

## 1. Add 8080 in your EC2 instance security group. Install corretto.

```
us-east-1.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-01293611fbbdd8742&osUser=ec2-user&region=us-east-1&sshPort=22#/

[ec2-user@ip-172-31-48-149 ~]$ sudo dnf update
Last metadata expiration check: 22:05:28 ago on Thu May 30 20:02:57 2024.

WARNING:
A newer release of "Amazon Linux" is available.

Available Versions:

Version 2023.4.20240528:
Run the following command to upgrade to 2023.4.20240528:

dnf upgrade --releasever=2023.4.20240528

Release notes:
https://docs.aws.amazon.com/linux/al2023/release-notes/relnotes-2023.4.20240528.html

Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-172-31-48-149 ~]$ sudo dnf install java-17-amazon-corretto -y
Last metadata expiration check: 22:05:47 ago on Thu May 30 20:02:57 2024.
Dependencies resolved.

Package Architecture Version Repository Size
Installing:

i-01293611fbbdd8742 (skillsmatch-k8)
PublicIPs: 54.221.176.114 PrivateIPs: 172.31.48.149

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

us-east-1.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-01293611fbbdd8742&osUser=ec2-user&region=us-east-1&sshPort=22#/

[ec2-user@ip-172-31-48-149 ~]$ sudo dnf install java-17-amazon-corretto -y
Last metadata expiration check: 22:05:47 ago on Thu May 30 20:02:57 2024.
Dependencies resolved.

Package Architecture Version Repository Size
Installing:
java-17-amazon-corretto x86_64 1:17.0.11+9-1.amzn2023.1 amazonlinux 187
Installing dependencies:
java-17-amazon-corretto-headless x86_64 1:17.0.11+9-1.amzn2023.1 amazonlinux 91

Transaction Summary
Install 2 Packages

Total download size: 91 M
Installed size: 236 M
Downloading Packages:
(1/2): java-17-amazon-corretto-17.0.11+9-1.amzn2023.1.x86_64.rpm 2.0 MB/s | 187 KB 00:00
(2/2): java-17-amazon-corretto-headless-17.0.11+9-1.amzn2023.1.x86_64.rpm 62 MB/s | 91 MB 00:01
Total 60 MB/s | 91 MB 00:01
Amazon Linux 2023 repository
Importing GPG key 0x0832C631:
Userid : "Amazon Linux <amazon-linux@amazon.com>"
Fingerprint: B21C 50FA 44A9 9720 EAA7 2F7F E951 904A D832 C631
From : /etc/pki/rpm-gpg/RPM-GPG-KEY-amazon-linux-2023
Key imported successfully
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.

i-01293611fbbdd8742 (skillsmatch-k8)
PublicIPs: 54.221.176.114 PrivateIPs: 172.31.48.149

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

## 2. Check java version

```
Installed:
java-17-amazon-corretto-1:17.0.11+9-1.amzn2023.1.x86_64 java-17-amazon-corretto-headless-1:17.0.11+9-1.amzn2023.1.x86_64

Complete!
[ec2-user@ip-172-31-48-149 ~]$ java -version
openjdk version "17.0.11" 2024-04-16 LTS
OpenJDK Runtime Environment Corretto-17.0.11.9.1 (build 17.0.11+9-LTS)
OpenJDK 64-Bit Server VM Corretto-17.0.11.9.1 (build 17.0.11+9-LTS, mixed mode, sharing)
[ec2-user@ip-172-31-48-149 ~]$

i-01293611fbbdd8742 (skillsmatch-k8)
PublicIPs: 54.221.176.114 PrivateIPs: 172.31.48.149
```

### 3. Install Jenkins on the AWS instance.

```
aws
Services Search [Alt+S]
[ec2-user@ip-172-31-48-149 ~]$ sudo wget -O /etc/yum.repos.d/jenkins.repo \
https://pkg.jenkins.io/redhat-stable/jenkins.repo
--2024-05-31 18:13:04-- https://pkg.jenkins.io/redhat-stable/jenkins.repo
Resolving pkg.jenkins.io (pkg.jenkins.io)... 146.75.34.133, 2a04:4e42:77::645
Connecting to pkg.jenkins.io (pkg.jenkins.io)[146.75.34.133]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 85
Saving to: '/etc/yum.repos.d/jenkins.repo'

/etc/yum.repos.d/jenkins.repo 100%[=====>] 85 --.-KB/s in 0s

2024-05-31 18:13:04 (6.65 MB/s) - '/etc/yum.repos.d/jenkins.repo' saved [85/85]

