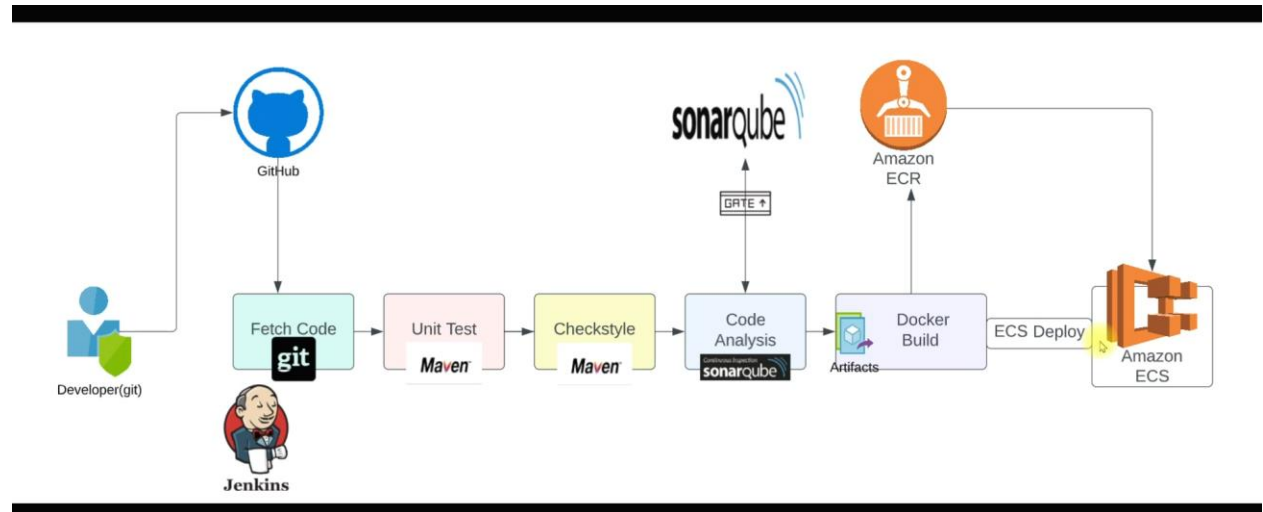


# Jenkins CI/CD with Docker

This project will demonstrate the steps to step guide on CI/CD with Jenkins, docker, Amazon ECR and ECS. Whenever developer make a code change and push it to github repository, Jenkins will poll github and start fetching the code, unit test with maven, do checkstyle analysis, sonar qube code analysis, and generate a docker build. The Image is then automatically push it to ECR registry and will create task in ECS. A load balancer was created for ECS to perform load balancing.



As a prerequisite, Host Jenkins, Nexus and Sonar qube. Refer the previous section Jenkins CI pipeline.

1. Install docker engine on jenkins

Refer the steps mentioned in <https://docs.docker.com/engine/install/ubuntu/>

```
root@ip-172-31-91-154: ~
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-backports InRelease
Get:4 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Get:5 https://download.docker.com/linux/ubuntu focal InRelease [57.7 kB]
Ign:6 https://pkg.jenkins.io/debian-stable binary/ InRelease
Hit:7 https://pkg.jenkins.io/debian-stable binary/ Release
Get:8 https://download.docker.com/linux/ubuntu focal/stable amd64 Packages [19.8 kB]
Fetched 191 kB in 1s (318 kB/s)
Reading package lists... Done
root@ip-172-31-91-154:~# sudo apt-get install docker-ce docker-ce-cli containerd.io docker-compose-plugin
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  docker-ce-rootless-extras docker-scan-plugin pigz slirp4netns
Suggested packages:
  aufs-tools cgroupfs-mount | cgroup-lite
The following NEW packages will be installed:
```

2. Add jenkins user to the jenkins group; And install awscli.

Reboot the instance.

```
root@ip-172-31-91-154: ~  
root@ip-172-31-91-154:~# id jenkins  
uid=113(jenkins) gid=119(jenkins) groups=119(jenkins)  
root@ip-172-31-91-154:~# usermod -a -G docker jenkins  
root@ip-172-31-91-154:~# id jenkins  
uid=113(jenkins) gid=119(jenkins) groups=119(jenkins),998(docker)  
root@ip-172-31-91-154:~# apt install awscli -y
```

3. Create IAM user with required access and download csv file

### Add user

1234

#### Set user details

You can add multiple users at once with the same access type and permissions. [Learn more](#)

User name\*

[Add another user](#)

#### Select AWS access type

Select how these users will primarily access AWS. If you choose only programmatic access, it does NOT prevent users from accessing the console using an assumed role. Access keys and autogenerated passwords are provided in the last step. [Learn more](#)

Select AWS credential type\*

☒ **Access key - Programmatic access**  
Enables an **access key ID** and **secret access key** for the AWS API, CLI, SDK, and other development tools.

