

Trendstore

GitHub Link : <https://github.com/Vennilavan12/Trend.git>

Overview

React application deployment to a production-ready state.

Brief description of the application, its purpose, and key features.

Example:

This project showcases deploying a React application from code to a production-ready, containerized application on Kubernetes (Amazon EKS) using a CI/CD pipeline with Jenkins.

Architecture Overview

- Application: React
- Containerization: Docker
- Orchestration: Kubernetes (Amazon EKS)
- CI/CD Pipeline: Jenkins
- Cloud Provider: AWS
- Monitoring & Observability: Prometheus and Grafana dashboards

Setup Instructions

Prerequisites

- Docker
- kubectl
- AWS CLI
- eksctl
- Kubectl
- Helm

CI/CD Pipeline Explanation

Pipeline Stages

1. Pipeline Stages
2. Code Checkout
3. Push Docker Image to Docker Hub
4. Deploy to Kubernetes

Pipeline Flow

- On code push to the Git repository, Jenkins automatically triggers the build via webhooks.
- The Docker image is built and pushed to Docker Hub.
- Kubernetes manifests are automatically applied to deploy the application.

Project Screenshots and Proof of Implementation

This section includes screenshots that provide visual proof of the successful implementation and execution of the project. The screenshots demonstrate key stages and components of the deployment, including:

- ❖ Source code repositories and project structure
- ❖ Successful Docker image build and push to Docker Hub
- ❖ Jenkins CI/CD pipeline execution (checkout, build, push, and deploy stages)
- ❖ Amazon EKS cluster creation and running status
- ❖ Kubernetes deployments and services running successfully
- ❖ Application accessibility via the exposed application URL

- ❖ Prometheus metrics collection and Grafana dashboards displaying real-time cluster and application health

These screenshots collectively validate the end-to-end workflow, from source code to a production-ready, containerized application deployed on Kubernetes.

GIT URL : Project files

The full project setup and configuration artifacts are version-controlled and available in the Git repository.

URL :

https://github.com/udhayakumarethiraj1/enterprise/tree/a03c4c416cbb50ea36980bd640562255cff3e0fc/Project_2_Trend_store

Trendstore.git (Private): Contains all supporting configuration and infrastructure files, including the Dockerfile, Jenkinsfile, Kubernetes manifests (deployment.yaml, service.yaml, node-exporter-lb.yaml, kube-state-metrics-lb.yaml), .gitignore, and Terraform files.

URL : <https://github.com/udhayakumarethiraj-git/Trendstore.git>

Stage : Application

Git Repo Cloned to my repo (Trend)

https://github.com/udhayakumarethiraj-git/project_2_trend_store.git

```
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ git clone https://github.com/Vennilavan12/Trend.git
Cloning into 'Trend'...
remote: Enumerating objects: 77, done.
remote: Counting objects: 100% (2/2), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 77 (delta 0), reused 0 (delta 0), pack-reused 75 (from 1)
Receiving objects: 100% (77/77), 8.58 MiB | 14.95 MiB/s, done.
Resolving deltas: 100% (1/1), done.
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ ll
total 12
drwxrwxr-x 3 ubuntu ubuntu 4096 Jan 20 05:54 ./
drwxrwxr-x 3 ubuntu ubuntu 4096 Jan 20 05:54 ../
drwxrwxr-x 4 ubuntu ubuntu 4096 Jan 20 05:54 Trend/
```

```
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ git remote set-url origin git@github.com:udhayakumarethiraj-git/Trendstore.git
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ git push origin main
Enumerating objects: 77, done.
Counting objects: 100% (77/77), done.
Delta compression using up to 2 threads
Compressing objects: 100% (76/76), done.
Writing objects: 100% (77/77), 8.58 MiB | 2.33 MiB/s, done.
Total 77 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), done.
To github.com:udhayakumarethiraj-git/Trendstore.git
 * [new branch]      main -> main
```

The screenshot shows the GitHub web interface for a repository named 'Trendstore' under the user 'udhayakumarethiraj-git'. The repository is public and has 1 branch (main) and 0 tags. The main content area shows a commit titled 'Ubuntu repo cloned' from 23 minutes ago. Below this, there is a section for 'Add a README' with a button to 'Add a README'. The right sidebar contains sections for 'About' (no description), 'Activity' (0 stars, 0 watching, 0 forks), 'Releases' (no releases published), and 'Packages' (no packages published).

Stage : Docker

Containerize the application by creating a Dockerfile, building the image, and testing the output.

Dockerfile

```
FROM nginx:alpine

# Remove default nginx static files
RUN rm -rf /usr/share/nginx/html/*

# Copy Vite build output to nginx
COPY dist /usr/share/nginx/html

# Expose port 80
EXPOSE 80

CMD ["nginx", "-g", "daemon off;"]
```

```
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ more Dockerfile
FROM nginx:alpine

# Remove default nginx static files
RUN rm -rf /usr/share/nginx/html/*

# Copy Vite build output to nginx
COPY dist /usr/share/nginx/html

# Expose port 80
EXPOSE 80

CMD ["nginx", "-g", "daemon off;"]
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$
```

```
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ vi Dockerfile
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ docker build -t trendstore:latest .
[+] Building 2.9s (8/8) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 260B
=> [internal] load metadata for docker.io/library/nginx:alpine
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [1/3] FROM docker.io/library/nginx:alpine@sha256:b0f7830b6bfaa1258f45d94c240ab668ced1b3651c8a222aefe6683447c7bf55
=> => resolve docker.io/library/nginx:alpine@sha256:b0f7830b6bfaa1258f45d94c240ab668ced1b3651c8a222aefe6683447c7bf55
=> [internal] load build context
=> => transferring context: 9.24MB
=> CACHED [2/3] RUN rm -rf /usr/share/nginx/html/*
=> [3/3] COPY dist /usr/share/nginx/html
=> exporting to image
=> => exporting layers
=> => exporting manifest sha256:3903c98e8483d64f6ff186eba40d68333737a1392d2a83a8949da88fe12bb2b6
=> => exporting config sha256:0fceb60cedaf899115369c9eaaefb9c2f61d17f7dd640ea65486708e014d1a0
=> => exporting attestation manifest sha256:f6b0959dd0b966b42c028d72dea9435015b013124a2b0e576f78af0855f7d565
=> => exporting manifest list sha256:bbb94072eab1136beb17f31ce308b910652626e0448a840949fb81fc5e67eb2e
=> => naming to docker.io/library/trendstore:latest
=> => unpacking to docker.io/library/trendstore:latest
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ docker images
```