[ec2-user@ip-172-31-48-149 ~]$ sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io-2023.key
[ec2-user@ip-172-31-48-149 ~]$ sudo dnf install jenkins -y
Last metadata expiration check: 22:30:17 ago on Thu May 30 19:43:42 2024.
Package jenkins-2.460-1.1.noarch is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-172-31-48-149 ~]$ sudo systemctl enable jenkins
Created symlink /etc/systemd/system/multi-user.target.wants/jenkins.service - /usr/lib/systemd/system/jenkins.service.
[ec2-user@ip-172-31-48-149 ~]$ sudo systemctl start jenkins
[ec2-user@ip-172-31-48-149 ~]$ sudo systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
   Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; preset: disabled)
   Active: active (running) since Thu 2024-05-30 20:04:09 UTC; 22h ago
```

i-01293611fbbdd8742 (skillsmatch-k8)

PublicIPs: 54.221.176.114 PrivateIPs: 172.31.48.149

### 4. Start Jenkins.

```
aws
Services Search [Alt+S]
Complete!
[ec2-user@ip-172-31-48-149 ~]$ sudo systemctl enable jenkins
Created symlink /etc/systemd/system/multi-user.target.wants/jenkins.service - /usr/lib/systemd/system/jenkins.service.
[ec2-user@ip-172-31-48-149 ~]$ sudo systemctl start jenkins
[ec2-user@ip-172-31-48-149 ~]$ sudo systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
   Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; preset: disabled)
   Active: active (running) since Thu 2024-05-30 20:04:09 UTC; 22h ago
     Main PID: 135901 (java)
       Tasks: 45 (limit: 9490)
      Memory: 596.7M
         CPU: 2min 8.851s
    CGroup: /system.slice/jenkins.service
            └─135901 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/var/cache/jenkins/war --httpPort=8080

May 30 20:04:06 ip-172-31-48-149.ec2.internal jenkins[135901]: 079aa87a76c74b5daa2b1b80f7ebe369
May 30 20:04:06 ip-172-31-48-149.ec2.internal jenkins[135901]: This may also be found at: /var/lib/jenkins/secrets/initialAdminPassword
May 30 20:04:06 ip-172-31-48-149.ec2.internal jenkins[135901]: *****
May 30 20:04:06 ip-172-31-48-149.ec2.internal jenkins[135901]: *****
May 30 20:04:06 ip-172-31-48-149.ec2.internal jenkins[135901]: *****
May 30 20:04:09 ip-172-31-48-149.ec2.internal jenkins[135901]: 2024-05-30 20:04:09.667+0000 [id=30] INFO jenkins.InitReactorRunner$1:onAttained: Comple
May 30 20:04:09 ip-172-31-48-149.ec2.internal jenkins[135901]: 2024-05-30 20:04:09.685+0000 [id=22] INFO hudson.lifecycle.Lifecycle#onReady: Jenkins is
May 30 20:04:09 ip-172-31-48-149.ec2.internal jenkins[135901]: 2024-05-30 20:04:09.729+0000 [id=47] INFO h.m.DownloadService$Downloadable#load: Obtained
May 30 20:04:09 ip-172-31-48-149.ec2.internal jenkins[135901]: 2024-05-30 20:04:09.730+0000 [id=47] INFO hudson.util.Retrier#start: Performed the action
```

i-01293611fbbdd8742 (skillsmatch-k8)

PublicIPs: 54.221.176.114 PrivateIPs: 172.31.48.149

### 5. Get Jenkins Password from the following:

```
aws
Services Search [Alt+S]
Created symlink /etc/systemd/system/multi-user.target.wants/jenkins.service - /usr/lib/systemd/system/jenkins.service.
[ec2-user@ip-172-31-48-149 ~]$ sudo systemctl start jenkins
[ec2-user@ip-172-31-48-149 ~]$ sudo systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
   Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; preset: disabled)
   Active: active (running) since Thu 2024-05-30 20:04:09 UTC; 22h ago
     Main PID: 135901 (java)
       Tasks: 45 (limit: 9490)
      Memory: 596.7M
         CPU: 2min 8.851s
    CGroup: /system.slice/jenkins.service
            └─135901 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/var/cache/jenkins/war --httpPort=8080