☐ **Password - AWS Management Console access**  
Enables a **password** that allows users to sign-in to the AWS Management Console.

### Add user

12345

Set permissions

Add user to group

Copy permissions from existing user

Attach existing policies directly

Create policy

Filter policies

Showing 10 results

	Policy name	Type	Used as
<input checked="" type="checkbox"/>	AmazonEC2ContainerRegistryFullAccess	AWS managed	None
<input type="checkbox"/>	AmazonEC2ContainerRegistryReadOnly	AWS managed	None

Creating state

Creating state

Create policy

Filter policies

Q ecs

Showing 5 results

	Policy name	Type	Used as
<input checked="" type="checkbox"/>	AmazonECS_FullAccess	AWS managed	None
<input type="checkbox"/>	AmazonECSTaskExecutionRolePolicy	AWS managed	Permissions policy (1)
<input type="checkbox"/>	AWSCodeDeployRoleForECS	AWS managed	None

#### 4. Create a ECR – registry.

Services

Search for services, features, blogs, docs, and more

[Alt+S]

Amazon ECR

>

Repositories

>

Create repository

Create repository

General settings

Visibility settings

Info

Choose the visibility setting for the repository.

☒ Private
 

Access is managed by IAM and repository policy permissions.

☐ Public
 

Publicly visible and accessible for image pulls.

Repository name

Provide a concise name. A developer should be able to identify the repository contents by the name.

224777171125.dkr.ecr.us-east-1.amazonaws.com/

satzprofileimag

15 out of 256 characters maximum (2 minimum). The name must start with a letter and can only contain lowercase letters, numbers, hyphens, underscores, periods and forward slashes.

Tag immutability

Info

Enable tag immutability to prevent image tags from being overwritten by subsequent image pushes using the same tag. Disable tag immutability to allow image tags to be overwritten.

☐ Disabled

Once a repository is created, the visibility setting of the repository can't be changed.

5. Go to manage plugins and install all the below ones,

### Plugin Manager

Updates Available Installed Advanced

Q cloudbees doc

Install	Name	
<input checked="" type="checkbox"/>	Docker Pipeline	521.v1a_a_dd2073b_2e pipeline DevOps Deployment docker Build and use Docker containers from pipelines.
<input checked="" type="checkbox"/>	Amazon ECR	1.73.v741d474abe74 aws This plugin generates Docker authentication token from Amazon Credentials to access Amazon ECR.
<input checked="" type="checkbox"/>	Amazon Web Services SDK :: All	1.12.287-357.vf02d85a_6eefd Library plugins (for use by other plugins) aws This plugin provides all AWS SDK for Java modules not packaged as standalone plugins.
<input checked="" type="checkbox"/>	CloudBees Docker Build and Publish	1.4.0 Build Tools Jenkins

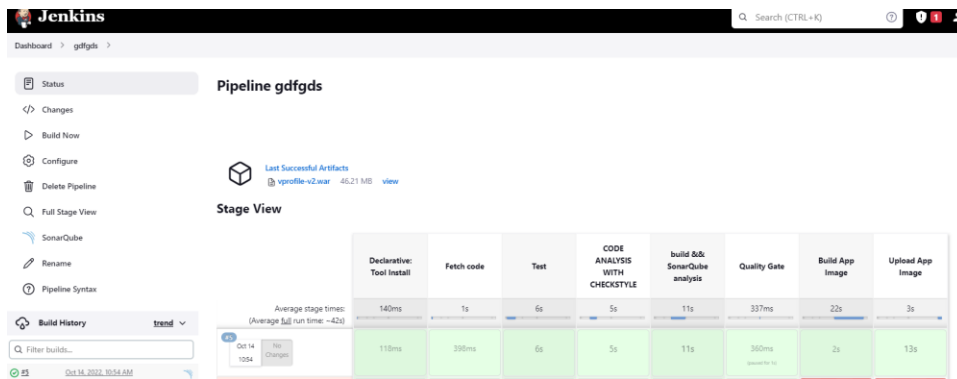
6. Creta AWS credentials in jenkins, based on the creds in the csv file

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
awscreds	AKIAIVODOC2NSWIOS (aws)	AWS Credentials	aws