IMAGE	ID	DISK USAGE	CONTENT SIZE	EXTRA	In Use
758234806674.dkr.ecr.ap-south-1.amazonaws.com/brain_task_project:latest	6fc15216dbed	92.7MB	26MB	U	
brain-task:latest	a4e360937599	92.7MB	26MB		
trendstore:latest	bbb94072eab1	111MB	34.9MB		

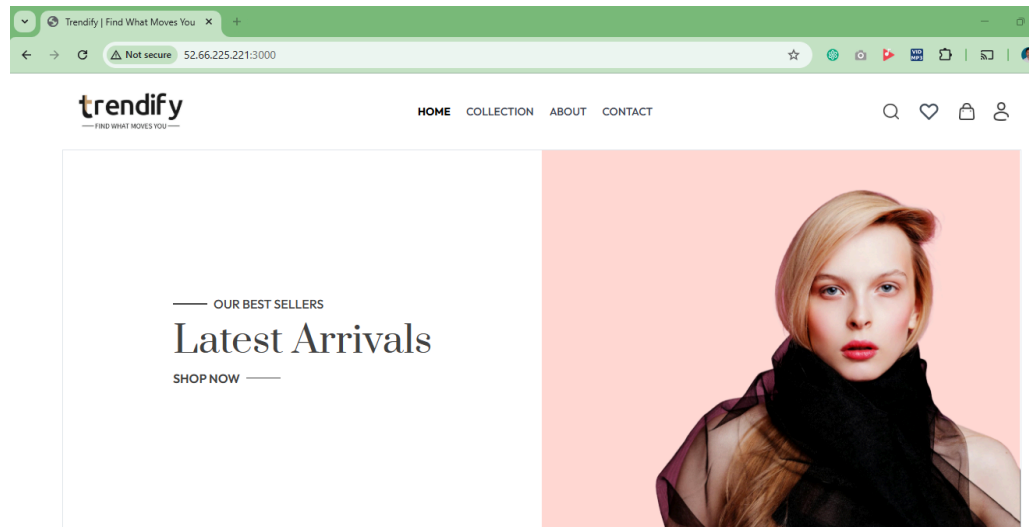
```
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ docker run -d -p 3000:80 trendstore:latest
cd33d7889d44432c60bee67847b54b08981cdba7553d38784c4fee300b53df53
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ more Dockerfile
```

```

ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS
AMBS          cd33d7889d44   trendstore:latest       "/docker-entrypoint..."  4 minutes ago  Up 4 minutes   0.0.0.0:3000->80/tcp, [::]:3000->80/tcp
ostalgic_fermi
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$

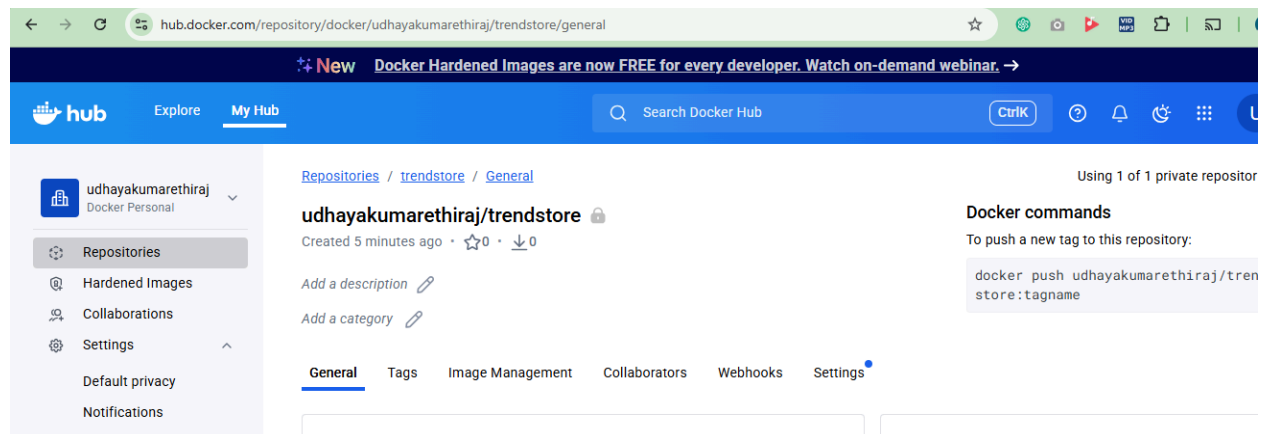
```

