockerpassword

Create

Not secure 54.178.66.43:8080/manage/credentials/store/system/domain/_/

Jenkins

Search ? Vijay log out

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

Global credentials (unrestricted)

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

ID	Name	Kind	Description
sonar-pro	SonarPWD	Secret text	SonarPWD
dockerPass	dockerpassword	Secret text	dockerpassword

Icon: S M L

REST API Jenkins 2.484

```
root@ip-172-31-43-145:~/var/lib/jenkins/workspace/CICD-Pipeline-Project
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# chmod 777 /var/run/docker.sock
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# docker images
REPOSITORY          TAG      IMAGE ID      CREATED        SIZE
dockerrhub1010/helm-rockets   v1.1      fc3467639665  36 seconds ago  919MB
node                14       1d12470fa662  19 months ago   912MB
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# docker ps
CONTAINER ID        IMAGE      COMMAND     CREATED      STATUS      PORTS      NAMES
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# cd ~/var/lib/jenkins/workspace
root@ip-172-31-43-145:~/var/lib/jenkins/workspace# ls
CICD-Pipeline-Project  CICD-Pipeline-Project@tmp
root@ip-172-31-43-145:~/var/lib/jenkins/workspace# cd CICD-Pipeline-Project
root@ip-172-31-43-145:~/var/lib/jenkins/workspace/CICD-Pipeline-Project# ls
Dockerfile  Helm  Jenkinsfile 'Jenkinsfile-skip stage'  README.md  buildspec.yml  node_modules  package-lock.json  package.json  rocket-app.yml  server.js
root@ip-172-31-43-145:~/var/lib/jenkins/workspace/CICD-Pipeline-Project#
```

Not secure 54.178.66.43:8080/job/CICD-Pipeline-Project/configure

Dashboard > CICD-Pipeline-Project > Configuration

Configure

Definition Pipeline script

General Build Triggers Advanced Project Options Pipeline

Script ?

```

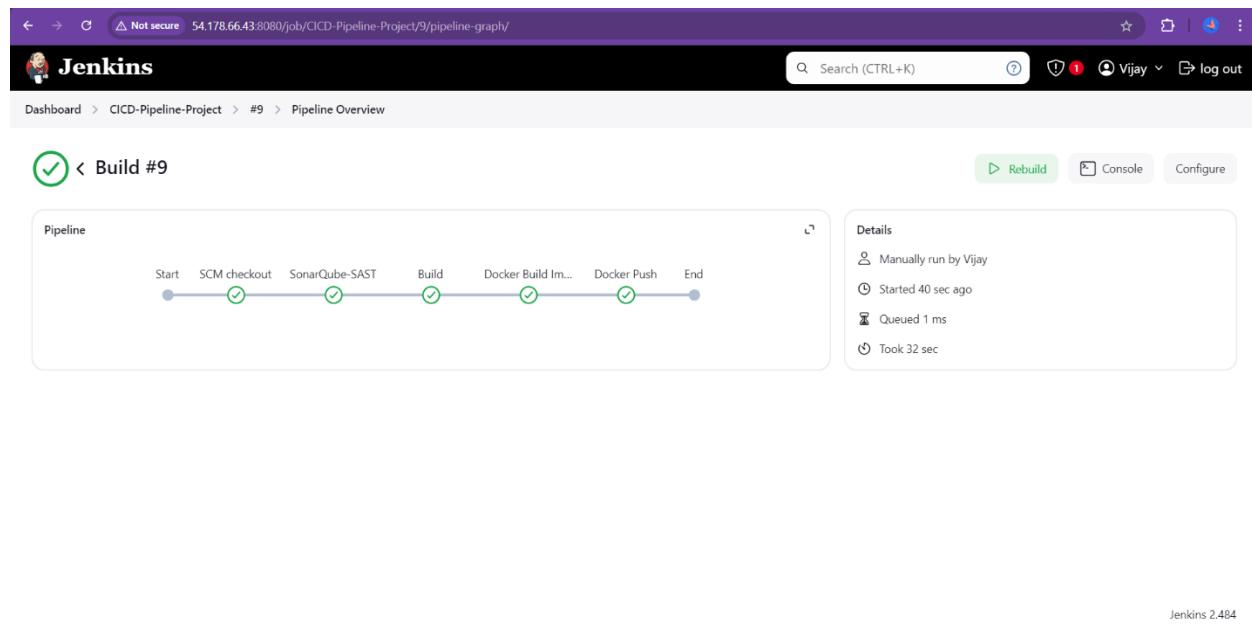
31  sh `docker build -t vijay63833/helm-rockets:v1.1 .`
32  sh `docker images`
33  }
34  }
35  }
36  *
37  stage('Docker Push') {
38    steps {
39      withCredentials([string(credentialsId: 'dockerPass', variable: 'dockerPassword')]) {
40        sh "echo ${dockerPassword} | docker login -u vijay63833 --password-stdin"
41        sh "docker push vijay63833/helm-rockets:v1.1"
42      }
43    }
44  }
45  }
46  }
47  
```

Use Groovy Sandbox ?