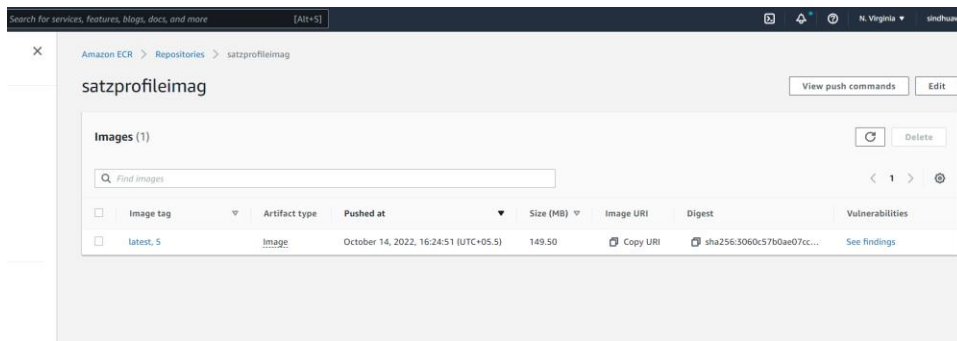
7. Update the endpoint and region in the pipeline file, and run the pipeline code  
PAAC\_CI\_Docker\_ECR+

```
pipeline {
  agent any
  tools {
    maven "MAVEN3"
    jdk "OracleJDK8"
  }
  environment {
    registryCredential = 'ecr:us-east-1:awscreds'
    appRegistry = "224777171125.dkr.ecr.us-east-1.amazonaws.com/satzprofileimag"
    vprofileRegistry = "https://224777171125.dkr.ecr.us-east-1.amazonaws.com"
  }
  stages {
    stage('Fetch code'){
```