Output Test result



Stage : DockerHUB

Docker Hub repo created



```
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ docker login -u udhayakumarethiraj

Info → A Personal Access Token (PAT) can be used instead.
To create a PAT, visit https://app.docker.com/settings

Password:

WARNING! Your credentials are stored unencrypted in '/home/ubuntu/.docker/config.json'.
Configure a credential helper to remove this warning. See
https://docs.docker.com/go/credential-store/

Login Succeeded
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$
```

```
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ docker tag trendstore:latest udhayakumarethiraj/trendstore:latest
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ docker push udhayakumarethiraj/trendstore:latest
The push refers to repository [docker.io/udhayakumarethiraj/trendstore]
da7c973d8b92: Pushed
33f95a0f3229: Pushed
085c5e5aaa8e: Pushed
f5aa8e0adca0: Pushed
567f84da6fbd: Pushed
0abf9e567266: Pushed
25f453064fd3: Pushed
d1081cf8f0ea: Pushed
1074353eec0d: Pushed
e096540205d5: Pushed
f56ea78a1a5f: Pushed
latest: digest: sha256:bbb94072eab1136beb17f31ce308b910652626e0448a840949fb81fc5e67eb2e size: 856
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ docker images
```

IMAGE	ID	DISK USAGE	CONTENT SIZE	EXTRA	In Use
758234806674.dkr.ecr.ap-south-1.amazonaws.com/brain_task_project:latest	6fc15216dbed	92.7MB	2.6MB	U	
brain-task:latest	a4e360937599	92.7MB	2.6MB	U	
trendstore:latest	bbb94072eab1	111MB	34.9MB	U	
udhayakumarethiraj/trendstore:latest	bbb94072eab1	111MB	34.9MB	U	

Docker image pushed to Docker hub

hub.docker.com/repository/docker/udhayakumarethiraj/trendstore/general

New Docker Hardened Images are now FREE for every developer. Watch on

hub Explore My Hub Search Docker Hub

Repositories / trendstore / General

udhayakumarethiraj/trendstore

Last pushed 1 minute ago • Repository size: 33.3 MB • ☆ 0 • ↓ 0

Add a description

Add a category

General Tags Image Management Collaborators Webhooks Settings

Tags DOCKER SCOUT INACTIVE [Activate](#)

This repository contains 0 tag(s).

Tag	OS	Type	Pulled	Pushed
latest		Image	less than 1 day	1 minute

[See all](#)

Stage : Terraform

Using Terraform VPC, IAM, EC2 created with Jenkins and EKS deployed

```
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo/Terraform$ ll
total 28
drwxrwxr-x 2 ubuntu ubuntu 4096 Jan 27 08:23 ./
drwxrwxr-x 5 ubuntu ubuntu 4096 Jan 27 07:04 ../
-rw-rw-r-- 1 ubuntu ubuntu 1429 Jan 27 08:22 ec2.tf
-rw-rw-r-- 1 ubuntu ubuntu  996 Jan 27 08:22 eks.tf
-rw-rw-r-- 1 ubuntu ubuntu 1036 Jan 27 08:22 iam_user.tf
-rw-rw-r-- 1 ubuntu ubuntu 1843 Jan 27 08:22 user_data.sh
-rw-rw-r-- 1 ubuntu ubuntu 1564 Jan 27 08:22 vpc.tf
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo/Terraform$
```

VPC

```
ubuntu@ip-172-31-28-193: ~/projects/trend/trend-repo/Terraform/envs/trendstore
+ from_port      = 22
+ ipv6_cidr_blocks = []
+ prefix_list_ids = []
+ protocol       = "tcp"
+ security_groups = []
+ self           = false
+ to_port        = 22
# (1 unchanged attribute hidden)
},
]
+ name           = "trendstore-trendstore-ec2-sg"
+ name_prefix    = (known after apply)
+ owner_id       = (known after apply)
+ region         = "ap-south-1"
+ revoke_rules_on_delete = false
+ tags           = {
+   "Environment" = "trendstore"
+   "Name"        = "trendstore-trendstore-ec2-sg"
+   "Project"     = "trendstore"
+ }
+ tags_all       = {
+   "Environment" = "trendstore"
+   "Name"        = "trendstore-trendstore-ec2-sg"
+   "Project"     = "trendstore"
+ }
+ vpc_id         = "vpc-0366bf05cb6382061"
}

Plan: 2 to add, 0 to change, 0 to destroy.
module.ec2.aws_security_group.ec2_sg: Creating...
module.ec2.aws_security_group.ec2_sg: Creation complete after 2s [id=sg-0e5905d82ab83aef3]
module.ec2.aws_instance.this: Creating...
module.ec2.aws_instance.this: Still creating... [00m10s elapsed]
module.ec2.aws_instance.this: Creation complete after 12s [id=i-0be80b957a19a3e4e]
Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
```


aws [Search] [Alt+S] Asia Pacific (Mumbai) udhayakumarethiraj [7]

VPC > Your VPCs > vpc-0366bf05cb6382061

VPC dashboard <

AWS Global View [↗](#)

Filter by VPC ▾

▼ Virtual private cloud

Your VPCs

- Subnets
- Route tables
- Internet gateways
- Egress-only internet gateways
- DHCP option sets
- Elastic IPs
- Managed prefix lists

vpc-0366bf05cb6382061 / trendstore-trendstore-vpc

Details [Info](#)

VPC ID vpc-0366bf05cb6382061	State Available	Block Public Access Off	DNS hostnames Enabled
DNS resolution Enabled	Tenancy default	DHCP option set dopt-0beef96a379412ee8	Main route table rtb-0cebf44cb929d795b trendstore-trendstore-vp
Main network ACL acl-081f4a34834da5e9b / trendstore-trendstore-vpc-default	Default VPC No	IPv4 CIDR 10.0.0.0/16	IPv6 pool -
IPv6 CIDR (Network border group) -	Network Address Usage metrics Disabled	Route 53 Resolver DNS Firewall rule groups -	Owner ID 758234806674
Encryption control ID -	Encryption control mode -		