Pipeline Syntax

Save Apply

✓ Saved REST API Jenkins 2.484



Not secure 54.178.66.43:8080/job/CICD-Pipeline-Project/9/console

Dashboard > CICD-Pipeline-Project > #9

```

0d5f5a015e5d: Layer already exists
3c777d951de2: Layer already exists
f8a91dd5fc84: Layer already exists
cb81227abde5: Layer already exists
e01a454893a9: Layer already exists
075d5b1b9460: Pushed
9585a8a2098c: Pushed
c45660adde37: Layer already exists
fe0fb3ab4a0f: Layer already exists
f1186e5061f2: Layer already exists
b2dba7477754: Layer already exists
991fb2ff565d: Pushed
7f1e7c90a710: Pushed
v1.1: digest: sha256:5510e4b8928c4a3c5826953bfe4bbe0e50bd3389624f0c86b27b23100f61341a size: 3051
[Pipeline] 
[Pipeline] // withCredentials
[Pipeline] 
[Pipeline] // script
[Pipeline] 
[Pipeline] // stage
[Pipeline] 
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
  
```

REST API Jenkins 2.484

CREATE KUBERNETES CLUSTER IN AWS USE KOPS

KOPS BINARY SETUP

```
# curl -Lo kops https://github.com/kubernetes/kops/releases/download/$(curl -s https://api.github.com/repos/kubernetes/kops/releases/latest | grep tag_name | cut -d '"' -f 4)/kops-linux-amd64  
  
# chmod +x ./kops  
  
# sudo mv ./kops /usr/local/bin/
```

KUBECTL BINARY SETUP

```
# curl -Lo kubectl https://storage.googleapis.com/kubernetes-release/release/$(curl -s https://storage.googleapis.com/kubernetes-release/release/stable.txt)/bin/linux/amd64/kubectl  
  
# chmod +x ./kubectl  
  
# sudo mv ./kubectl /usr/local/bin/kubectl
```

SETUP IAM USER (kops access aws resources)

```
# aws iam create-group --group-name kops  
# aws iam attach-group-policy --policy-arn  
arn:aws:iam::aws:policy/AmazonEC2FullAccess --group-name kops  
# aws iam attach-group-policy --policy-arn  
arn:aws:iam::aws:policy/AmazonRoute53FullAccess --group-name kops  
# aws iam attach-group-policy --policy-arn  
arn:aws:iam::aws:policy/AmazonS3FullAccess --group-name kops  
# aws iam attach-group-policy --policy-arn  
arn:aws:iam::aws:policy/IAMFullAccess --group-name kops  
# aws iam attach-group-policy --policy-arn  
arn:aws:iam::aws:policy/AmazonVPCFullAccess --group-name kops  
# aws iam create-user --user-name kops
```

```
# aws iam add-user-to-group --user-name kops --group-name kops  
# aws iam create-access-key --user-name kops
```

Cluster State storage

```
# aws s3api create-bucket --bucket awsdamo --region us-east-1  
# aws s3api put-bucket-versioning --bucket awsdamo --versioning-configuration  
Status=Enabled
```

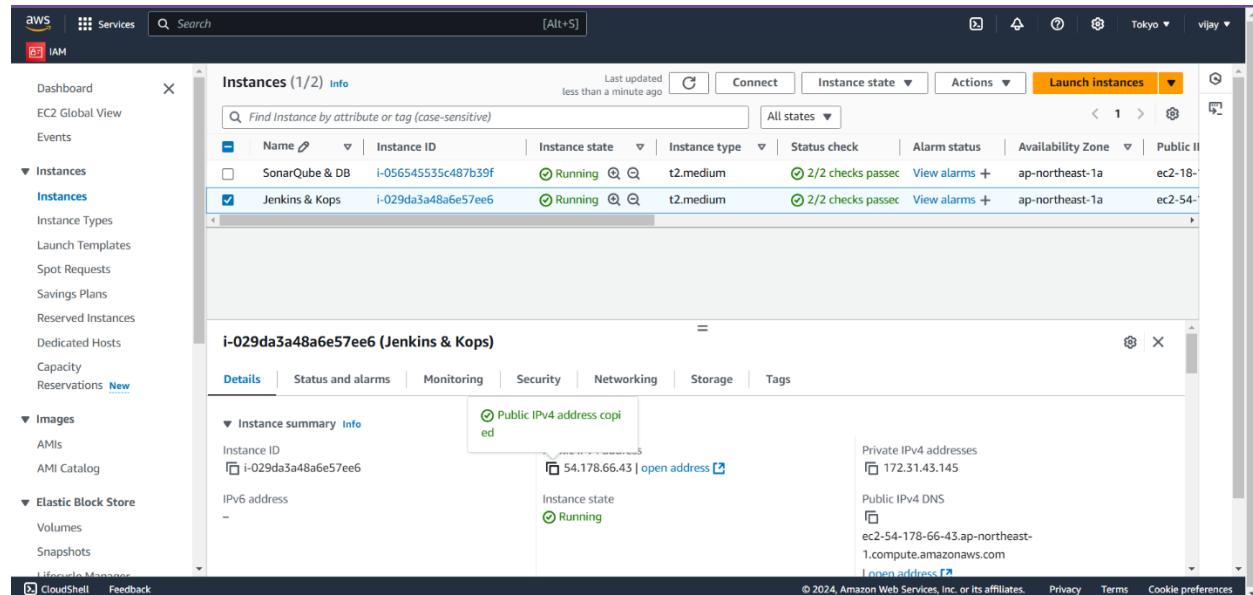
nodes authentication methods

create sshkey for machines

```
# ssh-keygen
```

Creating cluster:-

```
# export NAME=kops.k8s.local  
# export KOPS_STATE_STORE=s3://kops2914  
  
# kops create cluster --zones us-east-1a ${NAME}
```



```

root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# curl -Lo kops https://github.com/kubernetes/kops/releases/download/v1.23.0/kops-linux-amd64
root@ip-172-31-43-145:~# curl -s https://api.github.com/repos/kubernetes/kops/releases/latest | grep tag_name | cut -d ' '
Total  Received % Xferd  Average Speed   Time   Time  Current
          Dload Upload Total Spent  Left Speed
0     0    0    0      0      0 0:00:06 0:00:06 0:00:06 42.3M
100 238M 100 238M 0      0 37.3M 0 0:00:06 0:00:06 0:00:06 42.3M
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# chmod +x ./kops
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# sudo mv ./kops /usr/local/bin/
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# curl -Lo kubectl https://storage.googleapis.com/kubernetes-release/release/v1.23.0/bin/linux/amd64/kubectl
Total  Received % Xferd  Average Speed   Time   Time  Current
          Dload Upload Total Spent  Left Speed
100 53.7M 100 53.7M 0      0 20.2M 0 0:00:02 0:00:02 0:00:02 20.2M
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# chmod +x ./kubectl
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# sudo mv ./kubectl /usr/local/bin/kubectl
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~#

```

Screenshot of the AWS IAM console showing the "My security credentials" page for the "vijay" user.