May 30 20:04:06 ip-172-31-48-149.ec2.internal jenkins[135901]: 079aa87a76c74b5daa2b1b80f7ebe369
May 30 20:04:06 ip-172-31-48-149.ec2.internal jenkins[135901]: This may also be found at: /var/lib/jenkins/secrets/initialAdminPassword
May 30 20:04:06 ip-172-31-48-149.ec2.internal jenkins[135901]: *****
May 30 20:04:06 ip-172-31-48-149.ec2.internal jenkins[135901]: *****
May 30 20:04:06 ip-172-31-48-149.ec2.internal jenkins[135901]: *****
May 30 20:04:09 ip-172-31-48-149.ec2.internal jenkins[135901]: 2024-05-30 20:04:09.667+0000 [id=30] INFO jenkins.InitReactorRunner$1:onAttained: Comple
May 30 20:04:09 ip-172-31-48-149.ec2.internal jenkins[135901]: 2024-05-30 20:04:09.685+0000 [id=22] INFO hudson.lifecycle.Lifecycle#onReady: Jenkins is
May 30 20:04:09 ip-172-31-48-149.ec2.internal jenkins[135901]: 2024-05-30 20:04:09.729+0000 [id=47] INFO h.m.DownloadService$Downloadable#load: Obtained
May 30 20:04:09 ip-172-31-48-149.ec2.internal jenkins[135901]: 2024-05-30 20:04:09.730+0000 [id=47] INFO hudson.util.Retrier#start: Performed the action