IAM

aws [Search] [Alt+S] Global

IAM > Users > user_trendstore

Identity and Access Management (IAM) <

[Search IAM](#)

Dashboard

▼ Access Management

- User groups
- Users**
- Roles
- Policies
- Identity providers
- Account settings
- Root access management
- Temporary delegation requests
- [New](#)

▼ Access reports

- Access Analyzer
- Resource analysis [New](#)
- Unused access
- Analyzer settings
- [Credential report](#)

user_trendstore [Info](#)

Summary

ARN arn:aws:iam::758234806674:user/user_trendstore	Console access Disabled	Access key 1 Create access key
Created January 20, 2026, 15:35 (UTC+05:30)	Last console sign-in -	

[Permissions](#) [Groups](#) [Tags \(3\)](#) [Security credentials](#) [Last Accessed](#)

Permissions policies (3)

Permissions are defined by policies attached to the user directly or through groups.

[Search](#) [Filter by Type](#) ▾

<input type="checkbox"/>	Policy name ↗	Type ↗	Attached via ↗
<input type="checkbox"/>	AdministratorAccess	AWS managed - job function	Directly
<input type="checkbox"/>	Billing	AWS managed - job function	Directly
<input type="checkbox"/>	IAMFullAccess	AWS managed	Directly

EC2 Instance

aws [Alt+S] Asia Pacific (Mumbai) udhayakumarethiraj (7582-34)

EC2 > Instances > i-0be80b957a19a3e4e

EC2

- Dashboard
- AWS Global View
- Events
- ▼ Instances
 - Instances
 - Instance Types
 - Launch Templates
 - Spot Requests
 - Savings Plans
 - Reserved Instances
 - Dedicated Hosts
 - Capacity Reservations
 - Capacity Manager [New](#)
- ▼ Images

Instance summary for i-0be80b957a19a3e4e (trendstore-trendstore-ec2) [Info](#)

Updated less than a minute ago

[Connect](#) [Instance state](#) [Actions](#)

Instance ID i-0be80b957a19a3e4e	Public IPv4 address 3.6.91.230 open address	Private IPv4 addresses 10.0.101.40
IPv6 address -	Instance state Running	Public DNS ec2-3-6-91-230.ap-south-1.compute.amazonaws.com open address
Hostname type IP name: ip-10-0-101-40.ap-south-1.compute.internal	Private IP DNS name (IPv4 only) ip-10-0-101-40.ap-south-1.compute.internal	Elastic IP addresses -
Answer private resource DNS name -	Instance type t3.small	AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations. Learn more
Auto-assigned IP address 3.6.91.230 [Public IP]	VPC ID vpc-0366bf05cb6382061 (trendstore-trendstore-vpc) i	