The "Account details" section shows:

- Account name: vijay
- Email address: vijayavijayan145@gmail.com
- AWS account ID: 396913745209
- Canonical user ID: 14ad164f6913d209b15806ae7d3b45f9ae384bfda00752f1142f8c1272ddc903

The "Multi-factor authentication (MFA)" section shows:

- 0 MFA devices assigned.
- Buttons: Remove, Resync, Assign MFA device.

The "Access key created" section shows:

- Message: "Access key created" and "This is the only time that the secret access key can be viewed or downloaded. You cannot recover it later. However, you can create a new access key any time."
- Step 2: "Retrieve access key".
- Access key table:

Access key	Secret access key
AKIAVY2PHDU4QE67EVMM	***** Show

- Access key best practices:

 - Never store your access key in plain text, in a code repository, or in code.
 - Disable or delete access key when no longer needed.
 - Enable least-privilege permissions.
 - Rotate access keys regularly.

Screenshot of the AWS IAM console showing the "Access key created" confirmation message.

The "Access key" section shows:

- Access key: AKIAVY2PHDU4QE67EVMM
- Secret access key: ***** Show

The "Access key best practices" section contains the same list as the previous screenshot.

Buttons at the bottom:

- Download .csv file
- Done

Page footer:

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

```
root@ip-172-31-43-145:~# aws configure
Command 'aws' not found, but can be installed with:
snap install aws-cli # version 1.15.58, or
apt install awscli # version 2.14.6-1
See 'snap info aws-cli' for additional versions.
root@ip-172-31-43-145:~# ^C
root@ip-172-31-43-145:~# snap install aws-cli
error: The revision of snap "aws-cli" was published using classic confinement and thus may perform
arbitrary system calls outside of the security sandbox that snaps are usually confined to,
which may put your system at risk.

If you understand and want to proceed repeat the command including --classic.
root@ip-172-31-43-145:~# snap install aws-cli --classic
aws-cli (v2/stable) 2.19.3 from Amazon Web Services (aws-) installed
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# aws configure
AWS Access Key ID [None]: AKIAIVY2PHDU4QB67EVMM
AWS Secret Access Key [None]: IVM4abkKOZg2CuRUUFKk6OLwrJdErCPSESELPOJR
Default region name [None]: json
Default output format [None]: ap-northeast-1
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~#
```

```
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# aws iam create-group --group-name kops
{
    "Group": {
        "Path": "/",
        "GroupName": "kops",
        "GroupId": "AGFAYV2PHDU457YADD32K",
        "Arn": "arn:aws:iam::396913745209:group/kops",
        "CreateDate": "2024-11-08T03:41:44+00:00"
    }
}
root@ip-172-31-43-145:~# aws iam attach-group-policy --policy-arm arn:aws:iam::aws:policy/AmazonEC2FullAccess --group-name kops
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# aws iam attach-group-policy --policy-arm arn:aws:iam::aws:policy/AmazonRoute53FullAccess --group-name kops
root@ip-172-31-43-145:~# aws iam attach-group-policy --policy-arm arn:aws:iam::aws:policy/AmazonS3FullAccess --group-name kops
root@ip-172-31-43-145:~# aws iam attach-group-policy --policy-arm arn:aws:iam::aws:policy/IAMFullAccess --group-name kops
root@ip-172-31-43-145:~# aws iam attach-group-policy --policy-arm arn:aws:iam::aws:policy/AmazonVPCFullAccess --group-name kops
root@ip-172-31-43-145:~# aws iam create-user --user-name kops
{
    "User": {
        "Path": "/",
        "UserName": "kops",
        "UserId": "AIDAVY2PHDU45U4QSPFFKN",
        "Arn": "arn:aws:iam::396913745209:user/kops",
        "CreateDate": "2024-11-08T03:41:21+00:00"
    }
}
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# aws iam add-user-to-group --user-name kops --group-name kops
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# aws iam create-access-key --user-name kops
{
    "AccessKey": {
        "UserName": "kops",
        "AccessKeyId": "AKIAIVY2PHDU4UU5NDPG",
        "Status": "Active",
        "SecretAccessKey": "jmoYGlUwxe+UzEJF78Qs+lbbMnVPvYRHFGs/",
        "CreateDate": "2024-11-08T03:43:51+00:00"
    }
}
root@ip-172-31-43-145:~#
```

```
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# export AWS_ACCESS_KEY_ID=AKIAIVY2PHDU4QB67EVMM
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# export AWS_SECRET_ACCESS_KEY=IVM4abkKOZg2CuRUUFKk6OLwrJdErCPSESELPOJR