8. Jenkins show success build,



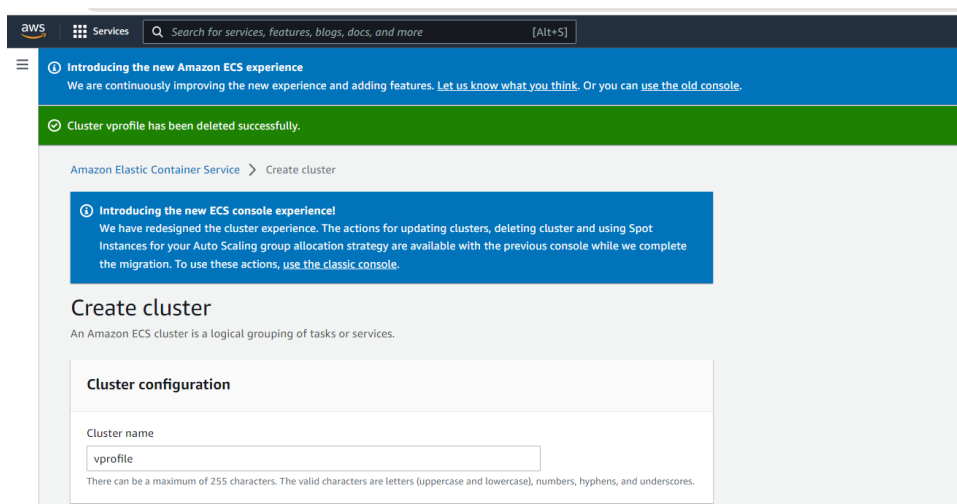
## 9. Ecr Show new images were created

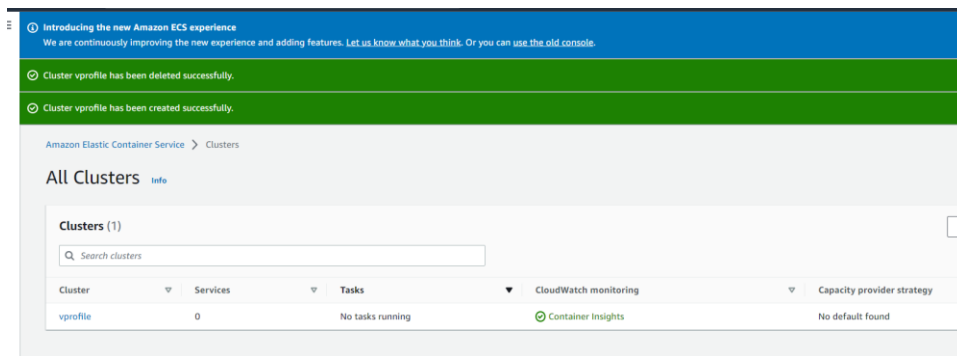


## 10. Let's get the image from ECR and host it into ECS,

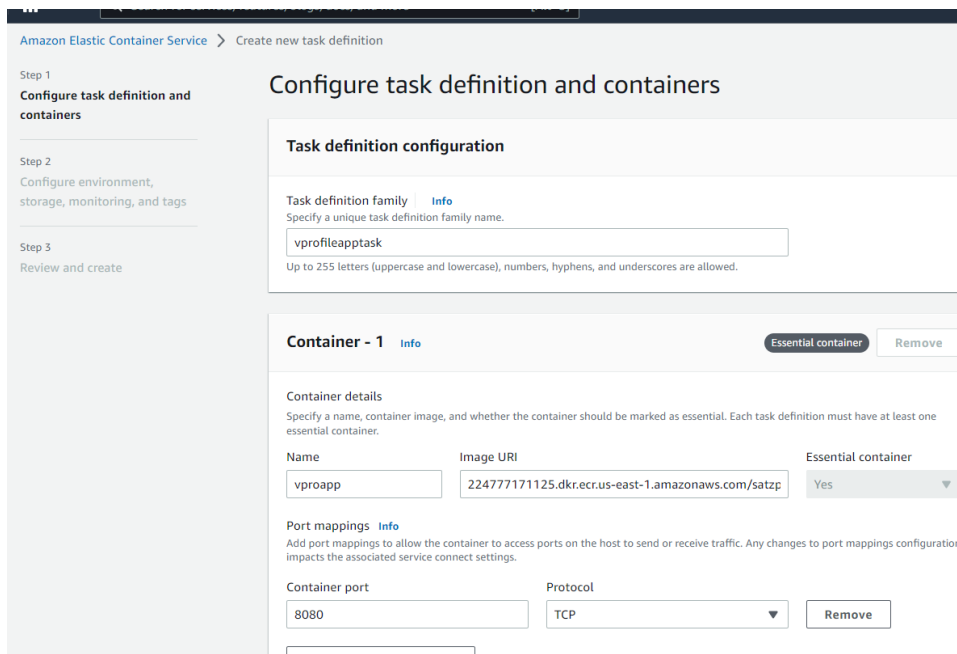
Refer the code PAAC\_CICD\_Docker\_ECR\_ECS\_satz

Create a new cluster in ECS,





## 11. Create new task definition with port 8080,



## 12. Open your cluster, select the app and do deploy,

vprofile

Cluster overview

ARN  
vprofile

Status  
Active

CloudWatch monitoring  
Container Insights

Services  
Draining  
-

Active  
-

Tasks  
Pending  
-

Services

Tasks

Infrastructure

Metrics

Tags

Services (0)

Filter services by value

All launch types

All service types

	Service name	Status	ARN	Service type	Deployments and tasks	Learn more
<div>No services</div> <div>No services to display.</div> <div>Deploy</div>						

#### Application type [Info](#)

Specify what type of application you want to run.

☒ **Service**

Launch a group of tasks handling a long-running computing work that can be stopped and restarted. For example, a web application.

☐ **Task**

Launch a standalone task that runs and terminates. For example, a batch job.

#### Task definition

Select an existing task definition. To create a new task definition, go to [Task definitions](#).

☐ **Specify the revision manually**

Manually input the revision instead of choosing from the 100 most recent revisions for the selected task definition family.

#### Family

vprofileapptask

#### Revision

1 (LATEST)

#### Service name

Assign a unique name for this service.

vprofileappsvc

#### Service type [Info](#)

Specify the service type that the service scheduler will follow.

☒ **Replica**

Place and maintain a desired number of tasks across your cluster.

☐ **Daemon**

Place and maintain one copy of your task on each container instance.

#### Desired tasks

Specify the number of tasks to launch.

1

## 12. Create Load balancer,

### ▼ Load balancing - optional



#### Load balancer type [Info](#)

Configure a load balancer to distribute incoming traffic across the tasks running in your service.

Application Load Balancer

#### Application Load Balancer

Specify whether to create a new load balancer or choose an existing one.

☒ **Create a new load balancer**

☐ **Use an existing load balancer**

#### Load balancer name

Assign a unique name for the load balancer.

vprofileappelbecs

#### Choose container to load balance

vproapp 8080:8080

## 13. Update target group



Specify whether to create a new target group or choose an existing one that the load balancer will use to route requests to the tasks in your service.

- ☒ Create new target group
- ☐ Use an existing target group  
You need to select an existing load balancer.