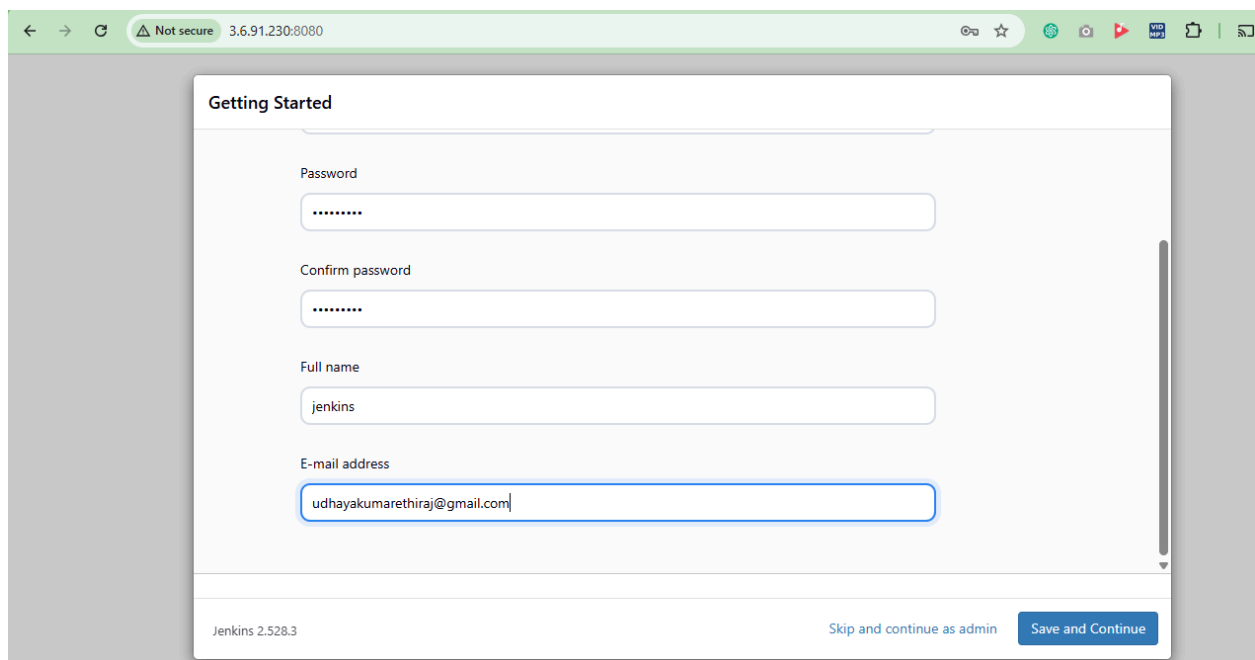
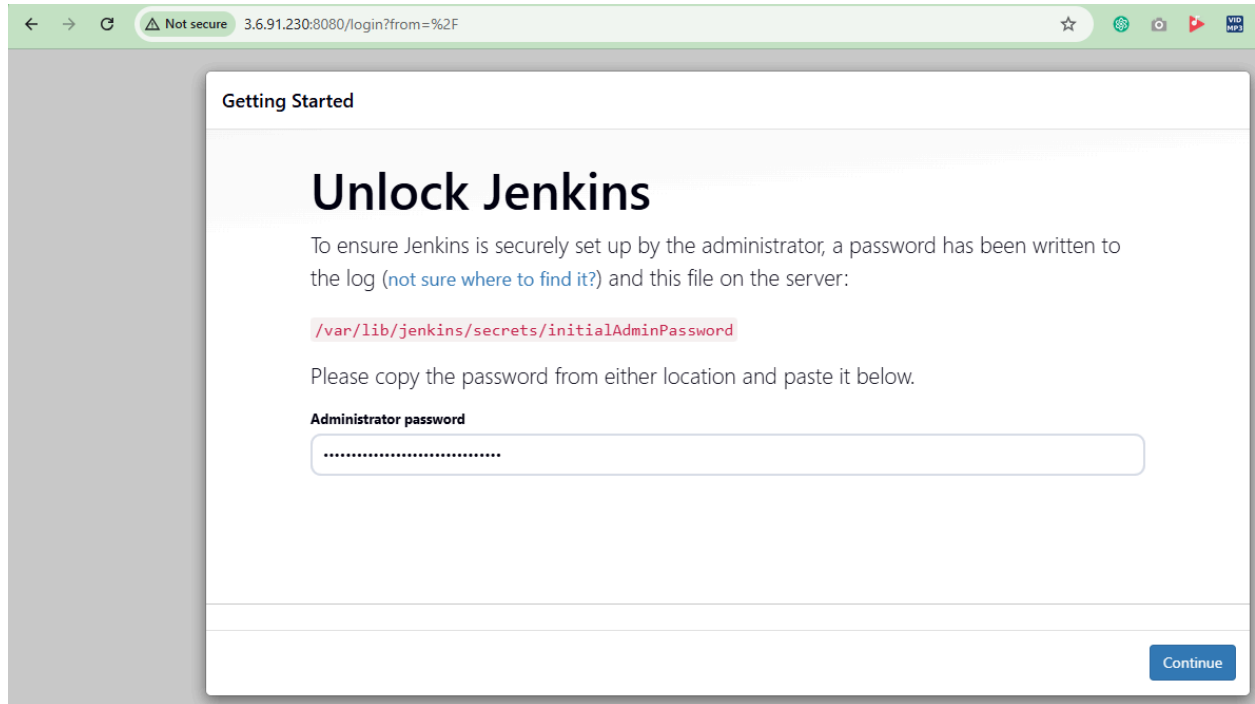
Installed packages