[ec2-user@ip-172-31-48-149 ~]$ sudo cat /var/lib/jenkins/secrets/initialAdminPassword
079aa87a76c74b5daa2b1b80f7ebe369
[ec2-user@ip-172-31-48-149 ~]$
```

i-01293611fbbdd8742 (skillsmatch-k8)

PublicIPs: 54.221.176.114 PrivateIPs: 172.31.48.149

## 6. Use the Public IP of the EC2 instance and Jenkins 8080 port to access Jenkins.

The top screenshot shows the Jenkins 2.460 'Getting Started' page. It features a grid of plugins to install, including Folders, OWASP Markup Formatter, Build Timeout, Credentials Binding, Timestampers, Workspace Cleanup, Ant, Gradle, Pipeline, GitHub Branch Source, Pipeline: GitHub Groovy Libraries, Pipeline Graph View, Git, SSH Build Agents, Matrix Authorization Strategy, PAM Authentication, LDAP, Email Extension, Mailer, and Dark Theme. A list of additional plugins is shown on the right, including Jakarta Activation API, Jakarta Mail API, Apache HttpComponents Client 4.x API, Mailer, Pipeline: Basic Steps, Gradle, Pipeline: Milestone Step, Pipeline: Build Step, Pipeline: Groovy Libraries, Pipeline: Stage Step, Joda Time API, Pipeline: Model API, Pipeline: Declarative Extension Points API, Branch API, Pipeline: Multibranch, Pipeline: Stage Tags Metadata, Pipeline: Input Step, Pipeline: Declarative Pipeline, Java JSON Web Token (JWT), and OAuth2.

The bottom screenshot shows the Jenkins 2.460 'Instance Configuration' page. It prompts the user to enter the Jenkins URL. The URL entered is `http://54.221.176.114:8080/`. The page includes a 'Not now' button and a 'Save and Finish' button.

## 7. Install the needed plugins for your pipelines.

The image displays three sequential screenshots of the Jenkins 'Manage Jenkins' > 'Plugins' interface, illustrating the process of installing various plugins for pipeline development.

**First Screenshot: Search for 'git'**

The search bar contains 'git'. The results list several plugins:

- GitLab Authentication 1.19** (Released 5 mo 1 day ago): Authentication and User Management. Description: This is the an authentication plugin using gitlab OAuth.
- GitHub Integration 0.7.0** (Released 3 mo 11 days ago): emailx Build Triggers. Description: GitHub Integration Plugin for Jenkins. (This plugin is checked for installation)
- GitHub Authentication 597.ve0c3480fcb\_d0** (Released 6 mo 29 days ago): github Security Authentication and User Management. Description: Authentication plugin using GitHub OAuth to provide authentication and authorization capabilities for GitHub and GitHub Enterprise.
- GitLab Branch Source 704.vc7f1202d7e14** (Released 28 days ago): Provides branch source and folder organization functionality for GitLab Repositories in Jenkins.
- GitHub Pull Request Builder 1.42.2** (Released 3 yr 3 mo ago): github Build Triggers. (Warning: This plugin version may not be safe to use. Please review the following security notices: CSRF vulnerability and missing permission checks, Missing permission check allows enumerating credentials IDs)

