EKS CodePipeline Deployment

Objective

To automate the continuous integration and continuous deployment (CI/CD) of containerized microservices into an Amazon Elastic Kubernetes Service (EKS) cluster using AWS CodePipeline, CodeBuild, and Amazon Elastic Container Registry (ECR). This ensures seamless updates with minimal downtime and reliable deployment of new application versions

Resource used

AWS Service/Resource	Purpose
Amazon EKS	Managed Kubernetes service to run containerized apps
AWS CodePipeline	Orchestrates the CI/CD workflow
AWS CodeBuild	Builds Docker images and runs tests
Amazon ECR	Container image repository
AWS IAM	Manages roles and permissions for the pipeline
AWS CloudFormation (optional)	Infrastructure as Code to provision resources
Amazon S3	Source code storage (if using S3 as a source)
Kubernetes (kubectl)	Command-line tool to interact with EKS cluster
Docker	Containerizes the application

Step-by-Step Deployment Process

Step:1 Connect to the linux ec2 instance

Install AWS CLI, kubectl, eksctl

```
sudo yum install -y unzip curl

curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"

unzip awscliv2.zip

sudo ./aws/install

# Verify

aws –version
```

Install kubectl

```
curl -LO "https://s3.us-west-2.amazonaws.com/amazon-eks/1.29.0/2024-05-31/bin/linux/amd64/kubectl"

chmod +x kubectl

sudo mv kubectl /usr/local/bin/

# Verify

kubectl version –client
```

Install eksctl

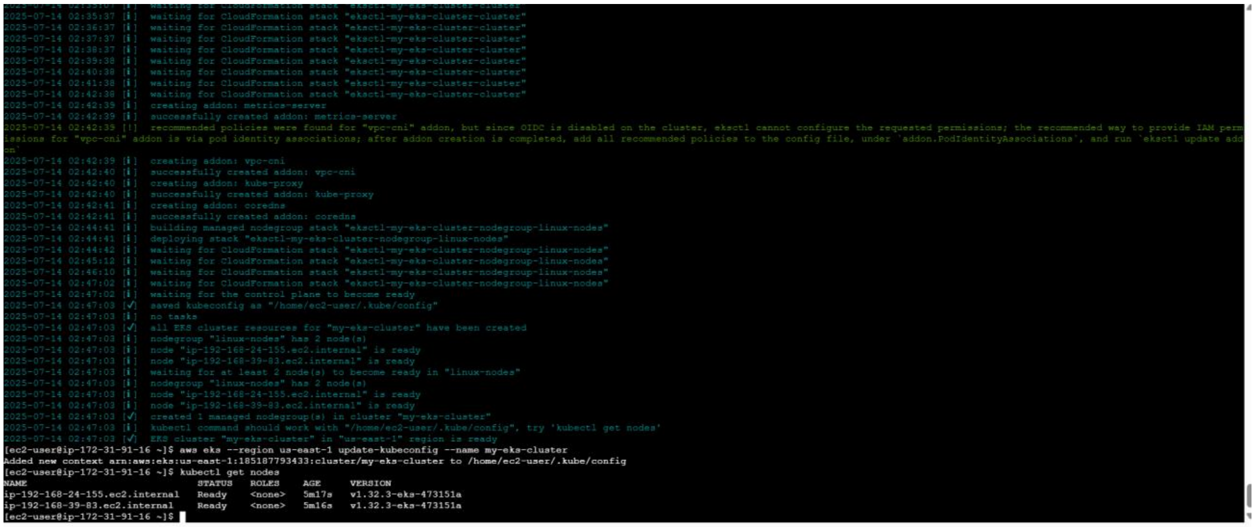
```
curl --silent --location "https://github.com/eksctl-io/eksctl/releases/latest/download/eksctl_Linux_amd64.tar.gz" | tar xz -C /tmp sudo mv /tmp/eksctl /usr/local/bin

eksctl version
```

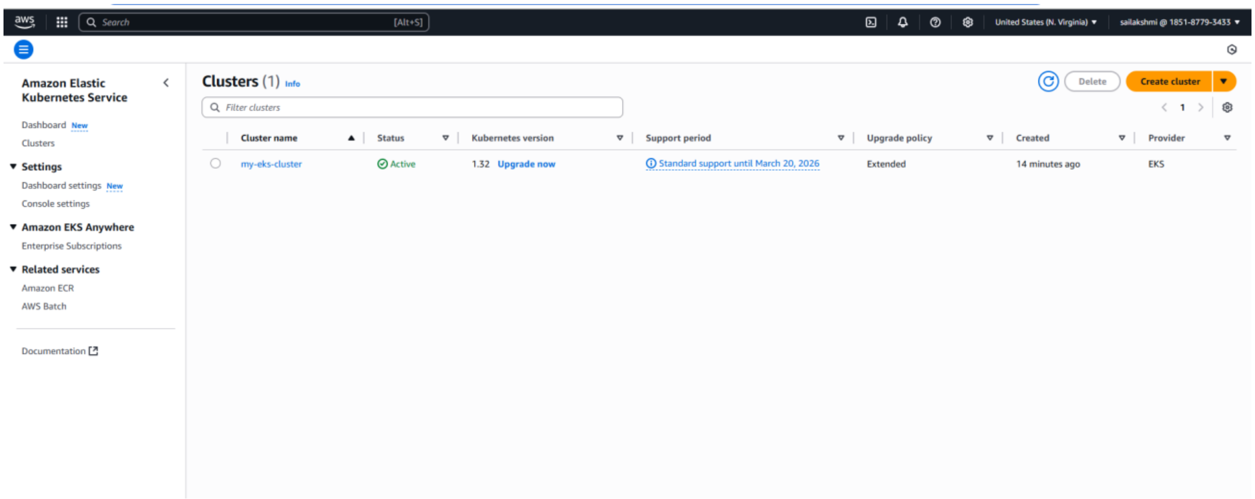
Step:2 Create EKS cluster

- Create an EKS cluster via AWS Console, CLI, or eksctl.
- Configure worker nodes to join the cluster.
- Configure kubectl with cluster credentials.

curl --silent --location "https://github.com/weaveworks/eksctl/releases/latest/download/eksctl_$(uname -s)_amd64.tar.gz" | tar xz -C /usr/local/bin eksctl create cluster --name my-eks-cluster --region us-east-1 --nodegroup-name linux-nodes --node-type t3.medium --nodes 2 --nodes-min 2 --nodes-max 3