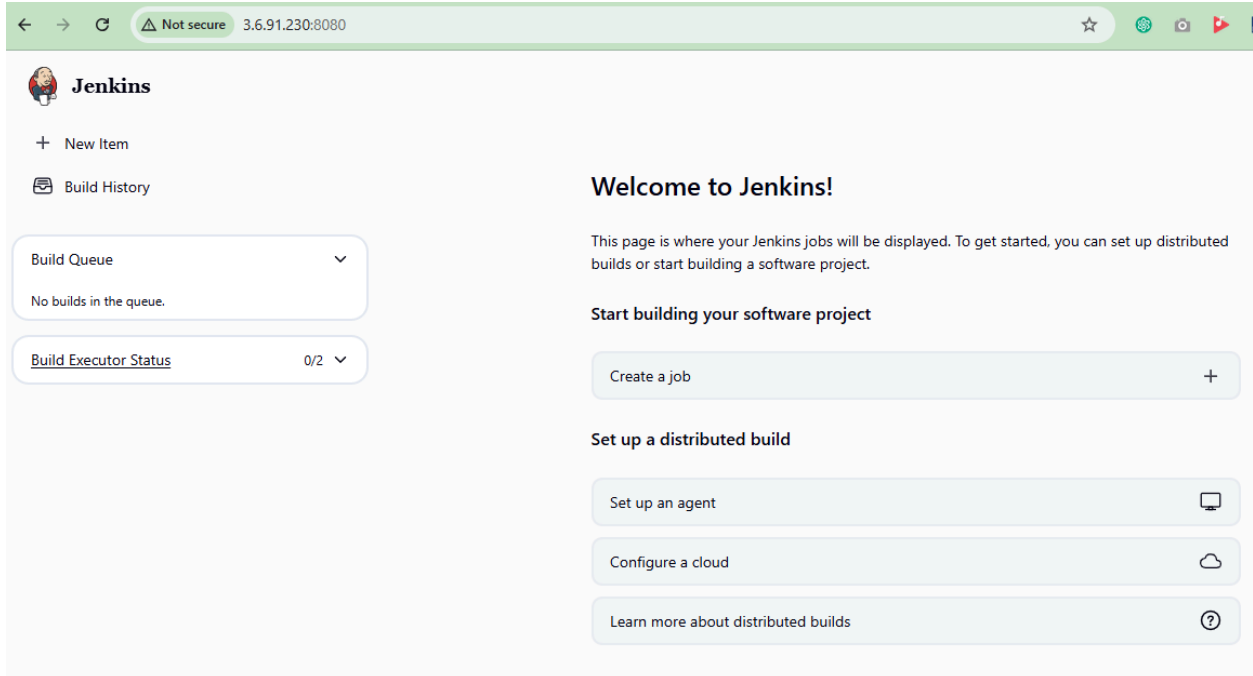
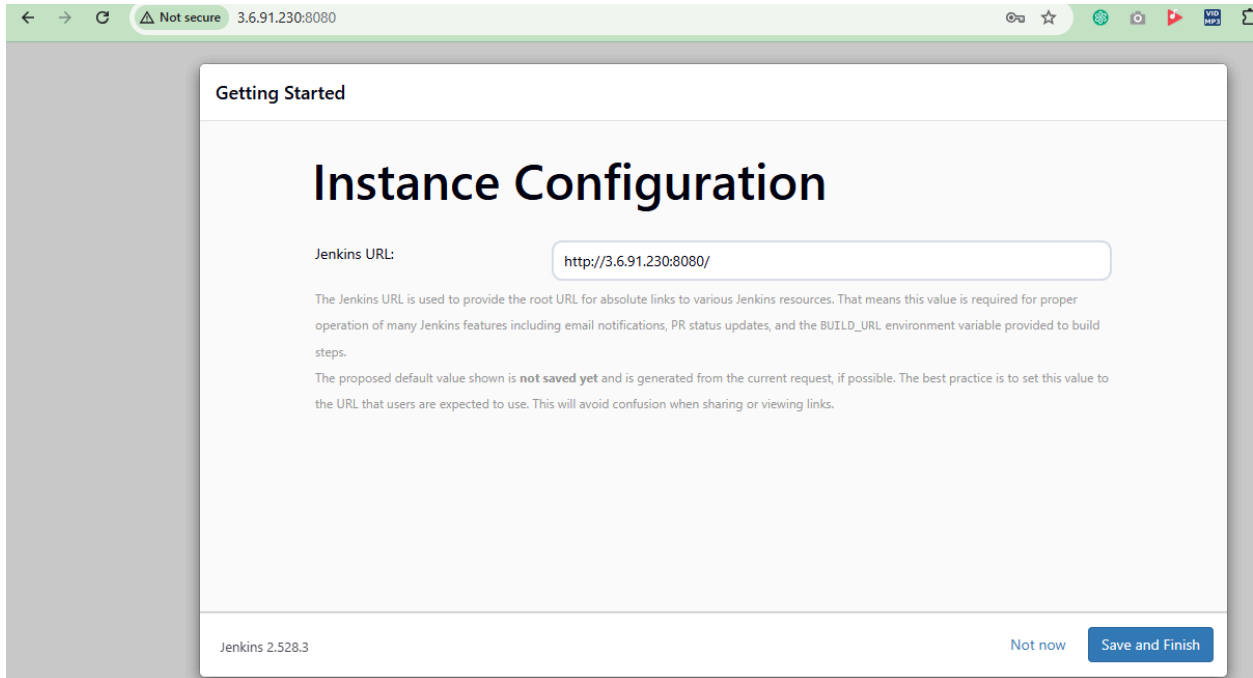
aws [Alt+S] Asia Pacific (Mumbai) Ask Amazon Q

```

ubuntu@ip-10-0-101-40:~$ docker --version
Docker version 29.1.5, build 0e6fee6
ubuntu@ip-10-0-101-40:~$ git --version
git version 2.34.1
ubuntu@ip-10-0-101-40:~$ aws --version
aws-cli/2.33.2 Python/3.13.11 Linux/6.8.0-1044-aws exe/x86_64.ubuntu.22
ubuntu@ip-10-0-101-40:~$ eksctl version
0.221.0
ubuntu@ip-10-0-101-40:~$ jenkins --version
2.528.3
ubuntu@ip-10-0-101-40:~$ kubectl version
Client Version: v1.35.0
Kustomize Version: v5.7.1
Error from server (Forbidden): <html><head><meta http-equiv='refresh' content='1;url=/login?from=%2Fversion%3Ftimeout%3D30&ta-redirect-url=/login?from=%2Fversion%3Ftimeout%3D32s' src='/static/9194c520/scripts/redirect.js'></script></head><body><div>Authentication required</div></body></html>
ubuntu@ip-10-0-101-40:~$
  