**Second Screenshot: Search for 'docker'**

The search bar contains 'docker'. The results list several plugins:

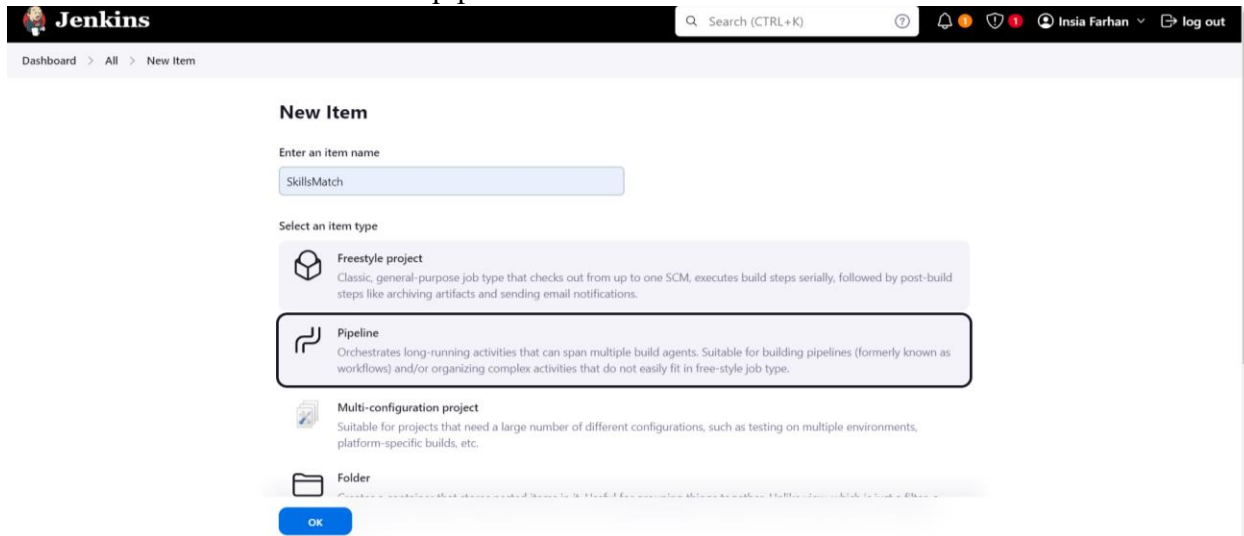
- Docker 1.6.1** (Released 18 days ago): Cloud Providers Cluster Management docker. Description: This plugin integrates Jenkins with Docker. (This plugin is checked for installation)
- Docker Commons 439.va\_3cb\_0a\_6a\_fb\_29** (Released 10 mo ago): Library plugins (for use by other plugins) docker. Description: Provides the common shared functionality for various Docker-related plugins.
- Docker Pipeline 580.vc0c340686b\_54** (Released 9 days 17 hr ago): pipeline DevOps Deployment docker. Description: Build and use Docker containers from pipelines. (This plugin is checked for installation)
- Docker API 3.3.4-86.v39b\_a\_5ede342c** (Released 6 mo 4 days ago): Library plugins (for use by other plugins) docker. Description: This plugin provides docker-java API for other plugins. (Note: This plugin is up for adoption! We are looking for new maintainers. Visit our Adopt a Plugin initiative for more information.)

**Third Screenshot: Search for 'Kubernetes'**

The search bar contains 'Kubernetes'. The results list several plugins:

- Kubernetes 4238.v41b\_3ef14a\_5d8** (Released 2 days 4 hr ago): Cloud Providers Cluster Management kubernetes Agent Management. Description: This plugin integrates Jenkins with Kubernetes. (This plugin is checked for installation)
- Kubernetes Client API 6.10.0-240.v57880ce8b\_0b\_2** (Released 4 mo 6 days ago): kubernetes Library plugins (for use by other plugins). Description: Kubernetes Client API plugin for use by other Jenkins plugins.
- Kubernetes Credentials 174.va\_36e093562d9** (Released 4 days 0 hr ago): kubernetes credentials. Description: Common classes for Kubernetes credentials.
- Kubernetes CLI 1.12.1** (Released 9 mo 5 days ago): kubernetes. Description: Configure kubectl for Kubernetes.
- Kubernetes Credentials Provider 1.262.v2670ef7ea\_0c5** (Released 3 mo 2 days ago): kubernetes credentials.

## 8. Create new item. Choose pipeline.



The Jenkins 'New Item' page is shown. The header includes the Jenkins logo, a search bar, and user information. The breadcrumb trail is 'Dashboard > All > New Item'. The main section is titled 'New Item' and contains a form to create a new item. The 'Enter an item name' field is filled with 'SkillsMatch'. Below this, the 'Select an item type' section shows four options: 'Freestyle project', 'Pipeline', 'Multi-configuration project', and 'Folder'. The 'Pipeline' option is highlighted with a red border. At the bottom of the form is an 'OK' button.

**New Item**

Enter an item name

SkillsMatch

Select an item type

**Freestyle project**  
Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.

