

MAIN PROJECT DEVOPS:

CREATING A PIPELINE :

The image displays a development environment (VS Code) and a Jenkins CI/CD pipeline overview.

VS Code Editor:

- File Explorer:** Shows the project structure with files like `package.json`, `Dockerfile`, and `.dockerignore`.
- Dockerfile:** Contains the following instructions:

```
1 You, 10 minutes ago | 1 author (You)
2 # Use official Node.js runtime
3 FROM node:18-slim
4
5 # Set working directory
6 WORKDIR /usr/src/app
7
8 # Copy only dependency files first
9 COPY package.json package-lock.json* ./
10
11 # Install dependencies
12 RUN npm install --production
13
14 # Copy app source
15 COPY . .
16
17 # Expose port
18 EXPOSE 3000
19
20 # Run app
21 CMD ["npm", "start"]
```
- Terminal:** Shows the output of a `git push` command, indicating a successful push to the main branch.

Jenkins Pipeline Overview:

- Pipeline Name:** App-Pipeline / #4 / Pipeline Overview
- Status:** #4 (Green checkmark)
- Actions:** Manually run by Rajesh, Started 29 sec ago, Queued 1 ms, Took 14 sec.
- Graph:** A horizontal flow diagram showing the pipeline stages: Start, Checkout SCM, Checkout, Build Docker image, Prepare ECR & Login, Tag & Push to ECR, Deploy to EKS, Post Actions, End. All stages are marked with green checkmarks.
- Search:** A search bar is present.
- Post Actions:** A list of actions with their durations and status:
 - Checkout SCM: 1.6s
 - Checkout: 1.3s
 - Build Docker image: 2.3s
 - Prepare ECR & Login: 2.2s
 - Tag & Push to ECR: 1.2s
 - Deploy to EKS: 3.4s
 - Post Actions: 0.11s
- Post Actions Details:** A detailed view of the Post Actions stage shows:
 - Delete workspace when build is done: 50ms
 - Pipeline finished successfully: 26ms
 - Pipeline finished successfully.

Instances | EC2 | ap-south-1

Elastic Container Registry - Private

CloudFormation - Stack

#4 - App-Pipeline - Jenkins

Install and Set Up kubectl on Linux

+

ap-south-1.console.aws.amazon.com/cloudformation/home?region=ap-south-1#/stacks?filteringText=&filteringStatus=active&viewNested=true

aws

Search

[Alt+S]

Asia Pacific (Mumbai)

Account ID: 2686-9716-8542

Rajesh S

CloudFormation

Stacks

StackSets

Exports

Infrastructure Composer

laC generator

Hooks overview

Invocation summary

Hooks

Registry

Public extensions

Activated extensions

Publisher

Spotlight

Feedback

Introducing a new Control Catalog integration for AWS CloudFormation Hooks

You can now create Hooks more easily by using proactive controls from the AWS Control Tower Control Catalog and view your historical invocations. Learn more about [Hook invocations](#) and the [Control Catalog](#).

View invocation summary

Stacks (2)

Filter status

Active

View nested

1

Stack name	Status	Created time	Description
eksctl-my-cluster-nodegroup-my-nodes	CREATE_COMPLETE	2025-09-10 18:15:32 UTC+05:30	EKS Managed Nodes (SSH access: false) [created by eksctl]
eksctl-my-cluster-cluster	CREATE_COMPLETE	2025-09-10 18:05:28 UTC+05:30	EKS cluster (dedicated VPC: true, dedicated IAM: true) [created and managed by eksctl]

CloudShell

Feedback

© 2025, Amazon Web Services, Inc. or its affiliates.

Privacy

Terms

Cookie preferences

Instances | EC2 | ap-south-1

Elastic Container Registry - Private

CloudFormation - Stack

#4 - App-Pipeline - Jenkins

Install and Set Up kubectl on Linux

+

ap-south-1.console.aws.amazon.com/ecr/repositories/private/268697168542/myapp?region=ap-south-1

aws

Search

[Alt+S]

Asia Pacific (Mumbai)

Account ID: 2686-9716-8542

Rajesh S

Amazon ECR

Private registry

Repositories

myapp

Amazon Elastic Container Registry

Private registry

Repositories

Summary

Images

Permissions

Lifecycle Policy

Repository tags

Features & Settings

Public registry

Repositories

Settings

ECR public gallery

Amazon ECS

Amazon EKS

Getting started

Documentation

Image scan overview, status, and full vulnerabilities has moved to the Image detail page. To access, click an image tag.

View push commands

Images (2)

Search artifacts

1

Image tag	Artifact type	Pushed at	Size (MB)	Image URI	Digest	Last recorded pull time
latest	Image	September 10, 2025, 18:59:07 (UTC+05.5)	71.46	Copy URI	sha256:4b2d44af2e6978f...	September 10, 2025, 18:59:11 (UTC+05.5)
-	Image	September 10, 2025, 18:48:02 (UTC+05.5)	71.46	Copy URI	sha256:395d5a87787469...	-

CloudShell

Feedback

© 2025, Amazon Web Services, Inc. or its affiliates.