```

Jenkins installed





Stage : Kubernetes

EKS Kubernetes cluster created using Terraform

```
ubuntu@ip-172-31-28-193: ~/projects/eks_test$
module.eks.aws_eks_addon.before_compute["vpc-cni"]: Creation complete after 14s [id=my-cluster:vpc-cni]
module.eks.time.sleep.this[0]: Still creating... [20s elapsed]
module.eks.aws_eks_addon.before_compute["eks-pod-identity-agent"]: Still creating... [20s elapsed]
module.eks.aws_eks_addon.before_compute["eks-pod-identity-agent"]: Creation complete after 24s [id=my-cluster:eks-pod-identity-agent]
module.eks.time.sleep.this[0]: Still creating... [30s elapsed]
module.eks.time.sleep.this[0]: Creation complete after 30s [id=2026-01-23T06:41:07Z]
module.eks.module.eks_managed_node_group["default"].module.user_data.null_resource.validate_cluster_service_cidr: Creating...
module.eks.module.eks_managed_node_group["default"].module.user_data.null_resource.validate_cluster_service_cidr: Creation complete after 0s [id=818369290281045381]
module.eks.module.eks_managed_node_group["default"].aws_launch_template.this[0]: Creating...
module.eks.module.eks_managed_node_group["default"].aws_launch_template.this[0]: Creation complete after 6s [id=lt-097e77f2b21f536e]
module.eks.module.eks_managed_node_group["default"].aws_eks_node_group.this[0]: Creating...
module.eks.module.eks_managed_node_group["default"].aws_eks_node_group.this[0]: Still creating... [10s elapsed]
module.eks.module.eks_managed_node_group["default"].aws_eks_node_group.this[0]: Still creating... [20s elapsed]
module.eks.module.eks_managed_node_group["default"].aws_eks_node_group.this[0]: Still creating... [30s elapsed]
module.eks.module.eks_managed_node_group["default"].aws_eks_node_group.this[0]: Still creating... [40s elapsed]
module.eks.module.eks_managed_node_group["default"].aws_eks_node_group.this[0]: Still creating... [50s elapsed]
module.eks.module.eks_managed_node_group["default"].aws_eks_node_group.this[0]: Still creating... [1m0s elapsed]
module.eks.module.eks_managed_node_group["default"].aws_eks_node_group.this[0]: Still creating... [1m10s elapsed]
module.eks.module.eks_managed_node_group["default"].aws_eks_node_group.this[0]: Still creating... [1m20s elapsed]
module.eks.module.eks_managed_node_group["default"].aws_eks_node_group.this[0]: Still creating... [1m30s elapsed]
module.eks.module.eks_managed_node_group["default"].aws_eks_node_group.this[0]: Still creating... [1m40s elapsed]
module.eks.module.eks_managed_node_group["default"].aws_eks_node_group.this[0]: Creation complete after 1m48s [id=my-cluster:default]
026012306411275010000000d]
module.eks.aws_eks_addon.this["coredns"]: Creating...
module.eks.aws_eks_addon.this["kube-proxy"]: Creating...
module.eks.aws_eks_addon.this["coredns"]: Still creating... [10s elapsed]
module.eks.aws_eks_addon.this["kube-proxy"]: Still creating... [10s elapsed]
module.eks.aws_eks_addon.this["coredns"]: Creation complete after 14s [id=my-cluster:coredns]
module.eks.aws_eks_addon.this["kube-proxy"]: Still creating... [20s elapsed]
module.eks.aws_eks_addon.this["kube-proxy"]: Creation complete after 24s [id=my-cluster:kube-proxy]

Apply complete! Resources: 59 added, 0 changed, 0 destroyed.
ubuntu@ip-172-31-28-193:~/projects/eks_test$ eksctl get cluster
NAME      REGION    EKSCTL_CREATED
my-cluster  ap-south-1  False
```

Kubernetes Cluster details

Kubernetes Node details

Kubernetes Pod details

```
ubuntu@ip-172-31-28-193: ~/projects/trend/trend-repo
```

```
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ eksctl get cluster
```

NAME	REGION	EKSCTL CREATED
my-cluster	ap-south-1	False

```
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ kubectl get nodes
```

NAME	STATUS	ROLES	AGE	VERSION
ip-10-0-0-71.ap-south-1.compute.internal	Ready	<none>	34m	v1.32.9-eks-ecaa3a6
ip-10-0-2-35.ap-south-1.compute.internal	Ready	<none>	34m	v1.32.9-eks-ecaa3a6

```
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ kubectl get pods
```

NAME	READY	STATUS	RESTARTS	AGE
trendstore-6dcf4bb887-fzcpt	1/1	Running	0	5m31s
trendstore-6dcf4bb887-jdjgm	1/1	Running	0	5m31s

```
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ kubectl get pods -A
```

NAMESPACE	NAME	READY	STATUS	RESTARTS	AGE
default	trendstore-6dcf4bb887-fzcpt	1/1	Running	0	5m36s
default	trendstore-6dcf4bb887-jdjgm	1/1	Running	0	5m36s
kube-system	aws-node-86m9l	2/2	Running	0	34m
kube-system	aws-node-sbjv9	2/2	Running	0	34m
kube-system	coredns-6f9885fdb6-6sc4q	1/1	Running	0	33m
kube-system	coredns-6f9885fdb6-bztf5	1/1	Running	0	33m
kube-system	eks-pod-identity-agent-5z6xh	1/1	Running	0	34m
kube-system	eks-pod-identity-agent-9ddpl	1/1	Running	0	34m
kube-system	kube-proxy-stx2p	1/1	Running	0	33m
kube-system	kube-proxy-z9w9c	1/1	Running	0	33m

```

ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ ll
total 36
drwxrwxr-x 5 ubuntu ubuntu 4096 Jan 23 07:10 ./
drwxrwxr-x 3 ubuntu ubuntu 4096 Jan 21 08:51 ../
drwxrwxr-x 8 ubuntu ubuntu 4096 Jan 20 16:13 .git/
-rw-rw-r-- 1 ubuntu ubuntu 216 Jan 20 10:43 .gitignore
-rw-rw-r-- 1 ubuntu ubuntu 221 Jan 20 06:23 Dockerfile
drwxrwxr-x 3 ubuntu ubuntu 4096 Jan 23 07:14 Terraform/
-rw-rw-r-- 1 ubuntu ubuntu 481 Jan 23 07:07 deployment.yaml
drwxrwxr-x 3 ubuntu ubuntu 4096 Jan 20 05:54 dist/
-rw-rw-r-- 1 ubuntu ubuntu 417 Jan 23 07:10 service.yaml
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$