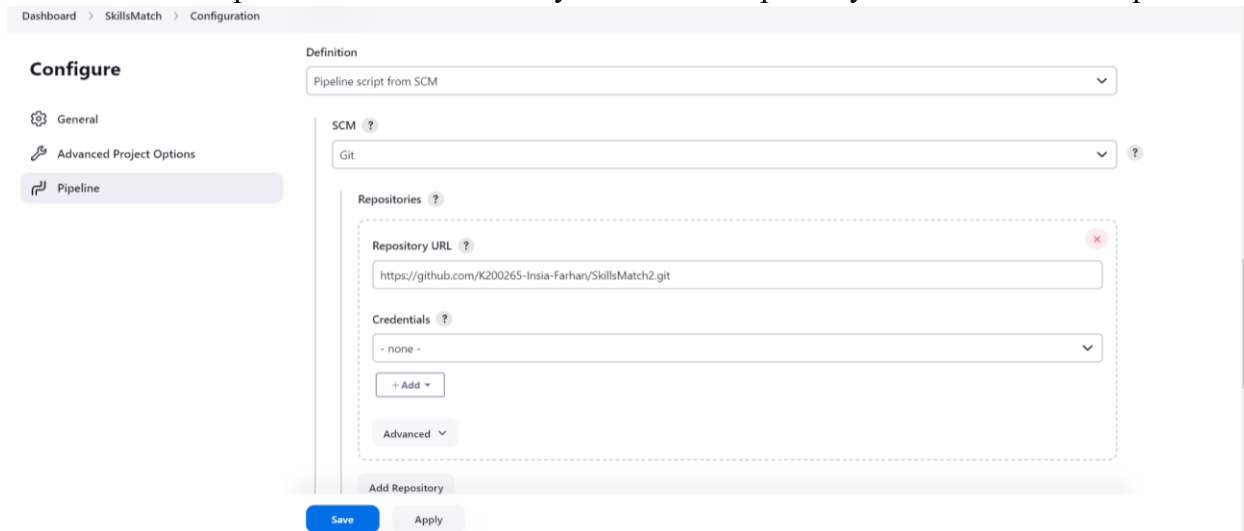
**Pipeline**  
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.

**Multi-configuration project**  
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.

**Folder**

OK

## 9. Choose Pipeline from SCM. Insert your GitHub Repository link and Jenkinsfile path.



The Jenkins 'Configure' page for the 'SkillsMatch' item is shown. The breadcrumb trail is 'Dashboard > SkillsMatch > Configuration'. The left sidebar shows the 'Configure' section with three tabs: 'General', 'Advanced Project Options', and 'Pipeline'. The 'Pipeline' tab is selected. The main section is titled 'Definition' and contains a form to configure the pipeline. The 'Definition' dropdown is set to 'Pipeline script from SCM'. The 'SCM' dropdown is set to 'Git'. The 'Repositories' section is expanded, showing a 'Repository URL' field with the value 'https://github.com/K200265-Insia-Farhan/SkillsMatch2.git'. The 'Credentials' dropdown is set to '- none -'. There is an '+ Add' button and an 'Advanced' dropdown. At the bottom of the form are 'Save' and 'Apply' buttons.

**Configure**

General  
Advanced Project Options  
Pipeline

**Definition**

Pipeline script from SCM

SCM ?

Git ?

Repositories ?

Repository URL ?

https://github.com/K200265-Insia-Farhan/SkillsMatch2.git

Credentials ?

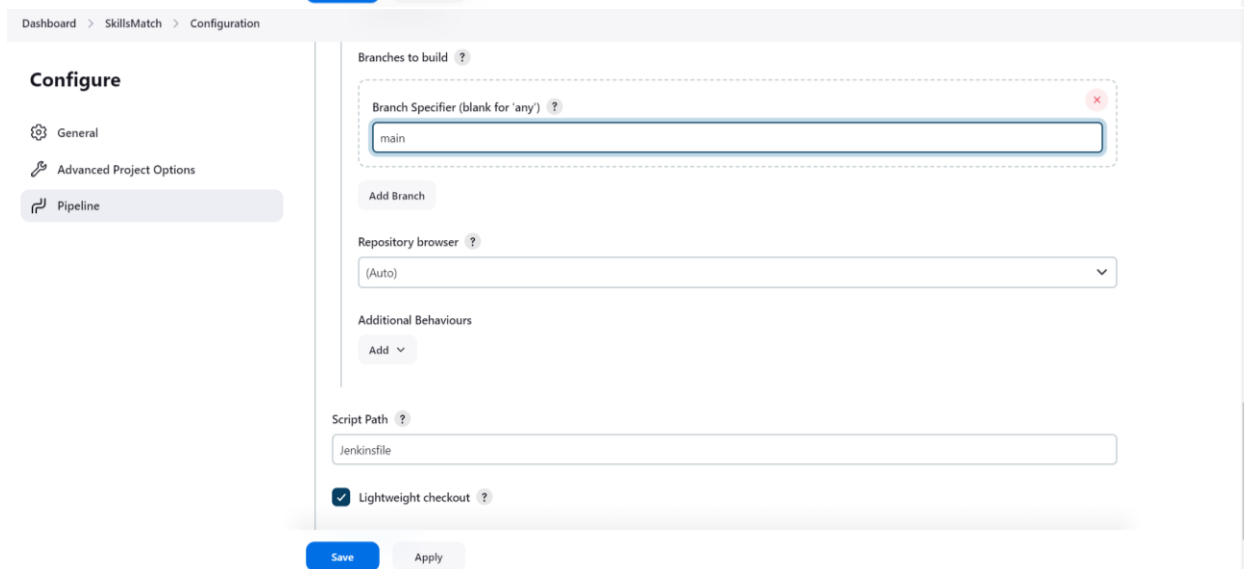
- none -

+ Add

Advanced

Add Repository

Save Apply



The Jenkins 'Configure' page for the 'SkillsMatch' item is shown. The breadcrumb trail is 'Dashboard > SkillsMatch > Configuration'. The left sidebar shows the 'Configure' section with three tabs: 'General', 'Advanced Project Options', and 'Pipeline'. The 'Pipeline' tab is selected. The main section is titled 'Definition' and contains a form to configure the pipeline. The 'Definition' dropdown is set to 'Pipeline script from SCM'. The 'SCM' dropdown is set to 'Git'. The 'Repositories' section is expanded, showing a 'Repository URL' field with the value 'https://github.com/K200265-Insia-Farhan/SkillsMatch2.git'. The 'Credentials' dropdown is set to '- none -'. There is an '+ Add' button and an 'Advanced' dropdown. At the bottom of the form are 'Save' and 'Apply' buttons.

**Configure**

General  
Advanced Project Options  
Pipeline

**Definition**

Pipeline script from SCM

SCM ?

Git ?

Repositories ?

Repository URL ?

https://github.com/K200265-Insia-Farhan/SkillsMatch2.git

Credentials ?

- none -

+ Add

Advanced

Add Repository

Save Apply

## 10. Run the following commands on the EC2 instance.

### Step 1: Ensure Jenkins User has Sudo Privileges for Docker Commands

You need to configure the sudoers file to allow the Jenkins user to execute Docker commands without a password. Here's how you can do it:

#### 1. Edit the Sudoers File:

Open the sudoers file using `visudo`:

```
sh
sudo visudo
```

#### 2. Add the Following Line:

```
sh
jenkins ALL=(ALL) NOPASSWD: /usr/bin/docker
```

This allows the "jenkins" user to run Docker commands without being prompted for a password.

#### 1. Test Sudo Configuration:

- Switch to the Jenkins user or a similar non-interactive session:

```
sh
sudo su - jenkins
```

- Test a Docker command with sudo to ensure it doesn't prompt for a password:

```
sh
sudo docker ps
```

### Step 4: Restart Services

Restart the necessary services to ensure the changes take effect:

```
sh
sudo systemctl restart jenkins
sudo systemctl restart docker
```

## 11. Configure Load Balancer for EC2 for Kubernetes.

**Load balancers (7)**

Elastic Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic.

Filter load balancers

	Name	DNS name	State	VPC ID	Availability Zones	Type
<input type="checkbox"/>	a1772ab7e0a1d4e579...	a1772ab7e0a1d4e5790bb...	-	vpc-07b13ab19dc07f0...	2 Availability Zones	classic
<input type="checkbox"/>	a80520fce59dc41fd81...	a80520fce59dc41fd8171b...	-	vpc-07b13ab19dc07f0...	2 Availability Zones	classic
<input type="checkbox"/>	a7a5a6f584b0a4b198...	a7a5a6f584b0a4b1986c1...	-	vpc-07b13ab19dc07f0...	2 Availability Zones	classic
<input type="checkbox"/>	a2e4bebbfa52946588...	a2e4bebbfa52946588f2d...	-	vpc-07b13ab19dc07f0...	2 Availability Zones	classic
<input type="checkbox"/>	ad06c58ab338f4a649...	ad06c58ab338f4a649aac...	-	vpc-07b13ab19dc07f0...	2 Availability Zones	classic

0 load balancers selected