Target group name

vproecstg

Protocol

HTTP

Health check path [Info](#)

/login

Health check protocol

HTTP

Health check grace period [Info](#)

seconds

## 14. Under networking, add a new rule for elb and deploy

### ▼ Networking

VPC [Info](#)

Choose the Virtual Private Cloud to use.

vpc-0f7768310869e62ba  
DefaultVPC | default

Subnets

Choose the subnets within the VPC that the task scheduler should consider for placement.

Choose subnets

subnet-0fc5b2d6e499e7574  
us-east-1e

subnet-05aab9ecb3fe84451  
us-east-1d

subnet-01ec494f0faf24e67  
us-east-1f

subnet-06aaecb2f18f6e86d  
us-east-1c

subnet-00e634a80c475b40c  
us-east-1b

subnet-0bd8735237bc58d26  
us-east-1a

Security group [Info](#)

Choose an existing security group or create a new security group.

- ☐ Use an existing security group
- ☒ Create a new security group

Security group details

Specify the configuration to use when creating the new security group.

Security group name

vproappeslb-sg

Security group description

The security group name can have up to 255 characters. Valid characters: A-Z, a-z, 0-9, spaces, and the . \_ - / ! \$ % & ' ( ) \* + , ; : . , @ [ ] ^ ` { } | ~ .

The security group description can have up to 255 characters. Valid characters: A-Z, a-z, 0-9, spaces, and the . \_ - / ! \$ % & ' ( ) \* + , ; : . , @ [ ] ^ ` { } | ~ .

Inbound rules for security groups

Add one or more ingress rules for your security group.

Type	Protocol	Port range	Source	Values
HTTP	TCP	80	An...	0.0.0.0/0

## 15. Go to load balancer,

Create Load Balancer

Actions

Filter by tags and attributes or search by keyword

Name	DNS name	State	VPC ID	Availability Zones
vprofileecslb	vprofileecslb-1728326094.us...	Active	vpc-0f7768310869e62ba	us-east-1b, us-east-1f,

Load balancer: vprofileecslb

Description

Listeners

Monitoring

Integrated services

Tags

Listeners listen for connection requests using their protocol and port. You can add, remove, or update listeners and listener rules.

To view and edit listener attributes, select the listener and choose Edit.

Add listener

Edit

Delete

Listener ID	Security policy	SSL Certificate	Rules
<input type="checkbox"/> <a href="#">HTTP : 80</a> am...e9b9b803614695fa	N/A	N/A	Default: forwarding to vproecstg <a href="#">View/edit rules</a>

## 16. Click on the target groups

EC2 > Target groups > vproecstg

vproecstg

Details

arnawselasticloadbalancing-us-east-1:224777171125:targetgroup/vproecstg/0fa721332284d62

Target type	Protocol : Port	Protocol version	VPC
IP	HTTP: 80	HTTP1	vpc-0f77683108
IP address type	Load balancer		
IPv4	vprofileecslb		

Total targets	Healthy	Unhealthy	Unused	Initial
0	0	0	0	0

Targets

Monitoring

Health checks

Attributes

Tags

Health check settings

Protocol	Path	Port	Healthy threshold
HTTP	/login	Traffic port	5 consecutive he
Unhealthy threshold	Timeout	Interval	Success codes
2 consecutive health check failures	5 seconds	30 seconds	200

Update the highlighted on target groups,

EC2 > Target groups > vproecstg > Edit health check settings

## Edit health check settings

### Health checks

The associated load balancer periodically sends requests, per the settings below, to the registered targets to test their status.

#### Health check protocol

HTTP

#### Health check path

Use the default path of "/" to ping the root, or specify a custom path if preferred.

/login

Up to 1024 characters allowed.

#### Advanced health check settings

Restore defaults

#### Port

The port the load balancer uses when performing health checks on targets. The default is the port on which each target receives traffic from the load balancer, but you can specify a different port.

☐ Traffic port

☒ Override

8080

1-65535

#### Healthy threshold

The number of consecutive health checks successes required before considering an unhealthy target healthy.

2

2-10

#### Unhealthy threshold

The number of consecutive health check failures required before considering a target unhealthy.

17. Service is created successfully,

The screenshot shows the Amazon Elastic Container Service (ECS) console. At the top, a green banner states "vprofileapp has been deployed successfully." Below this, the breadcrumb navigation is "Amazon Elastic Container Service > Clusters > vprofile > Services". The main heading is "vprofile". On the right, there are buttons for "Edit cluster" and "Delete cluster".