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~#
```

```
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# aws s3api create-bucket --bucket kops2914 --region ap-northeast-1 --create-bucket-configuration LocationConstraint=ap-northeast-1
{
    "Location": "http://kops2914.s3.amazonaws.com/"
}
root@ip-172-31-43-145:~# aws s3api put-bucket-versioning --bucket awsdamo --versioning-configuration Status=Enabled
An error occurred (AllAccessDisabled) when calling the PutBucketVersioning operation: All access to this object has been disabled
root@ip-172-31-43-145:~# ^C
root@ip-172-31-43-145:~# aws s3api put-bucket-versioning --bucket awsdamo --versioning-configuration Status=Enabled
An error occurred (AllAccessDisabled) when calling the PutBucketVersioning operation: All access to this object has been disabled
root@ip-172-31-43-145:~#
```

```
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# ssh-keygen
Generating public/private ed25519 key pair.
Enter file in which to save the key (/root/.ssh/id_ed25519):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_ed25519
Your public key has been saved in /root/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:7DIXEWs193m1f69zrVJphlrvV4BtmNX/F6iKhs+oOKM root@ip-172-31-43-145
The key's randomart image is:
+--[ED25519 256]--+
|          .+ |
|          . o...|
|          .+.+o.o|
|          ..Bo.++|
|          S. o.+o'|
|          ... .+o'|
|          . o.+ .+o'|
|          o..oo ..oo +|
|          +.++ . . .|
+---[SHA256]-----+
root@ip-172-31-43-145:~#
```

```
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# export NAME=kops.k8s.local
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# export KOPS_STATE_STORE=s3://kops2914
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~#
```

```
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# export NAME=kops.k8s.local
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# export KOPS_STATE_STORE=s3://kops2914
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# kops create cluster --zones ap-northeast-1a ${NAME}
W1108 03:53:54.944749 19269 new cluster.go:1435) Gossip is deprecated, using None DNS instead
I1108 03:53:54.944824 19269 new cluster.go:1454) Cloud Provider ID: "aws"
I1108 03:53:56.110680 19269 subnets.go:224) Assigned CIDR 172.20.0.0/16 to subnet ap-northeast-1a
Previewing changes that will be made:

I1108 03:54:00.878369 19269 builder.go:312] asset "https://dl.k8s.io/release/v1.30.2/bin/linux/amd64/kubelet" is not well-known, downloading hash
I1108 03:54:01.228942 19269 builder.go:312] asset "https://dl.k8s.io/release/v1.30.2/bin/linux/amd64/kubectl" is not well-known, downloading hash
I1108 03:54:01.463863 19269 builder.go:312] asset "https://github.com/containerd/containerd/releases/download/v1.7.16/containerd-1.7.16-linux-amd64.tar.gz" is not well-known, downloading hash
I1108 03:54:02.456321 19269 builder.go:312] asset "https://artifacts.k8s.io/binaries/kops/1.30.1/linux/amd64/nodeup" is not well-known, downloading hash
I1108 03:54:02.695612 19269 builder.go:312] asset "https://dl.k8s.io/release/v1.30.2/bin/linux/arm64/kubectl" is not well-known, downloading hash
I1108 03:54:03.015182 19269 builder.go:312] asset "https://dl.k8s.io/release/v1.30.2/bin/linux/arm64/kubelet" is not well-known, downloading hash
I1108 03:54:03.409496 19269 builder.go:312] asset "https://github.com/containerd/containerd/releases/download/v1.7.16/containerd-1.7.16-linux-arm64.tar.gz" is not well-known, downloading hash
I1108 03:54:04.818413 19269 builder.go:312] asset "https://artifacts.k8s.io/binaries/kops/1.30.1/linux/arm64/nodeup" is not well-known, downloading hash
I1108 03:54:04.818413 19269 builder.go:312] asset "https://artifacts.k8s.io/binaries/kops/1.30.1/linux/arm64/nodeup" is not well-known, downloading hash
```

```
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# kops validate cluster
Using cluster from kubectl context: kops.k8s.local
Validating cluster kops.k8s.local

INSTANCE GROUPS
NAME           ROLE      MACHINETYPE   MIN   MAX   SUBNETS
control-plane-ap-northeast-1a   ControlPlane  t3.medium    1     1     ap-northeast-1a
nodes-ap-northeast-1a          Node        t3.medium    1     1     ap-northeast-1a

NODE STATUS
NAME           ROLE      READY
i-09e609462b73371d5  control-plane  True
i-0c673b4b141d480d  node        True

Your cluster kops.k8s.local is ready
root@ip-172-31-43-145:~#
```