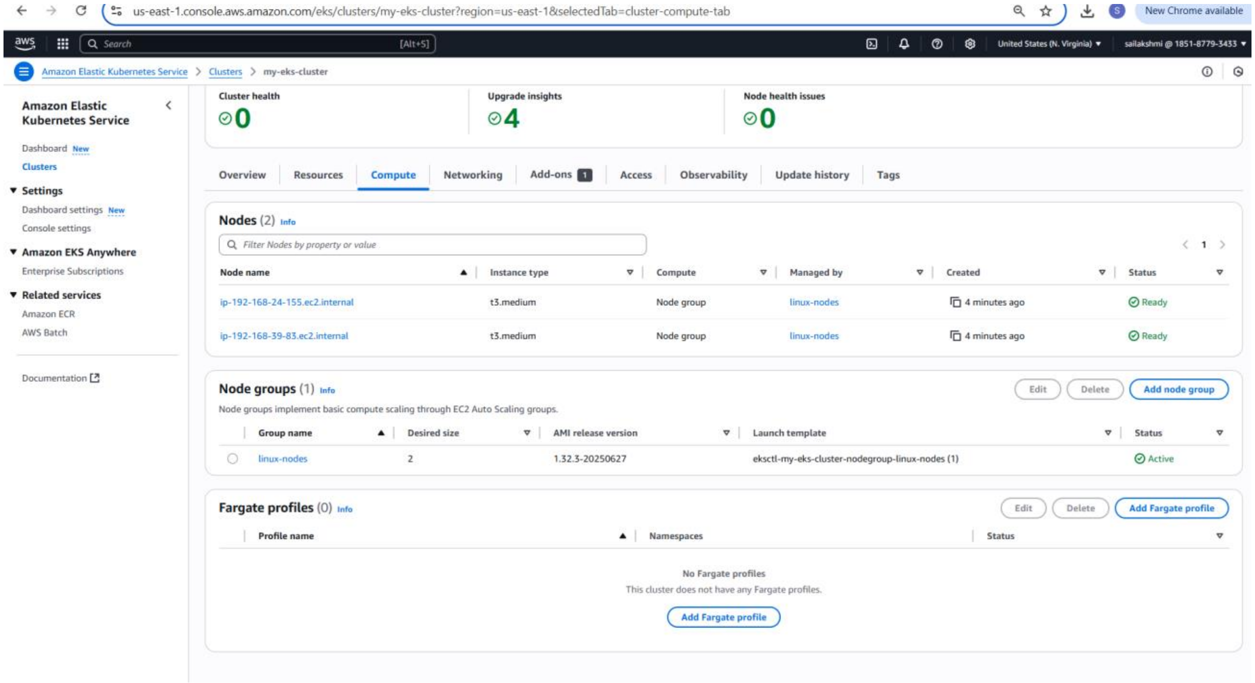


```
2025-07-14 02:35:07 [ℹ]  waiting for CloudFormation stack "eksctl-my-eks-cluster-cluster"
2025-07-14 02:35:37 [ℹ]  waiting for CloudFormation stack "eksctl-my-eks-cluster-cluster"
2025-07-14 02:36:37 [ℹ]  waiting for CloudFormation stack "eksctl-my-eks-cluster-cluster"
2025-07-14 02:37:37 [ℹ]  waiting for CloudFormation stack "eksctl-my-eks-cluster-cluster"
2025-07-14 02:38:37 [ℹ]  waiting for CloudFormation stack "eksctl-my-eks-cluster-cluster"
2025-07-14 02:39:38 [ℹ]  waiting for CloudFormation stack "eksctl-my-eks-cluster-cluster"
2025-07-14 02:40:38 [ℹ]  waiting for CloudFormation stack "eksctl-my-eks-cluster-cluster"
2025-07-14 02:41:38 [ℹ]  waiting for CloudFormation stack "eksctl-my-eks-cluster-cluster"
2025-07-14 02:42:38 [ℹ]  waiting for CloudFormation stack "eksctl-my-eks-cluster-cluster"
2025-07-14 02:42:39 [ℹ]  creating addon: metrics-server
2025-07-14 02:42:39 [ℹ]  successfully created addon: metrics-server
2025-07-14 02:42:39 [!]  recommended policies were found for "vpc-cni" addon, but since OIDC is disabled on the cluster, eksctl cannot configure the requested permissions; the recommended way to provide IAM perm
issions for "vpc-cni" addon is via pod identity associations; after addon creation is completed, add all recommended policies to the config file, under `addon.PodIdentityAssociations`, and run `eksctl update add
on`
2025-07-14 02:42:39 [ℹ]  creating addon: vpc-cni
2025-07-14 02:42:40 [ℹ]  successfully created addon: vpc-cni
2025-07-14 02:42:40 [ℹ]  creating addon: kube-proxy
2025-07-14 02:42:40 [ℹ]  successfully created addon: kube-proxy
2025-07-14 02:42:41 [ℹ]  creating addon: coredns
2025-07-14 02:42:41 [ℹ]  successfully created addon: coredns
2025-07-14 02:44:41 [ℹ]  building managed nodegroup stack "eksctl-my-eks-cluster-nodegroup-linux-nodes"
2025-07-14 02:44:41 [ℹ]  deploying stack "eksctl-my-eks-cluster-nodegroup-linux-nodes"
2025-07-14 02:44:42 [ℹ]  waiting for CloudFormation stack "eksctl-my-eks-cluster-nodegroup-linux-nodes"
2025-07-14 02:45:12 [ℹ]  waiting for CloudFormation stack "eksctl-my-eks-cluster-nodegroup-linux-nodes"
2025-07-14 02:46:10 [ℹ]  waiting for CloudFormation stack "eksctl-my-eks-cluster-nodegroup-linux-nodes"
2025-07-14 02:47:02 [ℹ]  waiting for CloudFormation stack "eksctl-my-eks-cluster-nodegroup-linux-nodes"
2025-07-14 02:47:02 [ℹ]  waiting for the control plane to become ready
2025-07-14 02:47:03 [✓]  saved kubeconfig as "/home/ec2-user/.kube/config"
2025-07-14 02:47:03 [ℹ]  no tasks
2025-07-14 02:47:03 [✓]  all EKS cluster resources for "my-eks-cluster" have been created
2025-07-14 02:47:03 [ℹ]  nodegroup "linux-nodes" has 2 node(s)
2025-07-14 02:47:03 [ℹ]  node "ip-192-168-24-155.ec2.internal" is ready
2025-07-14 02:47:03 [ℹ]  node "ip-192-168-39-83.ec2.internal" is ready
2025-07-14 02:47:03 [ℹ]  waiting for at least 2 node(s) to become ready in "linux-nodes"
2025-07-14 02:47:03 [ℹ]  nodegroup "linux-nodes" has 2 node(s)
2025-07-14 02:47:03 [ℹ]  node "ip-192-168-24-155.ec2.internal" is ready
2025-07-14 02:47:03 [ℹ]  node "ip-192-168-39-83.ec2.internal" is ready
2025-07-14 02:47:03 [✓]  created 1 managed nodegroup(s) in cluster "my-eks-cluster"
2025-07-14 02:47:03 [ℹ]  kubectl command should work with "/home/ec2-user/.kube/config", try 'kubectl get nodes'
2025-07-14 02:47:03 [✓]  EKS cluster "my-eks-cluster" in "us-east-1" region is ready
[ec2-user@ip-172-31-91-16 ~]$ aws eks --region us-east-1 update-kubeconfig --name my-eks-cluster
Added new context arn:aws:eks:us-east-1:185187793433:cluster/my-eks-cluster to /home/ec2-user/.kube/config
[ec2-user@ip-172-31-91-16 ~]$ kubectl get nodes
NAME                             STATUS   ROLES    AGE     VERSION
ip-192-168-24-155.ec2.internal   Ready    <none>   5m17s   v1.32.3-eks-473151a
ip-192-168-39-83.ec2.internal    Ready    <none>   5m16s   v1.32.3-eks-473151a
[ec2-user@ip-172-31-91-16 ~]$
```



Clusters (1) Info

	Cluster name	Status	Kubernetes version	Support period	Upgrade policy	Created	Provider
○	my-eks-cluster	Active	1.32 Upgrade now	Standard support until March 20, 2026	Extended	14 minutes ago	EKS



Cluster health 0

Upgrade insights 4

Node health issues 0

Nodes (2) Info

Node name	Instance type	Compute	Managed by	Created	Status
ip-192-168-24-155.ec2.internal	t3.medium	Node group	linux-nodes	4 minutes ago	Ready
ip-192-168-39-83.ec2.internal	t3.medium	Node group	linux-nodes	4 minutes ago	Ready

Node groups (1) Info

Node groups implement basic compute scaling through EC2 Auto Scaling groups.

	Group name	Desired size	AMI release version	Launch template	Status
○	linux-nodes	2	1.32.3-20250627	eksctl-my-eks-cluster-nodegroup-linux-nodes (1)	Active

Fargate profiles (0) Info

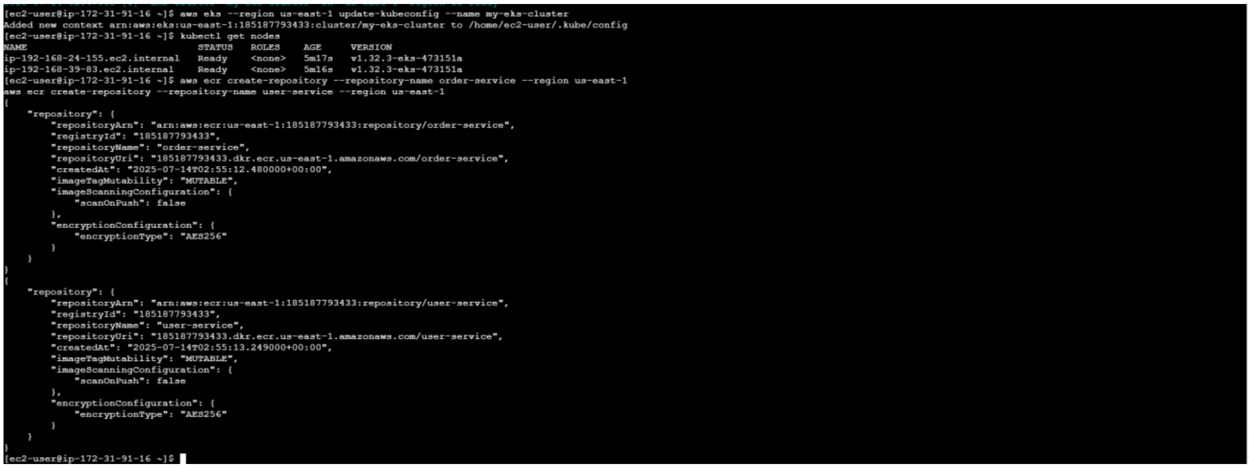
Profile name	Namespaces	Status

No Fargate profiles

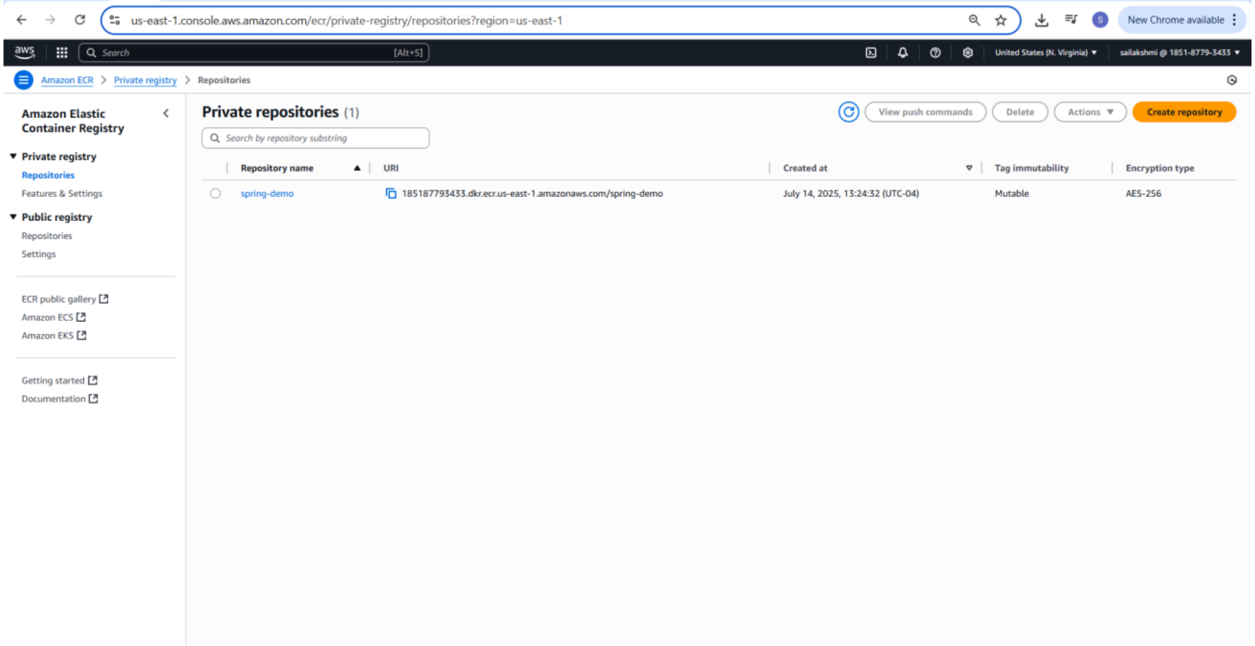
This cluster does not have any Fargate profiles.