The "Cluster overview" section displays the following information:

- ARN: vprofile
- Status: Active
- CloudWatch monitoring: Container Insights
- Registered container instances: -
- Services: Active
- Draining: 1
- Tasks: Pending
- Running: 1

Below the overview, there are tabs for "Services", "Tasks", "Infrastructure", "Metrics", and "Tags". The "Services" tab is selected, showing a list of services. The list has a search bar and filters for "All launch types" and "All service types".

Service name	Status	ARN	Service type	Deployments and tasks	Last deployment	Task definition	Revision	Launch type
vprofileapp	Active	arn:aws:ecs:us-east-1:123456789012:cluster/vprofile/service/vprofileapp	REPLICA	1/1 Tasks x...	In progress	vprofileapp-task	1	-

18. Update the ECS lb security group to allow traffic from 8080,

EC2 > Security Groups > sg-0d5438ffe10977340 - ec2-ec2-sg > Edit inbound rules

**Edit inbound rules** [info](#)

Inbound rules control the incoming traffic that's allowed to reach the instance.

Security group rule ID	Type	Protocol	Port range	Source	Description - optional
sg-0d5438ffe10977340	HTTP	TCP	80	Custom	
sg-0d5438ffe10977340	HTTP	TCP	80	Custom	
-	Custom TCP	TCP	8080	Anywhere IPv4	
-	Custom TCP	TCP	8080	Anywhere IPv4	

[Add rule](#)

[Cancel](#) [Preview changes](#) [Save rules](#)

19. Now come to ES service and click on Load balancing,

Amazon Elastic Container Service > Clusters > vprofile > Services > vprofile > Networking

**vprofile** [Edit service](#) [Delete service](#)

Health and metrics | Logs | Configuration and tasks | Deployments and events | **Networking** | Tags

**Network configuration**


<b>Network</b> vpc-077768310860e52ba Subnets subnet-065b236e4999e7574 subnet-00a0b0e0336d4451 subnet-010c4980f234a607 subnet-06a0eb2718f6e86d subnet-00e6134a80c475b40c subnet-0ba0f752370c5b20e	<b>Security groups</b> sg-0d5438ffe10977340 Auto-assign public IP ENABLED Health check grace period -	<b>Service role</b> AWSServiceRoleforECS	<b>Load balancers</b> vprofilealb DNS names vprofilealb-1728326094.us-east-1.elb.amazonaws.com ip-172-31-133-100.us-east-1.elb.amazonaws.com Target groups vprofiletg
--	--	---	---

20. Our app will be opened,

← → Not secure | vprofilealb-1728326094.us-east-1.elb.amazonaws.com/login

**VISUAL PATH** TECHNOLOGIES ABOUT BLOG

**LOGIN**



Username

Password

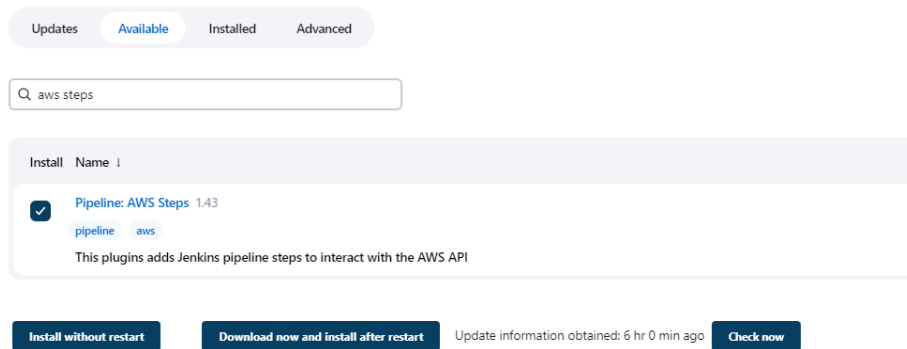
**LOGIN**

[Create an account](#)

21. Now will do all these from Pipeline, update cluster and service details in the pipeline code, refer PAAC\_CICD\_Docker\_ECR\_ECS\_satz in github

```
1 pipeline {
2   agent any
3   tools {
4     maven "MAVEN3"
5     jdk "OracleJDK8"
6   }
7
8   environment {
9     registryCredential = 'ecr:us-east-1:awscreds'
10    appRegistry = "224777171125.dkr.ecr.us-east-1.amazonaws.com/satzprofileimag"
11    vprofileRegistry = "https://224777171125.dkr.ecr.us-east-1.amazonaws.com"
12    cluster = "vprofile"
13    service = "vprofileap"
14  }
15  stages {
16    stage('Fetch code'){
```