Step 6: Deploy to Kubernetes Worker Node

- 1. Configure Kubernetes Plugin in Jenkins:**
 - Go to Manage Jenkins > Manage Plugins > Available and search for **Kubernetes Plugin**.
 - Install the plugin and restart Jenkins.
- 2. Set Up Kubernetes Credentials:**
 - Go to Manage Jenkins > Configure System.
 - Add your Kubernetes cluster credentials (API URL and authentication token).
- 3. Deploy Docker Image to Kubernetes:**
 - In your Jenkins pipeline, add a step to deploy the Docker image to Kubernetes using kubectl:

```
stage('Deploy on Kubernetes') {  
    steps {  
        script {  
            withKubeConfig([credentialsId: 'kubernetes']) {  
                sh 'kubectl apply -f rocket-app.yml'  
            }  
        }  
    }  
}
```

Kubernetes Deployment YAML

- Ensure that you have a Kubernetes deployment.yaml file to handle deployment.

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: rocket-app-deployment
spec:
  replicas: 3
  selector:
    matchLabels:
      app: devops-project
  template:
    metadata:
      labels:
        app: devops-project
    spec:
      containers:
        - name: helm-rocket-app
          image: vijay63833/helm-rockets:v1.1
          ports:
            - containerPort: 80
---
apiVersion: v1
kind: Service
metadata:
  name: rocket-app-service
spec:
  type: NodePort
  selector:
    app: devops-project
  ports:
    - port: 80
      targetPort: 80
      nodePort: 30007
```

The screenshot shows the AWS CloudWatch Instances console. On the left, a sidebar navigation includes: Dashboard, EC2 Global View, Events, Instances (selected), Instances Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity, and Reservations (New). The main area displays a table of instances with columns: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, and Public IP. One instance, 'Jenkins & Kops' (i-029da3a48a6e57ee6), is selected and highlighted in blue. Below the table, the instance details for 'i-029da3a48a6e57ee6 (Jenkins & Kops)' are shown, including its Details, Status and alarms, Monitoring, Security, Networking, Storage, and Tags. The public IPv4 address is listed as 54.178.66.43 | open address.

The screenshot shows the Jenkins Manage Jenkins interface. The left sidebar includes: + New Item, Build History, Project Relationship, Check File Fingerprint, Manage Jenkins (selected), My Views, Build Queue (No builds in the queue), and Build Executor Status (0/2). The main content area is titled 'Manage Jenkins'. It features a message: 'Building on the built-in node can be a security issue. You should set up distributed builds. See the documentation.' Below this are sections for System Configuration (System, Tools, Nodes, Managed files), Plugins (Plugins, Add, remove, disable or enable plugins that can extend the functionality of Jenkins), and Appearance (Configure the look and feel of Jenkins). A 'Search settings' bar is at the top right.

The screenshot shows the Jenkins Manage Jenkins > Plugins interface. The left sidebar includes: Updates, Available plugins (selected), Installed plugins, Advanced settings, and Download progress. The main content area is titled 'Plugins' and shows a search bar with 'kubernetes'. A table lists available plugins: 'Kubernetes Client API' (version 6.10.0-240.v57880ce8b_0b_2, released 9 mo 16 days ago), 'Kubernetes Credentials' (version 190.v03c305394deb_, released 1 mo 20 days ago), 'Kubernetes CLI' (version 1.12.1, released 2 days 18 hr ago), and 'Kubernetes Credentials Provider' (version 1.262.v2670ef7ea_0c5, released 8 mo 12 days ago). The 'Available plugins' tab is selected.

← → 🔍 Not secure 54.178.66.43:8080/manage/pluginManager/updates/

Jenkins

Dashboard > Manage Jenkins > Plugins

Plugins	Download progress
Updates	Preparation
Available plugins	<ul style="list-style-type: none"> • Checking internet connectivity • Checking update center connectivity • Success
Installed plugins	<ul style="list-style-type: none"> Config File Provider ✓ Success NodeJS ✓ Success Loading plugin extensions ✓ Success Kubernetes Client API ✓ Success Authentication Tokens API ✓ Success Kubernetes Credentials ✓ Success Kubernetes ✓ Success Kubernetes CLI ✓ Success Loading plugin extensions ✓ Success
Advanced settings	
Download progress	

→ Go back to the top page
(you can start using the installed plugins right away)

→ Restart Jenkins when installation is complete and no jobs are running