Privacy

Terms

Cookie preferences

File Edit Selection View Go Run Terminal Help

Devops Project

EXPLORER

DEVOPS PROJECT

package.json Jenkinsfile k8s-deployment.yaml app.js Dockerfile .dockerignore

Jenkinsfile

1 pipeline {
2 agent any
3
4 environment {
5 AWS_REGION = "ap-south-1" // <- change if different
6 AWS_ACCOUNT = "268697168542" // <- REPLACE with your AWS Account ID
7 ECR_REPO = "myapp" // <- your ECR repo name
8 IMAGE_TAG = "latest"
9 GIT_CRED_ID = "github-token" // your GitHub credential ID in Jenkins
10 AWS_CRED_ID = "project-user-cred" // your AWS credential ID in Jenkins
11 }
12
13 stages {
14 stage('Checkout') {
15 steps {
16 echo "Checking out source..."
17 git branch: 'main', credentialsId: env.GIT_CRED_ID, url: 'https://github.com/Rajesh0SR/Devops-Projec
18 }
19 }
20
21 stage('Build Docker image') {
22 steps {
23 echo "Building Docker image..."
24 sh "docker build -t \${ECR_REPO}:\${IMAGE_TAG} ."
25 }
26 }
27
28 stage('Prepare ECR & Login') {
29 steps {
30 echo "Ensure ECR repo exists and login..."
31 }
32 }
33 }
34 }

main Launchpad Amazon Q You, 23 hours ago User1 (22 hours ago) Ln 10, Col 39 (17 selected) Spaces: 2 UTF-8 CRLF Groovy Prettier

Rajesh0SR / Devops-Project

Code Issues Pull requests Actions Projects Security Insights Settings

Devops-Project Private

Watch 0 Fork 0 Star 0

main 1 Branch 0 Tags

Go to file Add file Code

User1 updated 43699ea · yesterday 31 Commits

.dockerignore	file changed	2 days ago
Dockerfile	file changed	2 days ago
Jenkinsfile	updated	yesterday
app.js	file changed	2 days ago
k8s-deployment.yaml	deploy file added	yesterday
package.json	file changed	2 days ago

README

Add a README

About

No description, website, or topics provided.

Activity

0 stars

0 watching

0 forks

Releases

No releases published

Create a new release

Packages

No packages published

Publish your first package

Languages

Dockerfile 54.6% JavaScript 45.4%

```
13.234.110.110 (ubuntu)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultExec Tunneling Packages Settings Help
Quick connect...
Reconnect SSH-browser
13.232.93.54 (ubuntu)
13.234.110.110 (ubuntu)
Client Version: v1.34.1
Kustomize Version: v5.7.1
root@ip-10-0-1-14:~# aws configure
AWS Access Key ID [*****MBGH]:
AWS Secret Access Key [*****hNeZ]:
Default region name [ap-south-1]:
Default output format [json]:
root@ip-10-0-1-14:~# kubectl get nodes
NAME                                STATUS    ROLES    AGE    VERSION
ip-192-168-18-32.ap-south-1.compute.internal Ready    <none>   51m    v1.32.8-eks-99d6cc0
ip-192-168-71-203.ap-south-1.compute.internal Ready    <none>   51m    v1.32.8-eks-99d6cc0
root@ip-10-0-1-14:~# aws eks --region ap-south-1 update-kubeconfig --name my-cluster
Updated context arn:aws:eks:ap-south-1:268697168542:cluster/my-cluster to /root/.kube/config
root@ip-10-0-1-14:~# kubectl get nodes
NAME                                STATUS    ROLES    AGE    VERSION
ip-192-168-18-32.ap-south-1.compute.internal Ready    <none>   51m    v1.32.8-eks-99d6cc0
ip-192-168-71-203.ap-south-1.compute.internal Ready    <none>   51m    v1.32.8-eks-99d6cc0
root@ip-10-0-1-14:~# kubectl get deployments
NAME    READY    UP-TO-DATE    AVAILABLE    AGE
myapp-deployment 2/2      2              2            19m
root@ip-10-0-1-14:~# kubectl get svc myapp-service
NAME    TYPE    CLUSTER-IP    EXTERNAL-IP    PORT(S)    AGE
myapp-service LoadBalancer 10.100.96.104  af399e6f813744c08a1ffdb914311c24-422889997.ap-south-1.elb.amazonaws.com 80:31050/TCP 10m
root@ip-10-0-1-14:~# kubectl describe svc myapp-service
Name:      myapp-service
Namespace: default
Labels:     <none>
Annotations: <none>
Selector:   app=myapp
Type:       LoadBalancer
IP Family Policy: SingleStack
IP Families: IPv4
IP:         10.100.96.104
IPs:        10.100.96.104
LoadBalancer Ingress: af399e6f813744c08a1ffdb914311c24-422889997.ap-south-1.elb.amazonaws.com
Port:      <unset> 80/TCP
TargetPort: 3000/TCP
NodePort:  <unset> 31050/TCP
Endpoints: 192.168.83.203:3000,192.168.31.143:3000
Session Affinity: None
External Traffic Policy: Cluster
Internal Traffic Policy: Cluster
Events:

```