22. Install pipeline aws steps plugin in jenkins



23. Create new pipeline and upload the pipeline script from PAAC\_CICD\_Docker\_ECR\_ECS\_satz in github.

24. Before build, Verify whether task and service are running fine. And note the container id. Bcz build is going to create a new container,

Amazon Elastic Container Service > Clusters > vprofile > Tasks > 566168e1082d4c07811befe9f5c66d3a > Configuration > Containers > vproapp

### 566168e1082d4c07811befe9f5c66d3a

arn:aws:ecs:us-east-1:224777171125:task/vprofile/566168e1082d4c07811befe9f5c66d3a

**Configuration** | Logs | Networking | Tags

---

**Status**

Last status 🟢 Running	Desired status 🟢 Running	Started at 10/14/2022, 11:43:41 UTC	Created at 10/14/2022, 11:43:09 UTC
--------------------------	-----------------------------	--	--

---

**Configuration**

Operating system/Architecture Linux/x86_64	Capacity provider FARGATE	ENI ID eni-0e455663a49ea0783 <a href="#">🔗</a>	Public IP 18.232.73.180   <a href="#">open address</a>
CPU   Memory 1 vCPU   2 GB	Launch type FARGATE	Network mode awspc	Private IP 172.31.79.9
Platform version 1.4.0	Task definition vprofileasptask:1	Subnet ID subnet-01ec4940fa24e67 <a href="#">🔗</a>	MAC address 16:3d:aa:1c:9e:01

---

**Containers (1)**

Container name	Container runtime ID	Image URI	Image Digest	Status	Health status
vproapp	566168e1082d4c07811befe9f5c66d3a-1606146...	224777171125.dkr.ecr.u...	sha256:3060c57b0ae07c...	🟢 Running	🟡 Unknown

25. We ran the build it is failed, because we gave us-east2 for cluster, actually it is us-east-1, reran the code then

**Pipeline cicd-pipeline-ecs** [Add description](#) [Disable Pipeline](#)

**Stage View**

	Declarative: Tool Install	Fetch code	Test	CODE ANALYSIS WITH CHECKSTYLE	build & SonarQube analysis	Quality Gate	Build App Image	Upload App Image	Deploy to ecs
Average stage times:	132ms	2s	8s	5s	11s	330ms	1s	2s	1s
Oct 14 11:56 <span>No Changes</span>	132ms	2s	8s	5s	11s	330ms <small>(skipped for 10)</small>	1s	2s	1s <span>Failed</span>

**SonarQube Quality Gate**

26. Build ran successfully,

**Pipeline cicd-pipeline-ecs** [Add description](#) [Disable Pipeline](#)

**Stage View**

	Declarative: Tool Install	Fetch code	Test	CODE ANALYSIS WITH CHECKSTYLE	build & SonarQube analysis	Quality Gate	Build App Image	Upload App Image	Deploy to ecs
Average stage times: (Average full run time: ~31s)	127ms	1s	7s	5s	11s	290ms	1s	2s	1s
Oct 14 11:58 <span>No Changes</span>	123ms	500ms	6s	5s	11s	250ms <small>(skipped for 10)</small>	664ms	2s	1s
Oct 14 11:56 <span>No Changes</span>	132ms	2s	8s	5s	11s	330ms <small>(skipped for 10)</small>	1s	2s	1s <span>Failed</span>

27. We can see the job ran 1 minute ago,

Amazon Elastic Container Service > Clusters > vprofile > Services > vprofile > Deployments

### vprofile

Health and metrics | Logs | Configuration and tasks | **Deployments and events** | Networking | Tags

[View pipeline](#)

**Deployment configuration**

Cluster vprofile	Platform version LATEST	Deployment type ECS	Min running tasks 100 % Max running tasks 200 %
---------------------	----------------------------	------------------------	--

► Advanced

**Deployments (2)**

Filter deployments

Start date	Status	Failed tasks	Tasks	Version	Task definition	Revision	Last deployment	Deployment ID
4 minutes ago	Primary <span>100%</span>	0	1 Running   0 Pending   1 Desired	1.4.0	vprofileaptask	1	In progress	ecs-svc/9634455106210279677
25 minutes ago	Active <span>100%</span>	0	1 Running   0 Pending   1 Desired	1.4.0	vprofileaptask	1	Completed	ecs-svc/7523394659542785696

**Events (27)**

In few mins, there will be only one deployment will be running, other will be deleted.

## 28. Clean up

First in the service update the desired task to 0.