```
root@ip-172-31-43-145:~/kube
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# ls -lrlta
total 44
-rw-r--r-- 1 root root 161 Apr 22 2024 .profile
-rw-r--r-- 1 root root 3106 Apr 22 2024 .bashrc
drwxr-xr-x 22 root root 4096 Nov 7 13:44 ..
-rw-r--r-- 1 root root 164 Nov 7 13:51 .wget-hsts
-rw-r----- 1 root root 807 Nov 7 19:28 .bash_history
drwxr----- 4 root root 4096 Nov 8 03:36 snap
drwxr-xr-x 2 root root 4096 Nov 8 03:39 .aws
-rw-r----- 1 root root 20 Nov 8 03:51 lesshist
drwxr-xr-x 2 root root 4096 Nov 8 03:51 .ssh
drwxr-xr-x 6 root root 4096 Nov 8 03:51 .
drwxr-xr-x 3 root root 4096 Nov 8 04:20 .kube
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# cd .kube
root@ip-172-31-43-145:~/.kube# ls
cache config
root@ip-172-31-43-145:~/.kube# cat config
apiVersion: vi
clusters:
- cluster:
  certificate-authority-data: LS0tLS1CRU4dTjIBDRVJUSUZJQ0FURS0tLs0tCk1JSUJmRENDXWVBD0f3SUJB0lNwR0FyaCxsKzE0WW2zaDRB1lRmEdD03FUu01lM0rrRUJpd1vBTUJneEZgVUk2md0VkjBTVREV3QxW1WeKjEVjBaWe1l0w
X3YvPwlbGmlhSYOpERUY4emc3C25Pwv1XhK0KAuWU0chDUvRyTfxAfxzq09KwGRVbypubnEzVnNTFTjX0Mh5hz17St5b7BMMWFKyHe1g34d3SnktURxRkdRvpyMD7JWHZmNgNcpkMgR3Zx5Qbkh1NvSyVmlySWJNk2tSY2Zt5C7MmNOVWhInk
yenJzZDwkJN2dR0p25tcnhsVxJ7dw00o1daN1AxZv21I1dssjh3pFcckd4S2VGK1ESL0PqbwsL1hVzdwWTBzL1FEMk36Wv0vTcdnT2IK1LRMcnc5SOY373ptZUNBb25NeLdsavJvQ2ZfYjBpQUzaM021WK1SHz21V1kWxLm1cmhd1d3
ak5iacaQjn2F6WlJ0cEzlao56d2hka0dj0dUNNEKRBNRMSWEWbd1z0LzY0T1FpERFRQnJvMEL1uURB070n7z1IUTfrCCkPn0EVcQUL0qVF40R3WURUWjBQWVYf10LjVXb4D0VCL1pB2E5nt1Z1UTfrRmdrTBRWW0W0MVJYk1t2s1wSmkTWLNxFBd
ZRNBUv3RF2SktrwK1odmnQOQVF2TJ5QUn20VCQX4cktrvGNJQj132kBtU1FlWw04enFQdVkevgpW6pkl3JiMU2Nk29Pitjclhb2V1NHB4S1V5cmv1NDRCTk7TSngp0kdhM0R2YU1eF2pmzmzGRgL01Wnh2CmZREZXk2Nx-Erjb3RtEMyc0
XNmnbmElEEsbHV2U0NFnbzNoAvJ3bzFM5jDaWxKNWjNytS09XK0U5N35aTAKY1Mxd1Fc0t1lTUxy1M8SmfdNFndsqVdsSi1k0uVpeCtTy9znhKMg5fR0J507J4bfrZLdHoyL0JSSWh1VER1UpfF0szv3hj0k02MUCrd21XSFJ0WE4z1NNkT7yzLl1
CUU1ByvswWU12bi7vhUfahVedpVko1UHRC1L1Facit5cmx2NPdgwNz5yL3JPT3J2mmXVEFvbNx2Zgd3ckNqS0srjBxL0x4NmlNell6czqybkd1tMriaHRzPqotLS0tUVORCBDRVJUSUJQ0fURs0tLS0tCg==
server: https://api-kops-k8s-local-airamg-Sa5e405c46bc78.elb.ap-northeast-1.amazonaws.com
  tls-server-name: api.internal.kops.k8s.local
  name: kops.k8s.local
contexts:
- context:
  cluster: kops.k8s.local
  user: kops.k8s.local
  name: kops.k8s.local
current-context: kops.k8s.local
kind: Config
preferences: {}
users:
- name: kops.k8s.local
  user:
    client-certificate-data: LS0tLS1CRU4dTjIBDRVJUSUZJQ0FURS0tLs0tCk1JSUJmRENDXWVBD0f3SUJB0lNwR0FyaCxsKzE0Wa20vaXd0MDB2TytJbnhQbkf3RFFZSkvtWkldmNQVFFTEJRQXcK0URFV01CUUdbMVVFQXhNTmEzVm1aWEpi1Wls
bGNSMwZVFEf1rcweU5ERXHNRf13TpVNE1E2GFqd5B5TkrFeAgNRg57VRVNE1E2GFNRF4RnlpVkJn1zCq95UrG5CNMzUmxiVHb0WV1ONPfYspNU1Vrxzd2RF2RURF03hyCmXSmxZM1puTfRkdm1zUxdn20VpTUEwR0NtCdrSwIzRFFQkFRV
UFbNs1CRHd2d2dnRtBh01CqVFEfPh0ndgKd1pSeHJG0U5u2h2gcnzsRhe0Mu9R1lR1zyt0dJNYK0Y5mcrlzA2d2h0sNzF81RfaDRGV1fPwWg2XNFQnhzWAp0S0NWSjBkzMyeEdxR3VW0HN3eStirHBBcW16qg3ZD2Rmpu0n7qT0NMVFpAkEvWT
```

Not secure 54.178.66.43:8080/manage/

Dashboard > Manage Jenkins

No builds in the queue.

Build Executor Status 0/2

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

Clouds
Add, remove, and configure cloud instances to provision agents on-demand.

Appearance
Configure the look and feel of Jenkins

Managed files
e.g. settings.xml for maven, central managed scripts, custom files, ...

Security

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

Credentials
Configure credentials

Credential Providers
Configure the credential providers and types

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

Status Information

System Information
Displays various environmental information to assist trouble-shooting.

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

Load Statistics
Check your resource utilization and see if you need more computers for your builds.