Step 3: Amazon ECR Repository

Create an ECR repository to store Docker images.



```
[ec2-user@ip-172-31-91-16 ~]$ aws eks --region us-east-1 update-kubeconfig --name my-eks-cluster
Added new context arn:aws:eks:us-east-1:185187793433:cluster/my-eks-cluster to /home/ec2-user/.kube/config
[ec2-user@ip-172-31-91-16 ~]$ kubectl get nodes
NAME                            STATUS   ROLES    AGE     VERSION
ip-192-168-24-155.ec2.internal   Ready    <none>   5m17s   v1.32.3-eks-473151a
ip-192-168-39-83.ec2.internal    Ready    <none>   5m16s   v1.32.3-eks-473151a
[ec2-user@ip-172-31-91-16 ~]$ aws ecr create-repository --repository-name order-service --region us-east-1
aws ecr create-repository --repository-name user-service --region us-east-1
{
    "repository": {
        "repositoryArn": "arn:aws:ecr:us-east-1:185187793433:repository/order-service",
        "registryId": "185187793433",
        "repositoryName": "order-service",
        "repositoryUri": "185187793433.dkr.ecr.us-east-1.amazonaws.com/order-service",
        "createdAt": "2025-07-14T02:55:12.480000+00:00",
        "imageTagMutability": "MUTABLE",
        "imageScanningConfiguration": {
            "scanOnPush": false
        },
        "encryptionConfiguration": {
            "encryptionType": "AES256"
        }
    }
}
{
    "repository": {
        "repositoryArn": "arn:aws:ecr:us-east-1:185187793433:repository/user-service",
        "registryId": "185187793433",
        "repositoryName": "user-service",
        "repositoryUri": "185187793433.dkr.ecr.us-east-1.amazonaws.com/user-service",
        "createdAt": "2025-07-14T02:55:13.249000+00:00",
        "imageTagMutability": "MUTABLE",
        "imageScanningConfiguration": {
            "scanOnPush": false
        },
        "encryptionConfiguration": {
            "encryptionType": "AES256"
        }
    }
}
[ec2-user@ip-172-31-91-16 ~]$ 
```

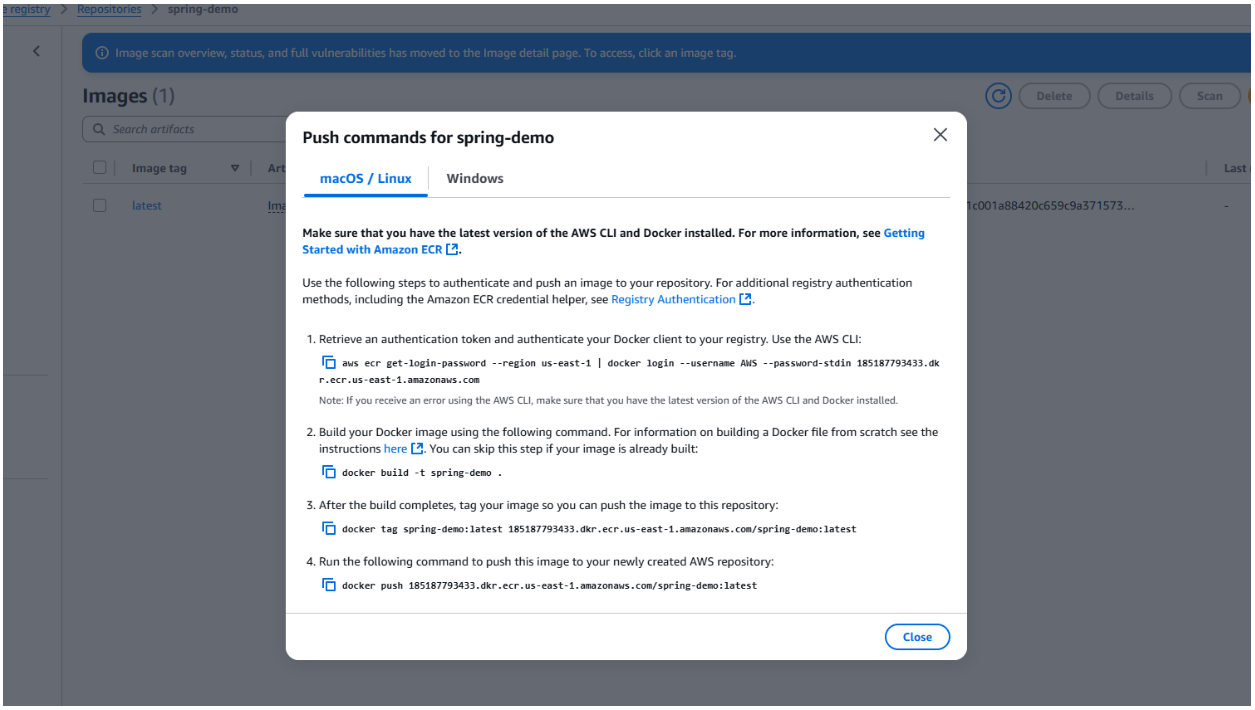


us-east-1.console.aws.amazon.com/ecr/private-registry/repositories?region=us-east-1

Private repositories (1)

	Repository name ▲	URI	Created at ▼	Tag immutability	Encryption type
○	spring-demo	185187793433.dkr.ecr.us-east-1.amazonaws.com/spring-demo	July 14, 2025, 13:24:32 (UTC-04)	Mutable	AES-256

Push docker image



registry > Repositories > spring-demo
Image scan overview, status, and full vulnerabilities has moved to the Image detail page. To access, click an image tag.
Images (1)
Delete
Details
Scan
Search artifacts
Image tag
latest

Push commands for spring-demo

macOS / Linux | Windows

Make sure that you have the latest version of the AWS CLI and Docker installed. For more information, see Getting Started with Amazon ECR.

Use the following steps to authenticate and push an image to your repository. For additional registry authentication methods, including the Amazon ECR credential helper, see Registry Authentication.

1. Retrieve an authentication token and authenticate your Docker client to your registry. Use the AWS CLI:

```
aws ecr get-login-password --region us-east-1 | docker login --username AWS --password-stdin 185187793433.dkr.ecr.us-east-1.amazonaws.com
```

   Note: If you receive an error using the AWS CLI, make sure that you have the latest version of the AWS CLI and Docker installed.

2. Build your Docker image using the following command. For information on building a Docker file from scratch see the instructions here. You can skip this step if your image is already built:

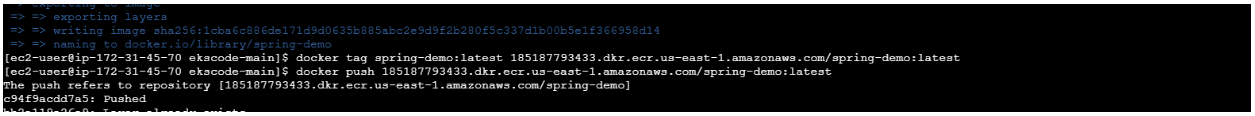
```
docker build -t spring-demo .
```

3. After the build completes, tag your image so you can push the image to this repository:

```
docker tag spring-demo:latest 185187793433.dkr.ecr.us-east-1.amazonaws.com/spring-demo:latest
```

4. Run the following command to push this image to your newly created AWS repository:

```
docker push 185187793433.dkr.ecr.us-east-1.amazonaws.com/spring-demo:latest
```



```
 => => exporting layers
 => => writing image sha256:1cba6c886de171d9d0635b885abc2e9d9f2b280f5c337d1b00b5e1f366958d14
 => => naming to docker.io/library/spring-demo
[ec2-user@ip-172-31-45-70 ekscode-main]$ docker tag spring-demo:latest 185187793433.dkr.ecr.us-east-1.amazonaws.com/spring-demo:latest
[ec2-user@ip-172-31-45-70 ekscode-main]$ docker push 185187793433.dkr.ecr.us-east-1.amazonaws.com/spring-demo:latest
The push refers to repository [185187793433.dkr.ecr.us-east-1.amazonaws.com/spring-demo]
c94f9acdd7a5: Pushed
```

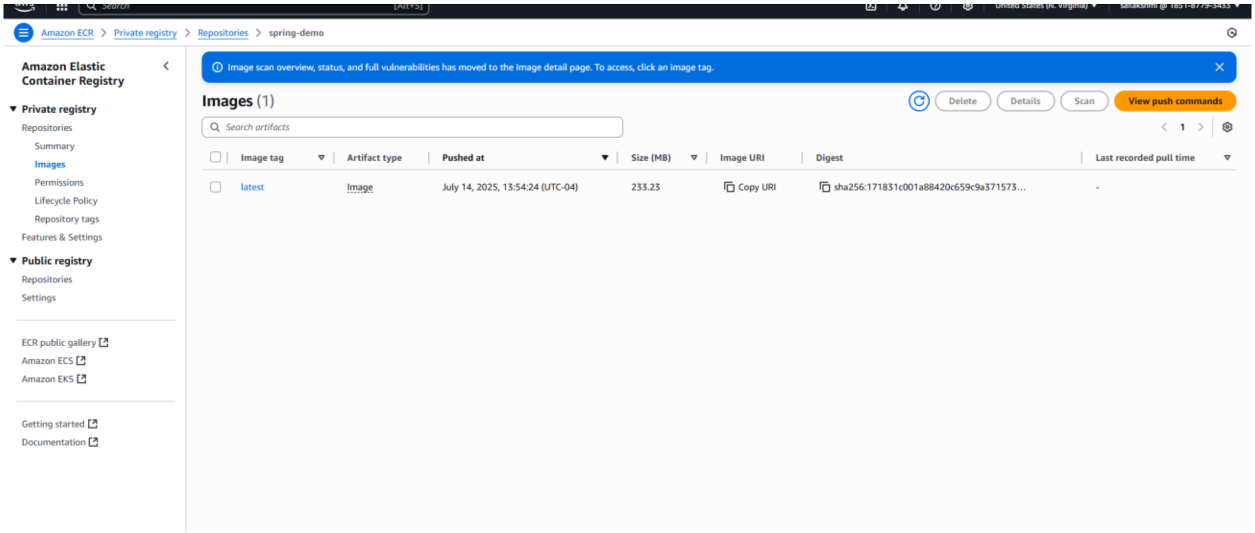


Image scan overview, status, and full vulnerabilities has moved to the Image detail page. To access, click an image tag.