```
[cloudshell-user@ip-10-140-110-81 ~]$ aws eks update-kubeconfig --name skillmatchfinal --region us-east-1
Added new context arn:aws:eks:us-east-1:211125493694:cluster/skillmatchfinal to /home/cloudshell-user/.kube/config
[cloudshell-user@ip-10-140-110-81 ~]$
[cloudshell-user@ip-10-140-110-81 ~]$ aws s3 mb s3://my-kubeconfig-bucket
make_bucket failed: s3://my-kubeconfig-bucket An error occurred (BucketAlreadyExists) when calling the CreateBucket operation: The requested bucket name is not available. The bucket namespace is shared by all users of the system. Please select a different name and try again.
[cloudshell-user@ip-10-140-110-81 ~]$ aws s3 mb s3://my-kubeconfig-bucket1
make_bucket: my-kubeconfig-bucket1
[cloudshell-user@ip-10-140-110-81 ~]$ aws s3 cp /home/cloudshell-user/.kube/config s3://my-kubeconfig-bucket1/kubeconfig
upload: .kube/config to s3://my-kubeconfig-bucket1/kubeconfig
[cloudshell-user@ip-10-140-110-81 ~]$
```

```
aws eks update-kubeconfig --name skillsmatchfinal --region us-east-1
aws s3 mb s3://my-kubeconfig-bucket1
aws s3 cp /home/cloudshell-user/.kube/config s3://my-kubeconfig-bucket1/kubeconfig
```

12. Get kubeconfig file, save it in S3 and then download it. (way to download anything from EC2)

Amazon S3

► **Account snapshot - updated every 24 hours** All AWS Regions [View Storage Lens dashboard](#)

Storage lens provides visibility into storage usage and activity trends. [Learn more](#)

---

**General purpose buckets** | Directory buckets

---

**General purpose buckets (1)** Info All AWS Regions [Refresh](#) [Copy ARN](#) [Empty](#) [Delete](#) [Create bucket](#)

Buckets are containers for data stored in S3.

	Name	AWS Region	IAM Access Analyzer	Creation date
<input type="radio"/>	<a href="#">my-kubeconfig-bucket1</a>	US East (N. Virginia) us-east-1	<a href="#">View analyzer for us-east-1</a>	May 29, 2024, 22:26:01 (UTC+05:00)

---

**Objects (1)** Info [Refresh](#) [Copy S3 URI](#) [Copy URL](#) [Download](#) [Open](#) [Delete](#) [Actions](#) [Create folder](#) [Upload](#)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

<input checked="" type="checkbox"/>	Name	Type	Last modified	Size	Storage class
<input checked="" type="checkbox"/>	<a href="#">kubeconfig</a>	-	May 29, 2024, 22:26:30 (UTC+05:00)	2.3 KB	Standard

13. Save necessary credentials on Jenkins.

Dashboard > Manage Jenkins > Credentials > System > Global credentials (unrestricted) > insiafarhan/\*\*\*\*\*

[Update](#) [Delete](#) [Move](#)

### Update credentials

Scope [?](#)  
Global (Jenkins, nodes, items, all child items, etc)

Username [?](#)  
insiafarhan

☐ Treat username as secret [?](#)

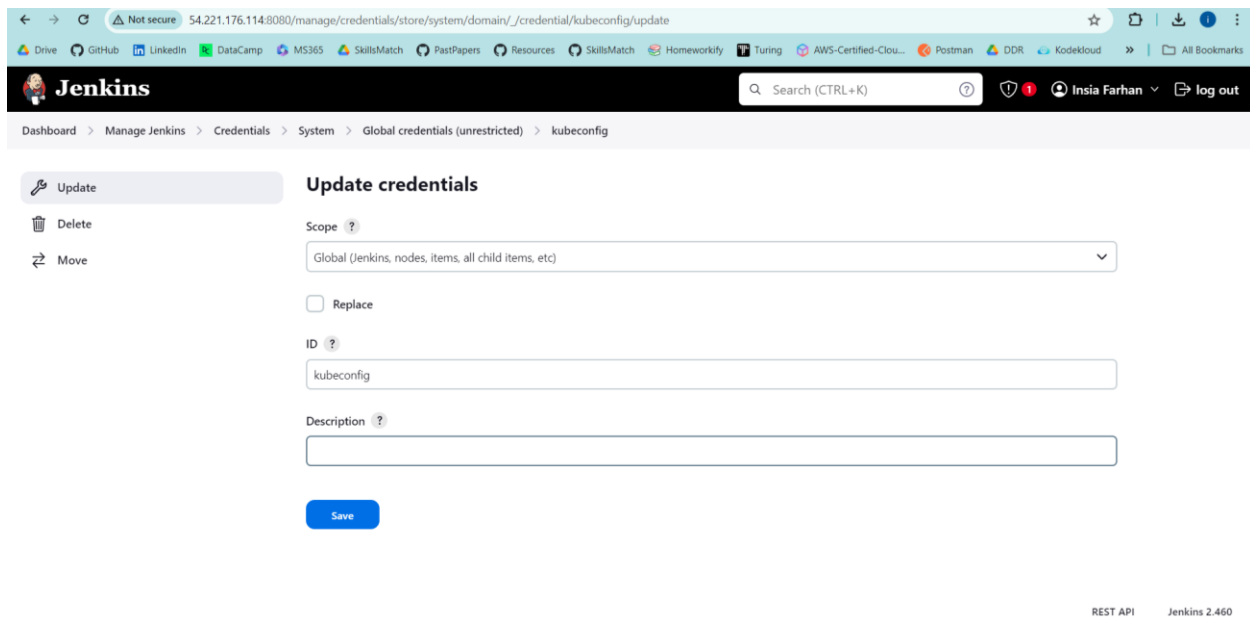
Password [?](#)  
Concealed [Change Password](#)

ID [?](#)  
dockerhub

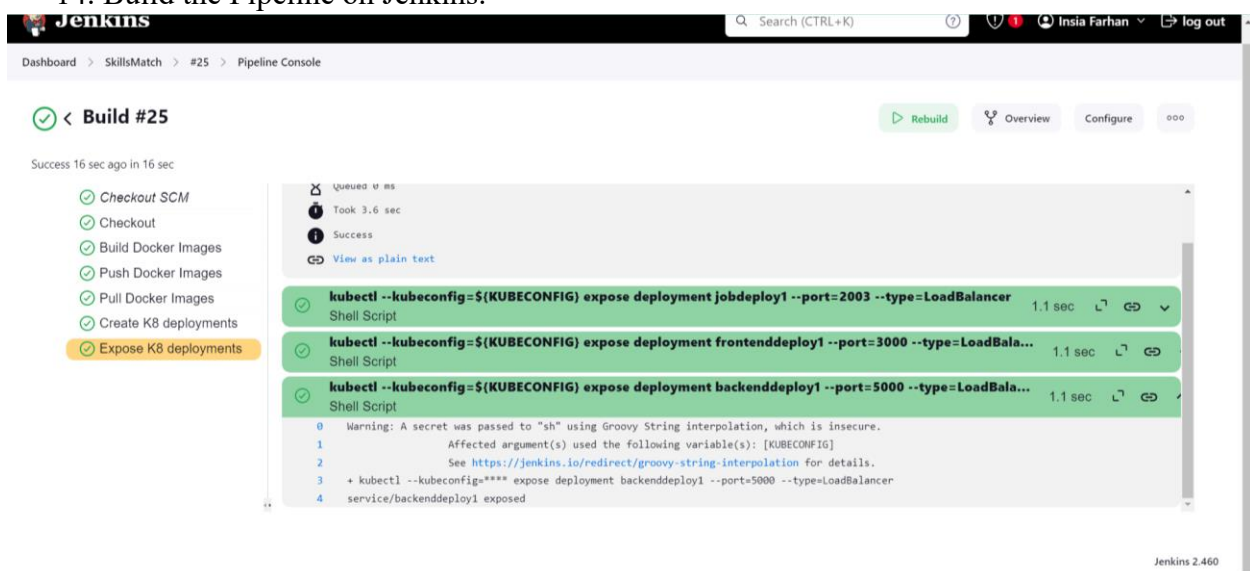
Description [?](#)

[Save](#)





## 14. Build the Pipeline on Jenkins.



### Side Notes:

Command to check logs: `nubectl logs -f "pod id"`

Winscp another solution for transfer of files from pc to ec2 instance, uses putty way and has drag drop feature.

### Reference link:

<https://medium.com/@belek.bagishbekov/how-to-install-and-configure-jenkins-on-amazon-linux-2023-a8d7463a0404>