54.178.66.43:8080/manage/credentials

Not secure 54.178.66.43:8080/manage/credentials/

Jenkins

Dashboard > Manage Jenkins > Credentials

Credentials

T	P	Store	Domain	ID	Name
System	System	(global)	sonar-pro	SonarPWD	
System	System	(global)	dockerPass	dockerpassword	

Stores scoped to Jenkins

P	Store	Domains
System	System	(global)

Icon: S M L

Not secure 54.178.66.43:8080/manage/credentials/store/system/

Jenkins

Dashboard > Manage Jenkins > Credentials > System >

System

+ Add domain

Domain	Description
Global credentials (unrestricted)	Credentials that should be available irrespective of domain specification to requirements matching.

Icon: S M L

54.178.66.43:8080/manage/credentials/store/system/newDomain REST API Jenkins 2.484

Not secure 54.178.66.43:8080/manage/credentials/store/system/domain/_/

Jenkins

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

Global credentials (unrestricted)

+ Add Credentials

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

ID	Name	Kind	Description	Actions
sonar-pro	SonarPWD	Secret text	SonarPWD	
dockerPass	dockerpassword	Secret text	dockerpassword	

Icon: S M L

54.178.66.43:8080/manage/credentials/store/system/domain/_/newCredentials REST API Jenkins 2.484

Not secure 54.178.66.43:8080/manage/credentials/store/system/domain/_/newCredentials

Jenkins

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

New credentials

Kind

Secret file

Scope ?

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

File

Choose File kubeconfiguration.txt

ID ?

kubernetes

Description ?

k8spwd

Create

Not secure 54.178.66.43:8080/manage/credentials/store/system/domain/ /

Jenkins

Search (CTRL+K) Vijay log out

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

Global credentials (unrestricted)

+ Add Credentials

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

ID	Name	Kind	Description
sonar-pro	SonarPWD	Secret text	SonarPWD
dockerPass	dockerpassword	Secret text	dockerpassword
kubernetes	kubeconfiguration.txt (k8spwd)	Secret file	k8spwd

Icon: S M L

REST API Jenkins 2.484

Not secure 54.178.66.43:8080

Jenkins

Search (CTRL+K) Vijay log out

Dashboard >

+ New Item

Build History All +

Project Relationship

Check File Fingerprint

Manage Jenkins

My Views

Build Queue Icon: S M L No builds in the queue.

Build Executor Status 0/2

54.178.66.43:8080/job/CICD-Pipeline-Project/ REST API Jenkins 2.484

Not secure 54.178.66.43:8080/job/CICD-Pipeline-Project/configure

Dashboard > CICD-Pipeline-Project > Configuration

Configure

Definition

Pipeline script

General

Build Triggers

Advanced Project Options

Pipeline

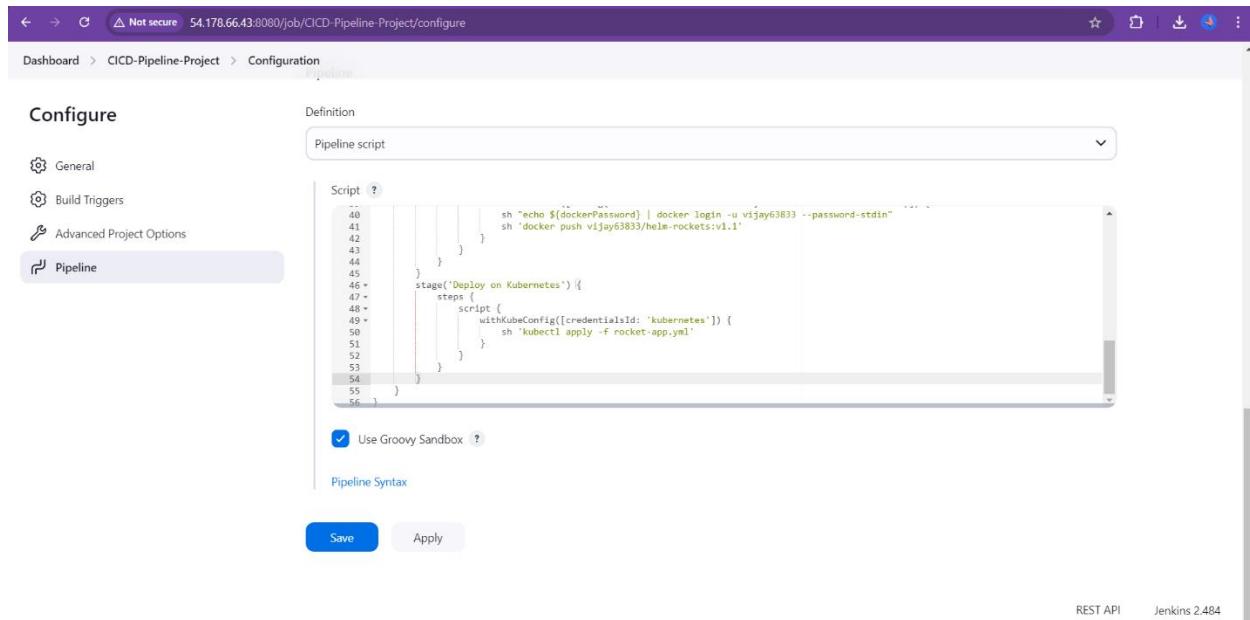
```
40 sh "echo ${dockerPassword} | docker login -u vijay63833 --password-stdin"
41
42 }
43 }
44 }
45 }
46 stage('Deploy on Kubernetes') {
47 steps {
48 script {
49 withKubeConfig([credentialsId: 'kubernetes']) {
50 sh 'kubectl apply -f rocket-app.yaml'
51 }
52 }
53 }
54 }
55 }
```

Use Groovy Sandbox ?

Pipeline Syntax

Save Apply

REST API Jenkins 2.484



Not secure 54.178.66.43:8080/job/CICD-Pipeline-Project/10/pipeline-graph/

Jenkins

Dashboard > CICD-Pipeline-Project > #10 > Pipeline Overview

Build #10

Pipeline

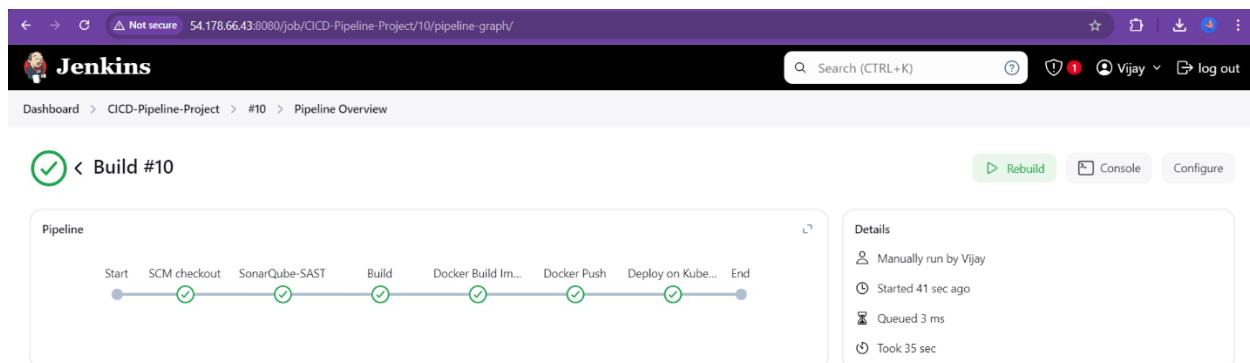
```
graph LR; Start((Start)) --> SCM_checkout((SCM checkout)); SCM_checkout --> SonarQube_SAST((SonarQube-SAST)); SonarQube_SAST --> Build((Build)); Build --> Docker_Build_Image((Docker Build Image)); Docker_Build_Image --> Docker_Push((Docker Push)); Docker_Push --> Deploy_on_Kubernetes((Deploy on Kubernetes)); Deploy_on_Kubernetes --> End((End))
```

Rebuild Console Configure

Details

- Manually run by Vijay
- Started 41 sec ago
- Queued 3 ms
- Took 35 sec

Jenkins 2.484



Not secure 54.178.66.43:8080/job/CICD-Pipeline-Project/10/console

Dashboard > CICD-Pipeline-Project > #10