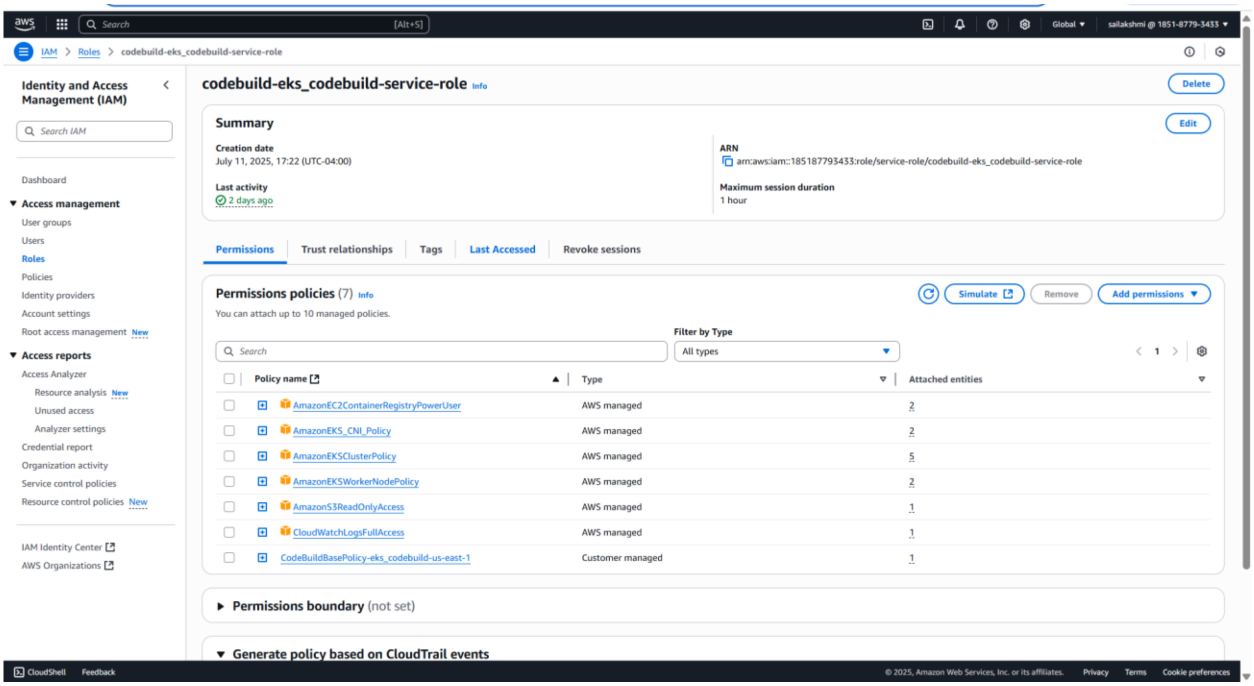
Images (1)

□	Image tag	Artifact type	Pushed at	Size (MB)	Image URI	Digest	Last recorded pull time
□	latest	Image	July 14, 2025, 13:54:24 (UTC-04)	233.23	Copy URI	sha256:171831c001a88420c659c9a371573...	-

Step 4 : Create IAM roles and policies for CodeBuild & CodePipeline

- **CodeBuild Role:** Needs permissions for ECR (push), EKS (update kubeconfig, kubectl), S3 (artifact storage), CloudWatch Logs
- AWS has managed policies you can attach:

- AmazonEC2ContainerRegistryPowerUser
- AmazonEKSClusterPolicy
- AmazonEKSWorkerNodePolicy
- AmazonS3FullAccess (or least privilege)
- CloudWatchLogsFullAccess



IAM > Roles > codebuild-eks_codebuild-service-role

codebuild-eks_codebuild-service-role Info

Summary

Creation date
July 11, 2025, 17:22 (UTC-04:00)

Last activity
2 days ago

ARN
arn:aws:iam::185187793433:role/service-role/codebuild-eks_codebuild-service-role

Maximum session duration
1 hour

Permissions policies (7) Info

You can attach up to 10 managed policies.

Filter by Type: All types

Policy name	Type	Attached entities
AmazonEC2ContainerRegistryPowerUser	AWS managed	2
AmazonEKS_CNI_Policy	AWS managed	2
AmazonEKSClusterPolicy	AWS managed	5
AmazonEKSWorkerNodePolicy	AWS managed	2
AmazonS3ReadOnlyAccess	AWS managed	1
CloudWatchLogsFullAccess	AWS managed	1
CodeBuildBasePolicy-eks_codebuild-us-east-1	Customer managed	1

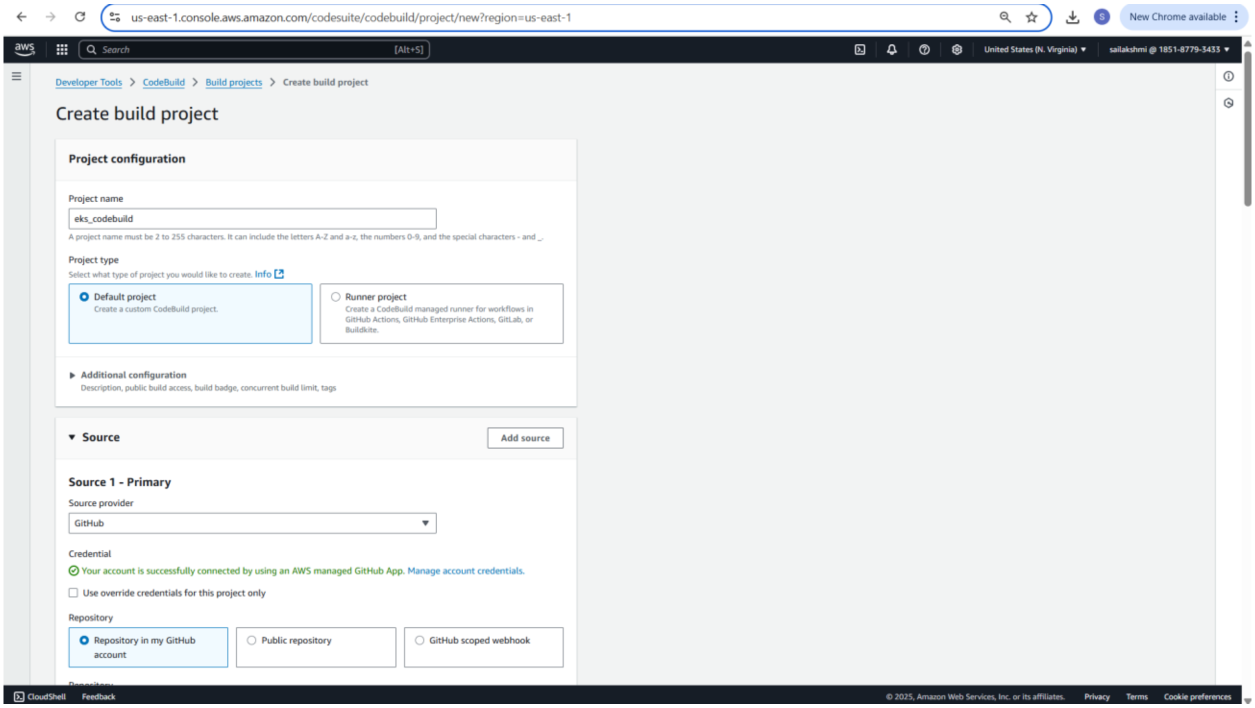
Permissions boundary (not set)

Generate policy based on CloudTrail events

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

Step: 5 Create CodeBuild project

- Source: connect to your GitHub repo
- Environment: Ubuntu, Docker enabled (aws/codebuild/standard:5.0)
- Buildspec: use your repo's buildspec.yml
- Service Role: use the role with permissions above



us-east-1.console.aws.amazon.com/codesuite/codebuild/project/new?region=us-east-1

Developer Tools > CodeBuild > Build projects > Create build project

Create build project

Project configuration

Project name

eks_codebuild

A project name must be 2 to 255 characters. It can include the letters A-Z and a-z, the numbers 0-9, and the special characters - and _.

Project type

Select what type of project you would like to create. Info

● Default project
Create a custom CodeBuild project.

○ Runner project
Create a CodeBuild managed runner for workflows in GitHub Actions, GitHub Enterprise Actions, GitLab, or Buildkite.

▸ Additional configuration
Description, public build access, build badge, concurrent build limit, tags

▾ Source

Add source

Source 1 - Primary

Source provider

GitHub

Credential

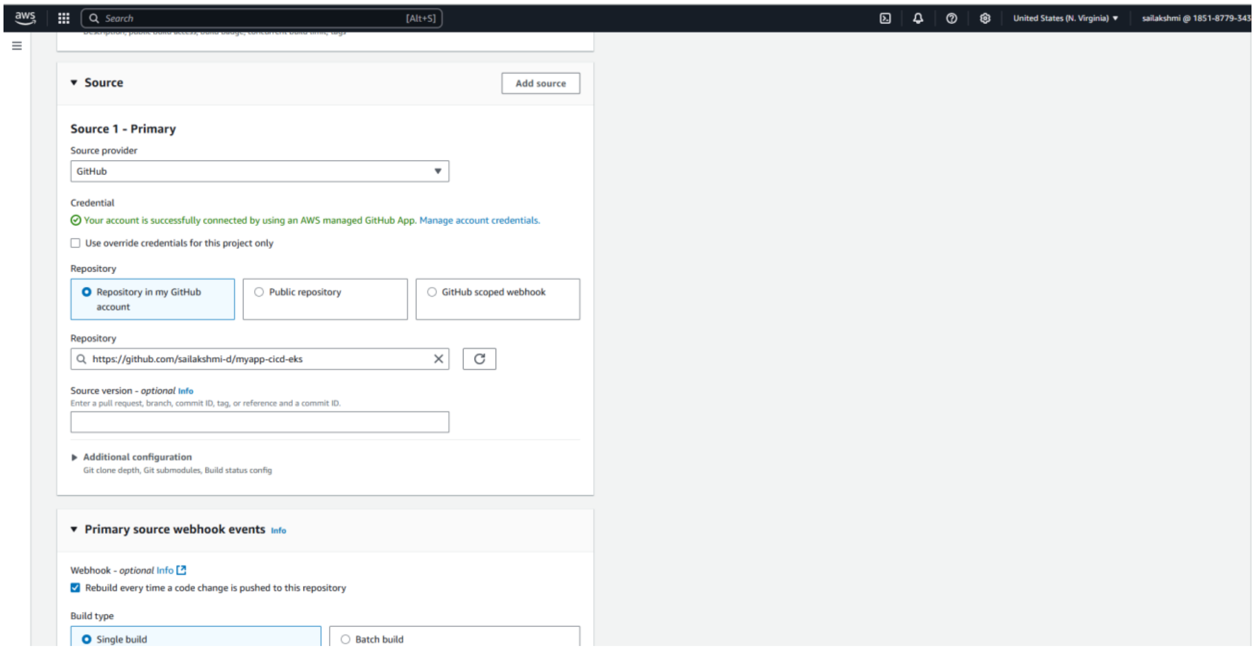
Your account is successfully connected by using an AWS managed GitHub App. Manage account credentials.