Amazon Elastic Container Service > Clusters > vprofile > Services > vprofile > Edit

### Edit vprofile

**Deployment configuration**

Task definition  
Select an existing task definition. To create a new task definition, go to [Task definitions](#).

☐ Specify the revision manually  
Manually input the revision instead of choosing from the 100 most recent revisions for the selected task definition family.

Family: vprofileaptask  
Revision: 1 (LATEST)

Service type  
REPLICA

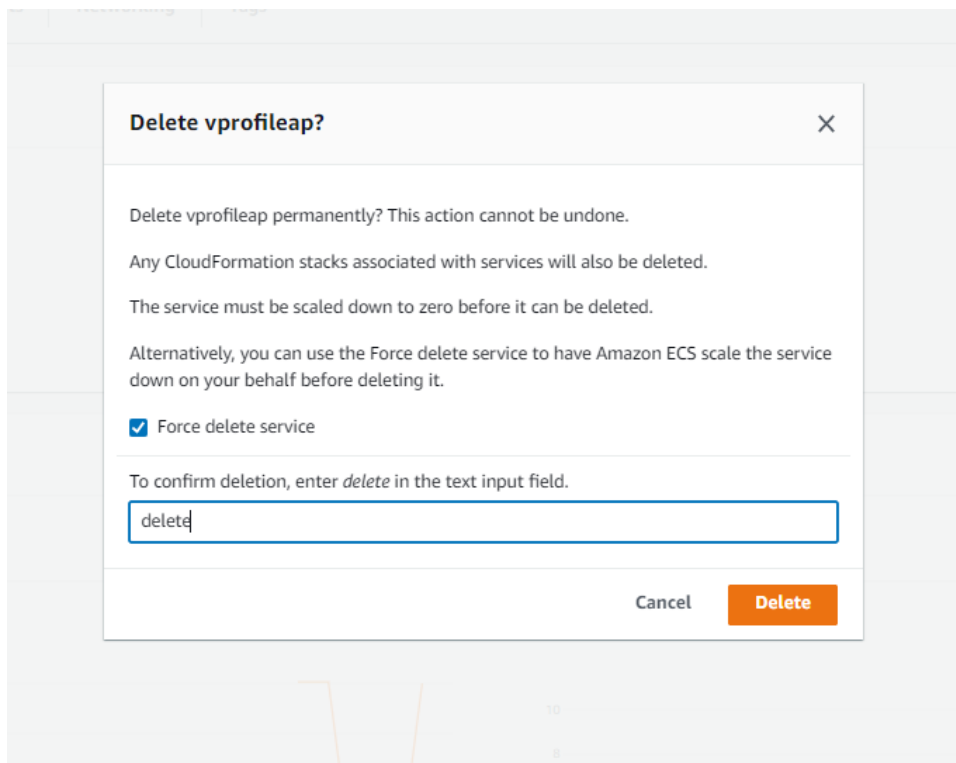
Desired tasks  
Specify the number of tasks to launch.  
0

► Deployment options

► Deployment failure detection [Info](#)

► **Service auto scaling - optional**  
Automatically adjust your service's desired count up and down within a specified range in response to CloudWatch alarms. You can modify your service auto scaling configuration at any time to meet the needs of your application.

Then delete the service



Stop the running tasks,

Task	Last status	Desired status	Task definition	Revision	Health status	Started at	Container instan...	Launch type	CPU	Memory
<input checked="" type="checkbox"/> 156354e480...	Running	Running	vprofileapp:task	1	Unknown	7 minutes ago	-	FARGATE	1 vCPU	2 GB
<input type="checkbox"/> 1012a0314f52...	Stopped	Stopped	vprofileapp:task	1	Unknown	28 minutes ago	-	FARGATE	1 vCPU	2 GB
<input checked="" type="checkbox"/> 56618b1082...	Running	Stopped	vprofileapp:task	1	Unknown	23 minutes ago	-	FARGATE	1 vCPU	2 GB
<input type="checkbox"/> 75b7312799f...	Stopped	Stopped	vprofileapp:task	1	Unknown	26 minutes ago	-	FARGATE	1 vCPU	2 GB
<input type="checkbox"/> 7909226797...	Stopped	Stopped	vprofileapp:task	1	Unknown	25 minutes ago	-	FARGATE	1 vCPU	2 GB
<input checked="" type="checkbox"/> 60745007917...	Stopped	Stopped	vprofileapp:task	1	Unknown	31 minutes ago	-	FARGATE	1 vCPU	2 GB

Now delete the cluster