```
[Pipeline] // script
[Pipeline]
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Deploy on Kubernetes)
[Pipeline] script
[Pipeline] {
[Pipeline] withKubeConfig
[Pipeline] {
[Pipeline] sh
+ kubectl apply -f rocket-app.yaml
deployment.apps/rocket-app-deployment created
service/rocket-app-service created
[Pipeline]
[kubernetes-cli] kubectl configuration cleaned up
[Pipeline] // withKubeConfig
[Pipeline]
[Pipeline] // script
[Pipeline]
[Pipeline] // stage
[Pipeline]
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```

REST API Jenkins 2.484

root@ip-172-31-43-145:~#

```
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
rocket-app-deployment-76cf8cc646-tvsvl  1/1    Running   0          50s
rocket-app-deployment-76cf8cc646-vq8tv   1/1    Running   0          50s
rocket-app-deployment-76cf8cc646-vsxms  1/1    Running   0          50s
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# kubectl get rs
NAME          DESIRED   CURRENT   READY   AGE
rocket-app-deployment-76cf8cc646   3         3        3      72s
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# kubectl get deployment
NAME          READY   UP-TO-DATE   AVAILABLE   AGE
rocket-app-deployment  3/3     3           3          87s
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~#
root@ip-172-31-43-145:~# kubectl get pods -o wide
NAME          READY   STATUS    RESTARTS   AGE   IP           NODE   NOMINATED NODE   READINESS GATES
rocket-app-deployment-76cf8cc646-tvsvl  1/1    Running   0      100.96.1.67  i-0c1673b4b141d480d  <none>   <none>
rocket-app-deployment-76cf8cc646-vq8tv   1/1    Running   0      100.96.1.211 i-0c1673b4b141d480d  <none>   <none>
rocket-app-deployment-76cf8cc646-vsxms  1/1    Running   0      100.96.1.15   i-0c1673b4b141d480d  <none>   <none>
```

Screenshot of the AWS EC2 Instances page showing a list of running instances. The instance **i-0c1673b4b141d480d** (nodes-ap-nort...), which is selected, is a t3.medium type running in ap-northeast-1a with a public IP ec2-18-18-12-11. It has 3/3 checks passed.

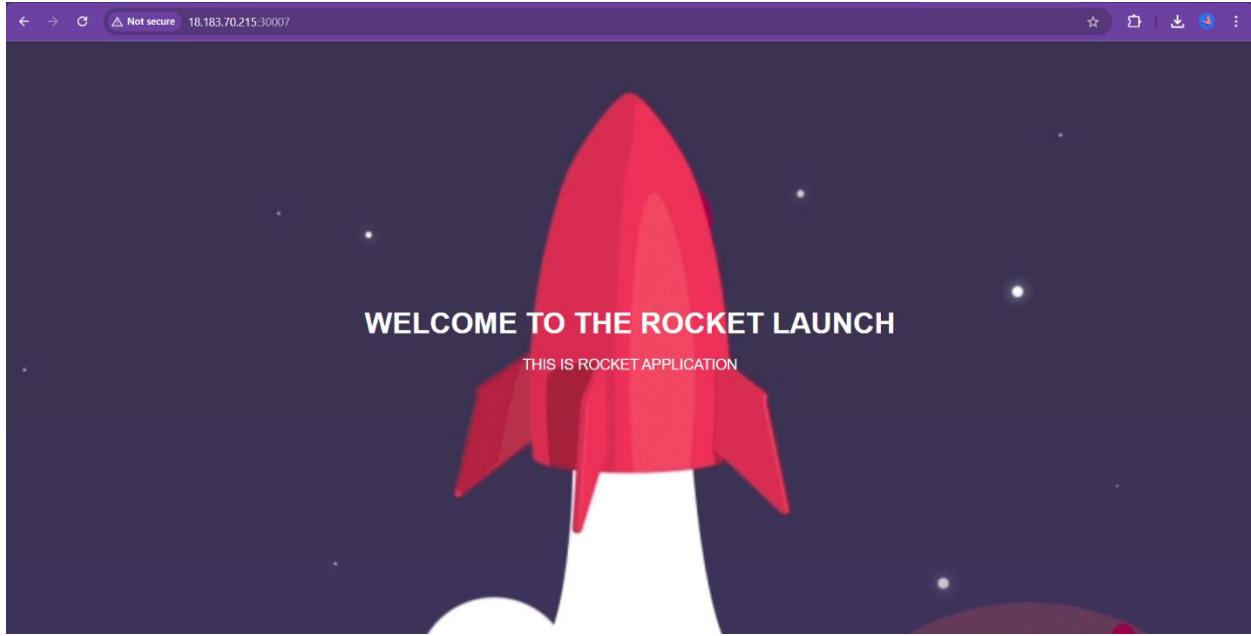
Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
SonarQube & DB	i-056545535c487b39f	Running	t2.medium	2/2 checks passed	View alarms	ap-northeast-1a	ec2-18-18-12-11
nodes-ap-nort...	i-0c1673b4b141d480d	Running	t3.medium	3/3 checks passed	View alarms	ap-northeast-1a	ec2-18-18-12-11
Jenkins & Kops	i-029da3a4bae57ee6	Running	t2.medium	2/2 checks passed	View alarms	ap-northeast-1a	ec2-54-17-11-11
control-plane....	i-09c609462b73371d5	Running	t3.medium	3/3 checks passed	View alarms	ap-northeast-1a	ec2-13-11-11-11

The security details for the selected instance show no roles attached to the instance profile. The launch time was Fri Nov 08 2024 09:28:15 GMT+0530 (India Standard Time).

Screenshot of the AWS Security Groups page showing the modify inbound security group rules for security group sg-077a04f0e0c7d6a67. The table lists five rules:

Security group rule ID	Type	Protocol	Port range	Source	Description - optional
sgr-0953ecd0fcfc109f	SSH	TCP	22	Custom	sg-077a04f0e0c7d6a67
sgr-07a6a0b6020dbf8a4	SSH	TCP	22	Custom	0.0.0.0/0
sgr-0ace2fe2ae42c08d9	All traffic	All	All	Custom	sg-077a04f0e0c7d6a67
sgr-010465862f0b7a2c4	All traffic	All	All	Custom	sg-06783c77fa34a8cdf
-	Custom TCP	TCP	30007	Anyw...	0.0.0.0/0

A warning message at the bottom states: "Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only." There is also a note about the security group rule sg-077a04f0e0c7d6a67 being deleted.



This CI/CD pipeline setup automates the process of checking out the Node.js code from GitHub, performing static analysis using SonarQube, building the application, creating and pushing Docker images, and deploying the images to a Kubernetes cluster. Each step is connected in a Jenkins pipeline that ensures the automated deployment of the application to a production environment.