□ Use override credentials for this project only

Repository

● Repository in my GitHub account ○ Public repository ○ GitHub scoped webhook

New Chrome available
Search
[Alt+S]
United States (N. Virginia)
sailakshmi @ 1851-8779-3433
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences



▾ Source

Add source

Source 1 - Primary

Source provider

GitHub

Credential

Your account is successfully connected by using an AWS managed GitHub App. Manage account credentials.

□ Use override credentials for this project only

Repository

● Repository in my GitHub account ○ Public repository ○ GitHub scoped webhook

Repository

https://github.com/sailakshmi-d/myapp-cicd-eks

Source version - *optional* Info

Enter a pull request, branch, commit ID, tag, or reference and a commit ID.

▸ Additional configuration
Git clone depth, Git submodules, Build status config

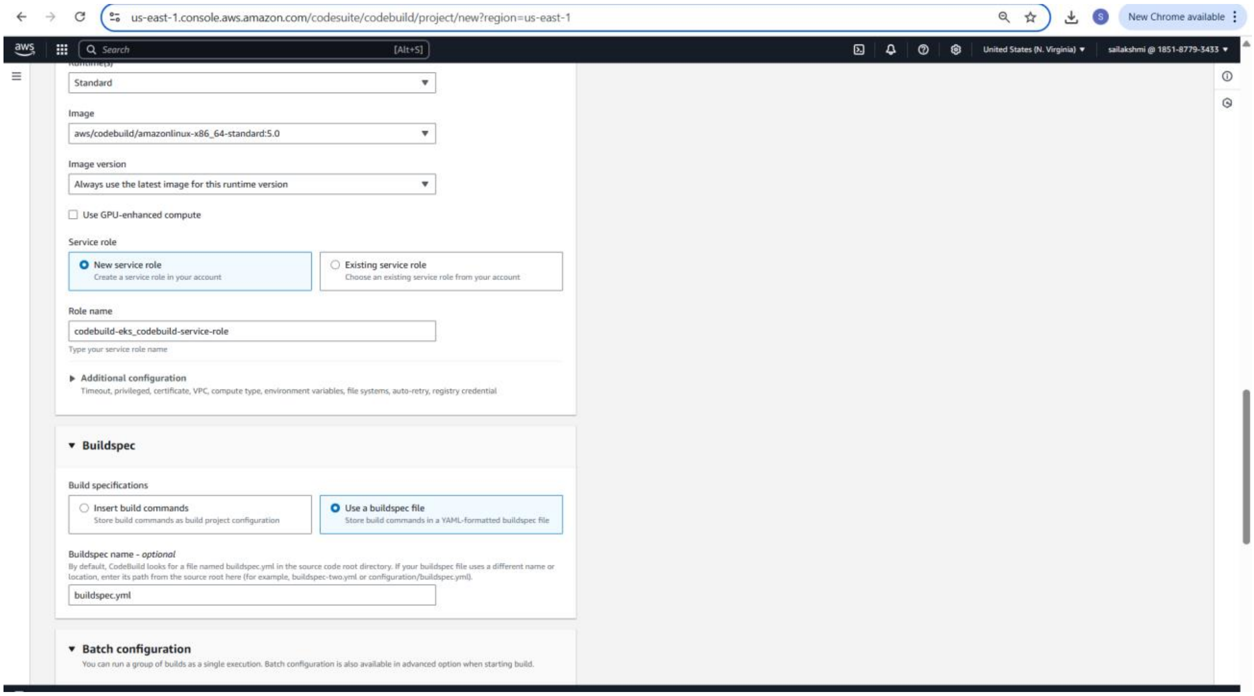
▾ Primary source webhook events Info

Webhook - *optional* Info

☑ Rebuild every time a code change is pushed to this repository

Build type

● Single build ○ Batch build


us-east-1.console.aws.amazon.com/codesuite/codebuild/project/new?region=us-east-1
New Chrome available
Search [Alt+S]
United States (N. Virginia)
sailakshmi @ 1851-8779-3433

Standard

Image
aws/codebuild/amazonlinux-x86_64-standard:5.0

Image version
Always use the latest image for this runtime version

☐ Use GPU-enhanced compute

Service role
● New service role
Create a service role in your account
○ Existing service role
Choose an existing service role from your account

Role name
codebuild-eks_codebuild-service-role
Type your service role name

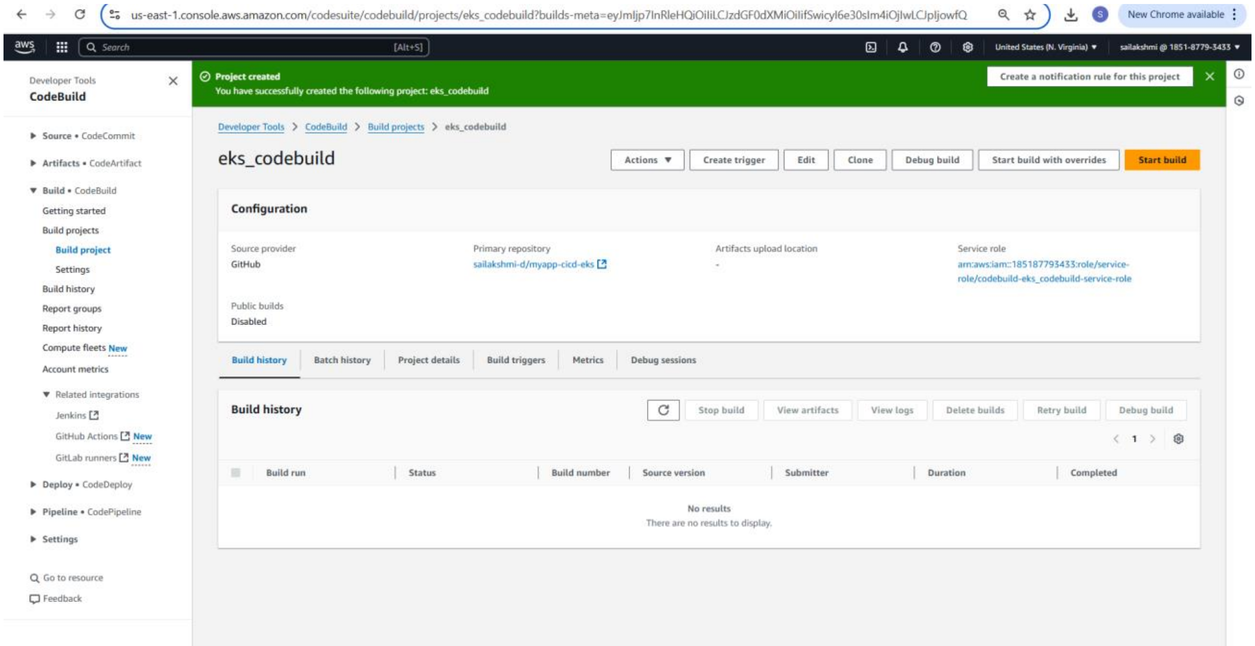
▶ Additional configuration
Timeout, privileged, certificate, VPC, compute type, environment variables, file systems, auto-retry, registry credential

▼ Buildspec

Build specifications
○ Insert build commands
Store build commands as build project configuration
● Use a buildspec file
Store build commands in a YAML-formatted buildspec file

Buildspec name - *optional*
By default, CodeBuild looks for a file named buildspec.yml in the source code root directory. If your buildspec file uses a different name or location, enter its path from the source root here (for example, buildspec-two.yml or configuration/buildspec.yml).
buildspec.yml

▼ Batch configuration
You can run a group of builds as a single execution. Batch configuration is also available in advanced option when starting build.



Project created
You have successfully created the following project: eks_codebuild

eks_codebuild

Configuration

Source provider
GitHub

Primary repository
sailakshmi-d/myapp-cicd-eks

Artifacts upload location
-

Service role
arn:aws:iam::185187793433:role/service-role/codebuild-eks_codebuild-service-role

Public builds
Disabled

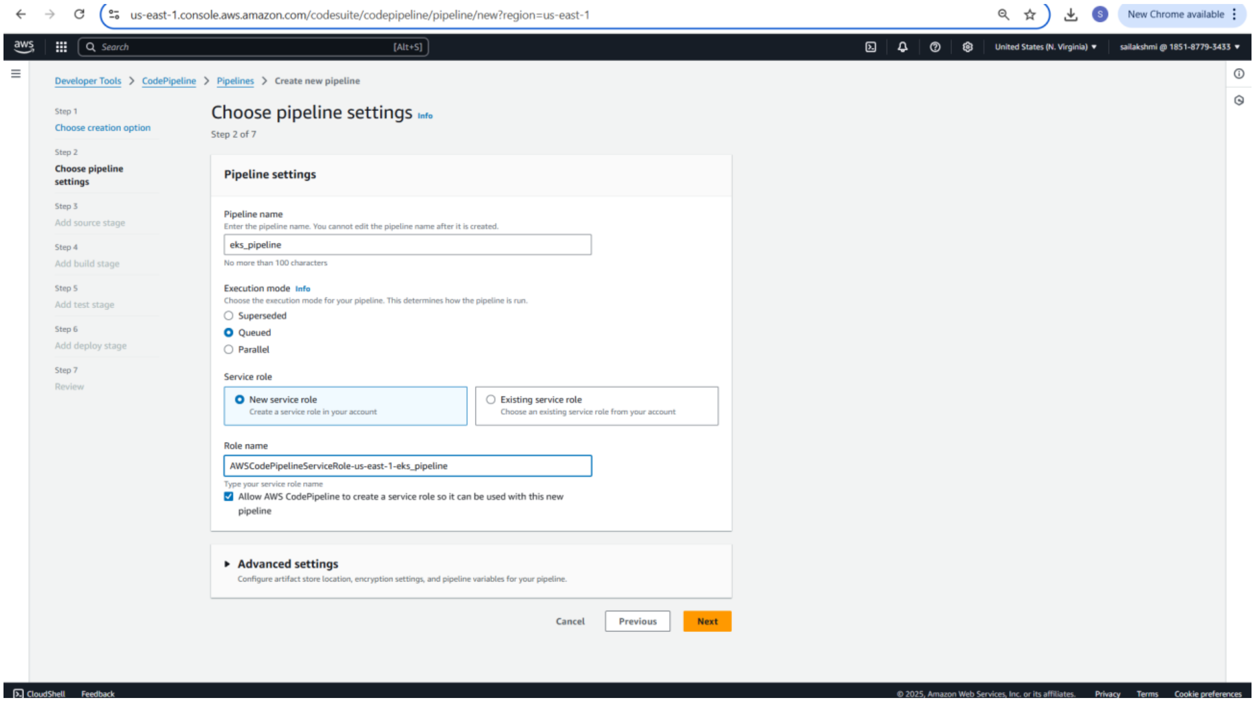
Build history

Build run	Status	Build number	Source version	Submitter	Duration	Completed

No results
There are no results to display.

Step: 6 Create a Code pipeline

- Source stage: GitHub repo
- Build stage: CodeBuild project
- Deploy stage: optional (our deploy is inside buildspec via kubectl)



us-east-1.console.aws.amazon.com/codesuite/codepipeline/pipeline/new?region=us-east-1

United States (N. Virginia) ▾ sailakshmi @ 1851-8779-3433 ▾

Developer Tools > CodePipeline > Pipelines > Create new pipeline

Step 1 Choose creation option
Step 2 **Choose pipeline settings**
Step 3 Add source stage
Step 4 Add build stage
Step 5 Add test stage
Step 6 Add deploy stage
Step 7 Review

Choose pipeline settings Info

Step 2 of 7

Pipeline settings

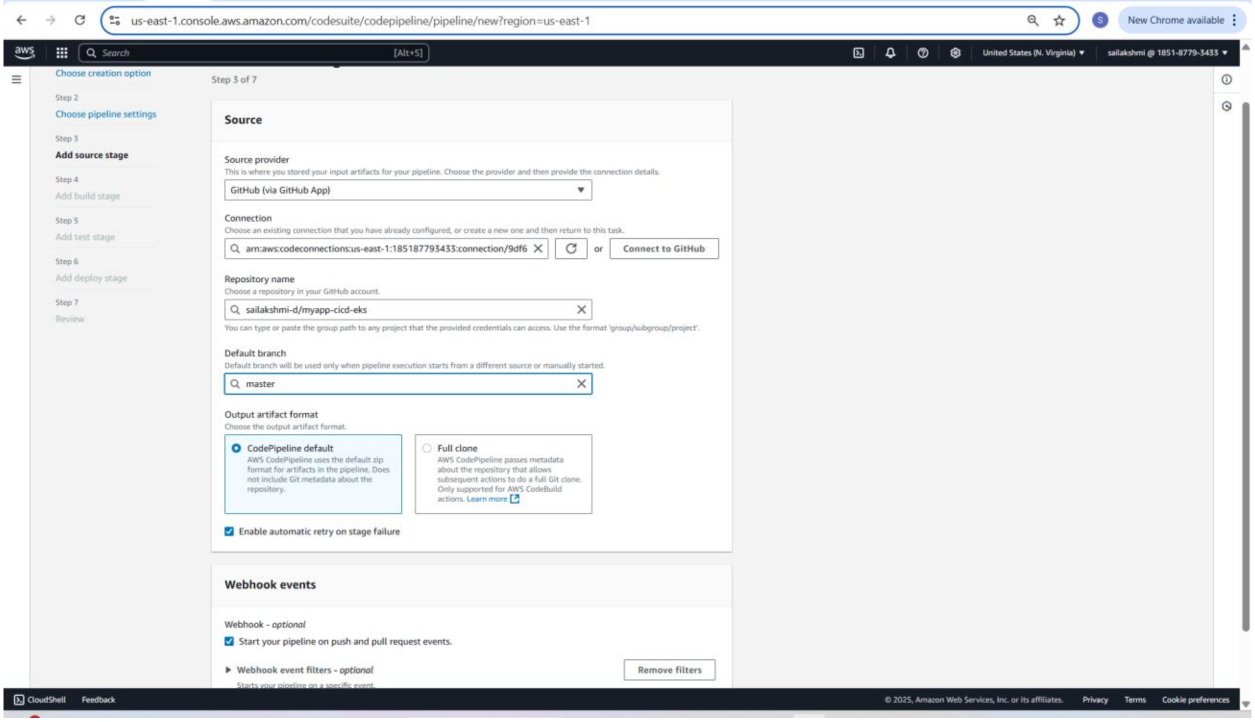
Pipeline name
Enter the pipeline name. You cannot edit the pipeline name after it is created.
eks_pipeline
No more than 100 characters

Execution mode Info
Choose the execution mode for your pipeline. This determines how the pipeline is run.
○ Superseded
● Queued
○ Parallel

Service role
● New service role
Create a service role in your account
○ Existing service role
Choose an existing service role from your account

Role name
AWSCodePipelineServiceRole-us-east-1-eks_pipeline
Type your service role name
☑ Allow AWS CodePipeline to create a service role so it can be used with this new pipeline

▸ **Advanced settings**
Configure artifact store location, encryption settings, and pipeline variables for your pipeline.



us-east-1.console.aws.amazon.com/codesuite/codepipeline/pipeline/new?region=us-east-1

United States (N. Virginia) ▾ sailakshmi @ 1851-8779-3433 ▾

Choose creation option
Step 2 Choose pipeline settings
Step 3 **Add source stage**
Step 4 Add build stage
Step 5 Add test stage
Step 6 Add deploy stage
Step 7 Review

Step 3 of 7

Source

Source provider
This is where you stored your input artifacts for your pipeline. Choose the provider and then provide the connection details.
GitHub (via GitHub App)

Connection
Choose an existing connection that you have already configured, or create a new one and then return to this task.
arn:aws:codeconnections:us-east-1:185187793433:connection/9df6 or Connect to GitHub

Repository name
Choose a repository in your GitHub account.
sailakshmi-d/myapp-cicd-eks
You can type or paste the group path to any project that the provided credentials can access. Use the format 'group/subgroup/project'.

Default branch
Default branch will be used only when pipeline execution starts from a different source or manually started.
master

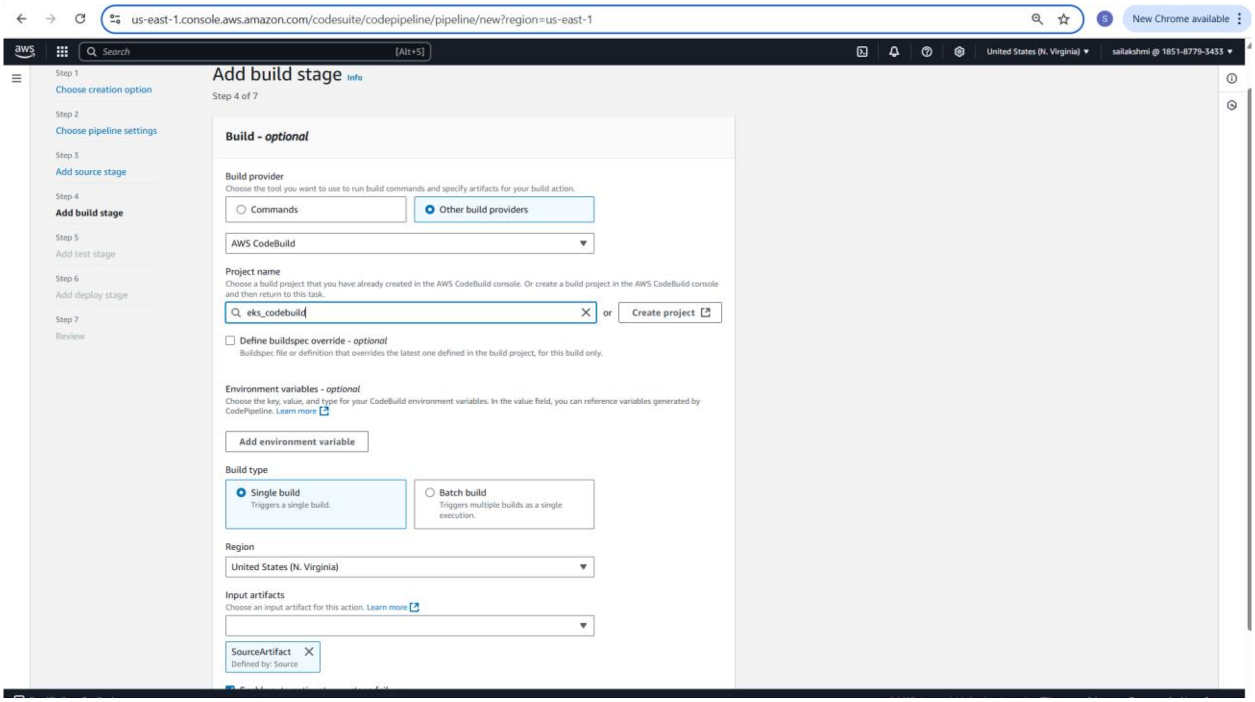
Output artifact format
Choose the output artifact format.
● CodePipeline default
AWS CodePipeline uses the default zip format for artifacts in the pipeline. Does not include Git metadata about the repository.
○ Full clone
AWS CodePipeline passes metadata about the repository that allows subsequent actions to do a full Git clone. Only supported for AWS CodeBuild actions. Learn more

☑ Enable automatic retry on stage failure

Webhook events

Webhook - *optional*
☑ Start your pipeline on push and pull request events.

▸ Webhook event filters - *optional*
Starts your pipeline on a specific event.



Add build stage Info

Step 4 of 7

Build - *optional*

Build provider
Choose the tool you want to use to run build commands and specify artifacts for your build action.
○ Commands ● Other build providers

AWS CodeBuild ▾

Project name
Choose a build project that you have already created in the AWS CodeBuild console. Or create a build project in the AWS CodeBuild console and then return to this task.
eks_codebuild or Create project

□ Define buildspec override - *optional*
Buildspec file or definition that overrides the latest one defined in the build project, for this build only.

Environment variables - *optional*
Choose the key, value, and type for your CodeBuild environment variables. In the value field, you can reference variables generated by CodePipeline. Learn more

Add environment variable

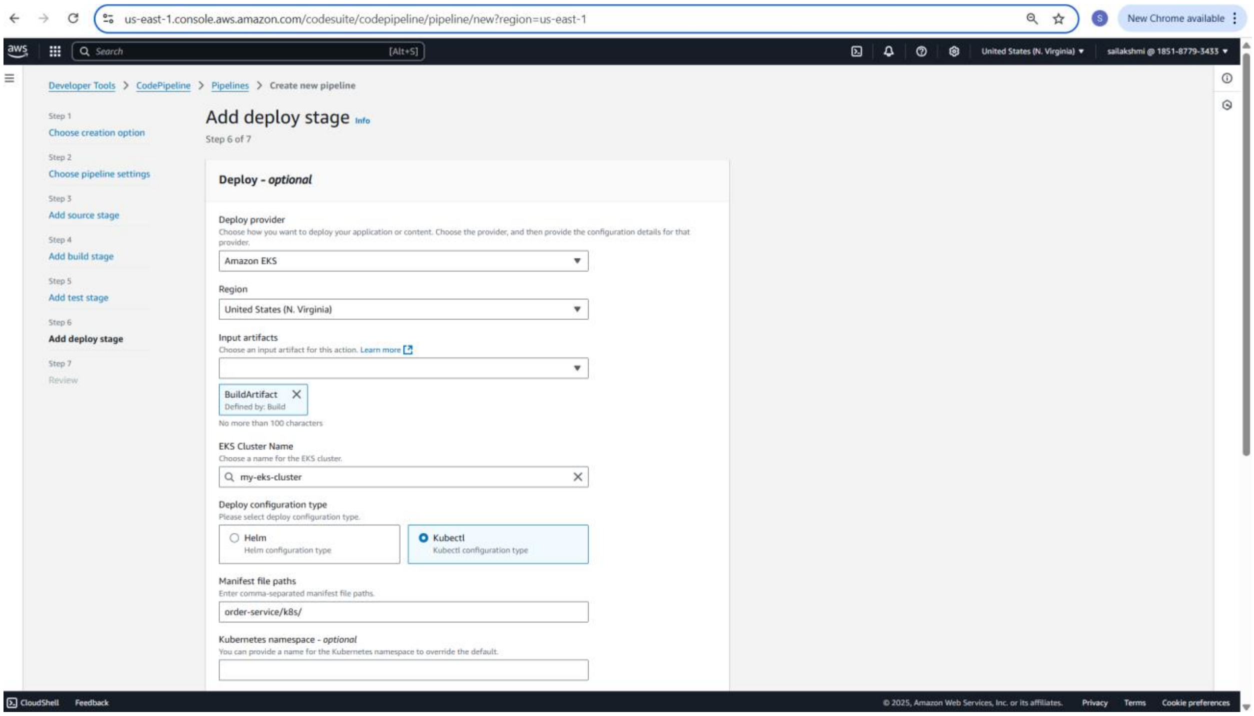
Build type
● Single build
Triggers a single build.
○ Batch build
Triggers multiple builds as a single execution.

Region
United States (N. Virginia) ▾

Input artifacts
Choose an input artifact for this action. Learn more

SourceArtifact
Defined by: Source

Code deploy


us-east-1.console.aws.amazon.com/codesuite/codepipeline/pipeline/new?region=us-east-1
New Chrome available
aws
Search [Alt+S]
United States (N. Virginia) ▾
sailakshmi @ 1851-8779-3433 ▾
Developer Tools > CodePipeline > Pipelines > Create new pipeline
Step 1
Choose creation option
Step 2
Choose pipeline settings
Step 3
Add source stage
Step 4
Add build stage
Step 5
Add test stage
Step 6
Add deploy stage
Step 7
Review
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Add deploy stage Info

Step 6 of 7

Deploy - *optional*

Deploy provider
Choose how you want to deploy your application or content. Choose the provider, and then provide the configuration details for that provider.
Amazon EKS ▾

Region
United States (N. Virginia) ▾

Input artifacts
Choose an input artifact for this action. Learn more

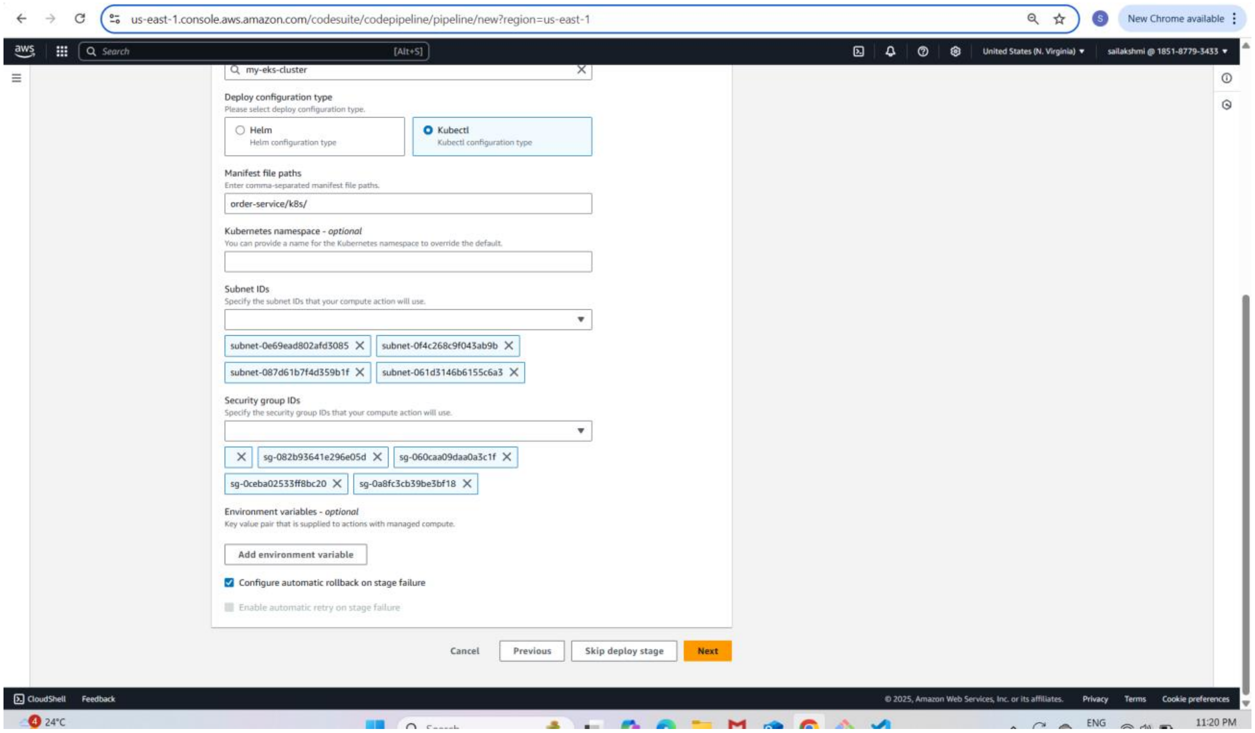
BuildArtifact
Defined by: Build
No more than 100 characters

EKS Cluster Name
Choose a name for the EKS cluster.
my-eks-cluster

Deploy configuration type
Please select deploy configuration type.
○ Helm
Helm configuration type
● Kubectl
Kubectl configuration type

Manifest file paths
Enter comma-separated manifest file paths.
order-service/k8s/

Kubernetes namespace - *optional*
You can provide a name for the Kubernetes namespace to override the default.



us-east-1.console.aws.amazon.com/codesuite/codepipeline/pipeline/new?region=us-east-1
New Chrome available
Search [Alt+S]
United States (N. Virginia) ▾
sailakshmi @ 1851-8779-3433 ▾

my-eks-cluster

Deploy configuration type
Please select deploy configuration type.

○ Helm
Helm configuration type

● Kubectl
Kubectl configuration type

Manifest file paths
Enter comma-separated manifest file paths.

order-service/k8s/

Kubernetes namespace - *optional*
You can provide a name for the Kubernetes namespace to override the default.

Subnet IDs
Specify the subnet IDs that your compute action will use.

subnet-0e69ead802afd3085 ✕ subnet-0f4c268c9f043ab9b ✕
subnet-087d61b7f4d359b1f ✕ subnet-061d3146b6155c6a3 ✕

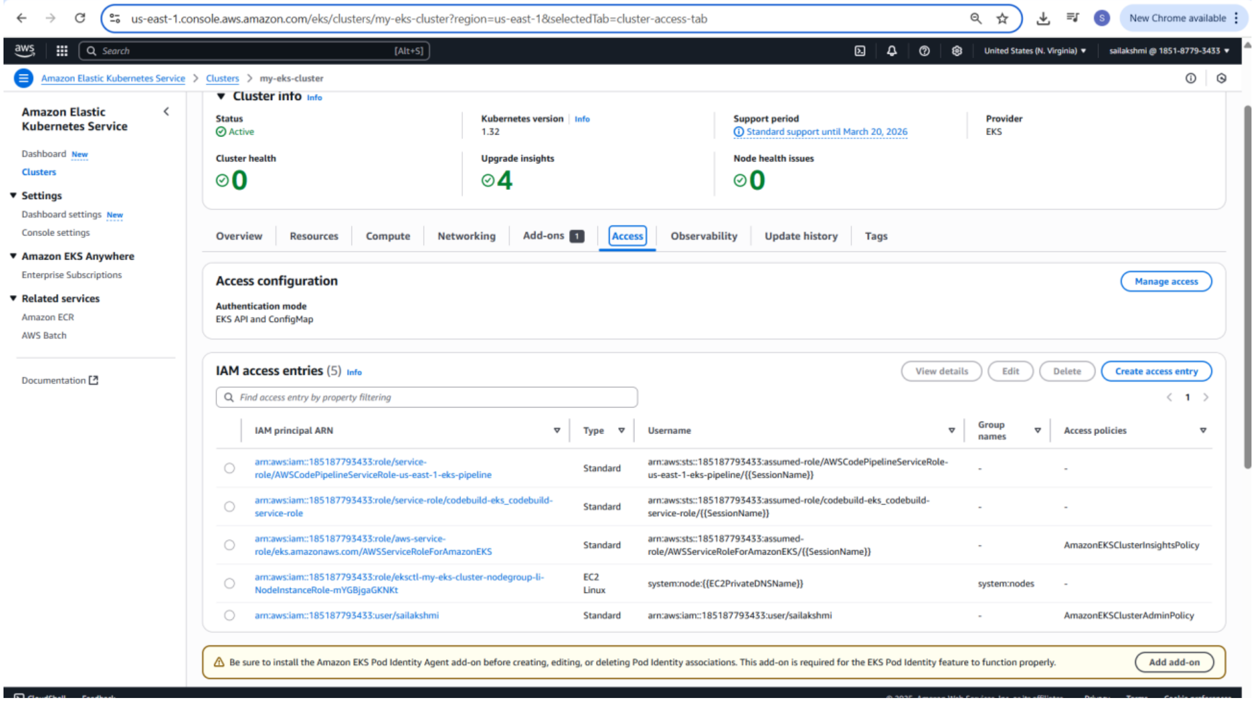
Security group IDs
Specify the security group IDs that your compute action will use.

✕ sg-082b93641e296e05d ✕ sg-060caa09daa0a3c1f ✕
sg-0ceba02533ff8bc20 ✕ sg-0a8fc3cb39be3bf18 ✕

Environment variables - *optional*
Key value pair that is supplied to actions with managed compute.

☑ Configure automatic rollback on stage failure

□ Enable automatic retry on stage failure



us-east-1.console.aws.amazon.com/eks/clusters/my-eks-cluster?region=us-east-1&selectedTab=cluster-access-tab
New Chrome available
Search [Alt+S]
United States (N. Virginia) ▾
sailakshmi @ 1851-8779-3433 ▾
Amazon Elastic Kubernetes Service > Clusters > my-eks-cluster
Amazon Elastic Kubernetes Service
Dashboard New
Clusters
Settings
Dashboard settings New
Console settings
Amazon EKS Anywhere
Enterprise Subscriptions
Related services
Amazon ECR
AWS Batch
Documentation

Cluster info Info

Status
Active

Kubernetes version Info
1.32

Support period
Standard support until March 20, 2026

Provider
EKS

Cluster health
0

Upgrade insights
4

Node health issues
0

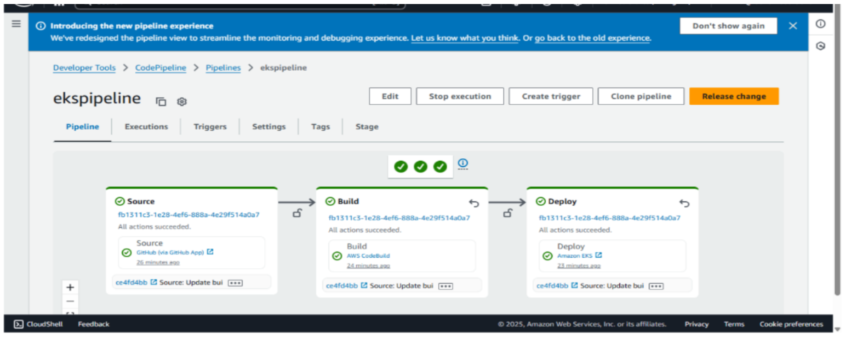
Access configuration

Authentication mode
EKS API and ConfigMap

IAM access entries (5) Info

	IAM principal ARN	Type	Username	Group names	Access policies
○	arn:aws:iam::185187793433:role/service-role/AWSCodePipelineServiceRole-us-east-1-eks-pipeline	Standard	arn:aws:sts::185187793433:assumed-role/AWSCodePipelineServiceRole-us-east-1-eks-pipeline/{{SessionName}}	-	-
○	arn:aws:iam::185187793433:role/service-role/codebuild-eks_codebuild-service-role	Standard	arn:aws:sts::185187793433:assumed-role/codebuild-eks_codebuild-service-role/{{SessionName}}	-	-
○	arn:aws:iam::185187793433:role/aws-service-role/eks.amazonaws.com/AWSServiceRoleForAmazonEKS	Standard	arn:aws:sts::185187793433:assumed-role/AWSServiceRoleForAmazonEKS/{{SessionName}}	-	AmazonEKSClusterInsightsPolicy
○	arn:aws:iam::185187793433:role/eksctl-my-eks-cluster-nodegroup-li-NodeInstanceRole-mYGBjgaGKNKt	EC2 Linux	system:node:{{EC2PrivateDNSName}}	system:nodes	-
○	arn:aws:iam::185187793433:user/sailakshmi	Standard	arn:aws:iam::185187793433:user/sailakshmi	-	AmazonEKSClusterAdminPolicy

⚠ Be sure to install the Amazon EKS Pod Identity Agent add-on before creating, editing, or deleting Pod Identity associations. This add-on is required for the EKS Pod Identity feature to function properly.



We've redesigned the pipeline view to streamline the monitoring and debugging experience. Let us know what you think. Or go back to the old experience.

Source
fb1311c3-1e28-4ef6-888a-4e29f514a0a7
All actions succeeded.
Source
GitHub (via GitHub App)
26 minutes ago
ce4fd4bb Source: Update bui

Build
fb1311c3-1e28-4ef6-888a-4e29f514a0a7
All actions succeeded.
Build
AWS CodeBuild
24 minutes ago
ce4fd4bb Source: Update bui

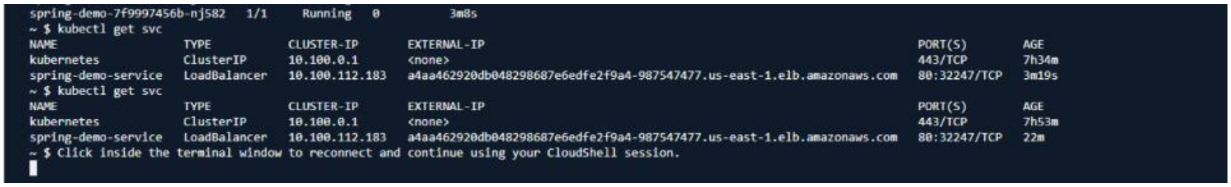
Deploy
fb1311c3-1e28-4ef6-888a-4e29f514a0a7
All actions succeeded.
Deploy
Amazon EKS
23 minutes ago
ce4fd4bb Source: Update bui

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

Trigger Pipeline

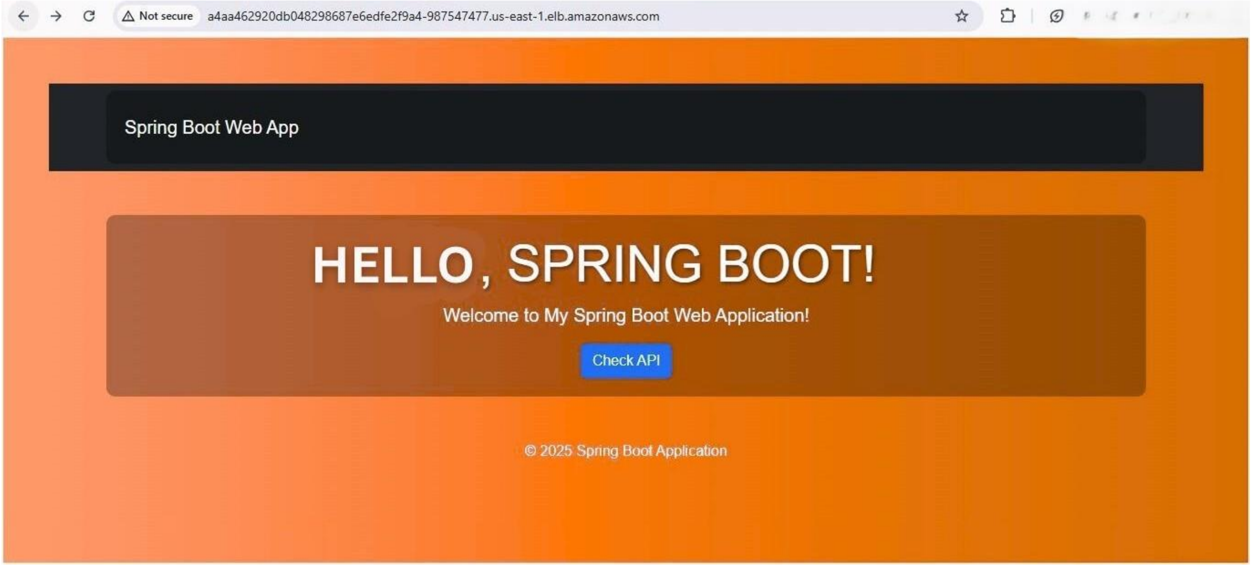
- Push code changes to source repository.
- Pipeline automatically triggers build and deploy.

Verify Deployment and service



```
spring-demo-7f9997456b-nj582   1/1     Running   0          3m8s
~ $ kubectl get svc
NAME                  TYPE           CLUSTER-IP       EXTERNAL-IP                                                                PORT(S)        AGE
kubernetes            ClusterIP      10.100.0.1       <none>                                                                     443/TCP        7h34m
spring-demo-service   LoadBalancer   10.100.112.183   a4aa462920db048298687e6edfe2f9a4-987547477.us-east-1.elb.amazonaws.com   80:32247/TCP   3m19s
~ $ kubectl get svc
NAME                  TYPE           CLUSTER-IP       EXTERNAL-IP                                                                PORT(S)        AGE
kubernetes            ClusterIP      10.100.0.1       <none>                                                                     443/TCP        7h53m
spring-demo-service   LoadBalancer   10.100.112.183   a4aa462920db048298687e6edfe2f9a4-987547477.us-east-1.elb.amazonaws.com   80:32247/TCP   22m
~ $ Click inside the terminal window to reconnect and continue using your CloudShell session.
█
```

Access the application via Load Balancer or Ingress.



Not secure
a4aa462920db048298687e6edfe2f9a4-987547477.us-east-1.elb.amazonaws.com
Spring Boot Web App
Check API

HELLO, SPRING BOOT!

Welcome to My Spring Boot Web Application!

© 2025 Spring Boot Application