```

```

ubuntu@ip-172-31-28-193: ~/projects/trend/trend-repo
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ ll
total 36
drwxrwxr-x 5 ubuntu ubuntu 4096 Jan 23 07:10 ./
drwxrwxr-x 3 ubuntu ubuntu 4096 Jan 21 08:51 ../
drwxrwxr-x 8 ubuntu ubuntu 4096 Jan 20 16:13 .git/
-rw-rw-r-- 1 ubuntu ubuntu 216 Jan 20 10:43 .gitignore
-rw-rw-r-- 1 ubuntu ubuntu 221 Jan 20 06:23 Dockerfile
drwxrwxr-x 3 ubuntu ubuntu 4096 Jan 23 07:14 Terraform/
-rw-rw-r-- 1 ubuntu ubuntu 481 Jan 23 07:07 deployment.yaml
drwxrwxr-x 3 ubuntu ubuntu 4096 Jan 20 05:54 dist/
-rw-rw-r-- 1 ubuntu ubuntu 417 Jan 23 07:10 service.yaml
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ kubectl get svc

```

NAME	AGE	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)
kubernetes	38m	ClusterIP	172.20.0.1	<none>	443/TCP
trendstore	2/TCP	LoadBalancer	172.20.128.99	a8392e2ba0a6844878839c8fa2307040-b2ec893399960f82.elb.ap-south-1.amazonaws.com	3000:3115

```

ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$

```

Kubernetes deployment

Kubernetes deployment and service created

Secrets added for Docker hub private image pull request

kubectl create secret generic dockerhub-secret

--from-file=.dockerconfigjson=\$HOME/.docker/config.json

--type=kubernetes.io/dockerconfigjson

```

ubuntu@ip-172-31-28-193:~/projects/eks_test$ kubectl create secret generic dockerhub-secret --from-file=.dockerconfigjson=$HOME/.docker/config.json --type=kubernetes.io/dockerconfigjson
secret/dockerhub-secret created
ubuntu@ip-172-31-28-193:~/projects/eks_test$ kubectl get secrets

```

NAME	TYPE	DATA	AGE
dockerhub-secret	kubernetes.io/dockerconfigjson	1	3m1s

```

ubuntu@ip-172-31-28-193:~/projects/eks_test$

```

Application deployed using Deployment.yaml & Service.yaml

```
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ more deployment.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: trendstore
  labels:
    app: trendstore
spec:
  replicas: 2
  selector:
    matchLabels:
      app: trendstore
  template:
    metadata:
      labels:
        app: trendstore
    spec:
      imagePullSecrets:
        - name: dockerhub-secret
      containers:
        - name: trendstore
          image: udhayakumarethiraj/trendstore:latest
          imagePullPolicy: Always
          ports:
            - containerPort: 3000
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$
```

```
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ more service.yaml
apiVersion: v1
kind: Service
metadata:
  name: trendstore
  annotations:
    service.beta.kubernetes.io/aws-load-balancer-type: "classic"
    service.beta.kubernetes.io/aws-load-balancer-scheme: "internet-facing"
spec:
  type: LoadBalancer
  selector:
    app: trendstore
  ports:
    - port: 3000          # external port (what clients use)
      targetPort: 3000    # container port inside the pod (updated)
      protocol: TCP
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$
```

Stage : Jenkins

Jenkins setup

A Jenkinsfile created to define the CI/CD pipeline stages, including checkout, build, push, and deployment.

```

ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ ll
total 40
drwxrwxr-x 5 ubuntu ubuntu 4096 Jan 23 10:00 ./
drwxrwxr-x 3 ubuntu ubuntu 4096 Jan 23 07:28 ../
drwxrwxr-x 8 ubuntu ubuntu 4096 Jan 23 10:01 .git/
-rw-rw-r-- 1 ubuntu ubuntu 216 Jan 20 10:43 .gitignore
-rw-rw-r-- 1 ubuntu ubuntu 221 Jan 20 06:23 Dockerfile
-rw-rw-r-- 1 ubuntu ubuntu 2184 Jan 23 10:00 Jenkinsfile
drwxrwxr-x 3 ubuntu ubuntu 4096 Jan 23 07:14 Terraform/
-rw-rw-r-- 1 ubuntu ubuntu 481 Jan 23 07:07 deployment.yaml
drwxrwxr-x 3 ubuntu ubuntu 4096 Jan 20 05:54 dist/
-rw-rw-r-- 1 ubuntu ubuntu 417 Jan 23 07:10 service.yaml
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ 

```

Jenkins CI/CD Pipeline

Trendstore pipeline project created

The screenshot shows the Jenkins web interface for a project named 'trendstore-project'. The browser address bar indicates the URL is '65.0.120.218:8080/job/trendstore-project/'. The Jenkins logo and 'trendstore-project' are at the top. On the left, a sidebar contains links for 'Status', 'Changes', 'Build Now', 'Configure', 'Delete Pipeline', 'Stages', 'Rename', and 'Pipeline Syntax'. The main area shows the project status as 'trendstore-project' with a green checkmark. Below this, there's a 'Permalinks' section with a list of build links: 'Last build (#13), 2 min 35 sec ago', 'Last stable build (#13), 2 min 35 sec ago', 'Last successful build (#13), 2 min 35 sec ago', 'Last failed build (#7), 1 hr 27 min ago', 'Last unsuccessful build (#7), 1 hr 27 min ago', and 'Last completed build (#13), 2 min 35 sec ago'. At the bottom left, a 'Builds' panel shows a list of recent builds: '#13 10:02 am', '#12 10:00 am', and '#11 9:47 am', each with a green status icon. The bottom of the screen shows a Windows taskbar with various icons and system information like '27°C Mostly cloudy' and '15:36 23-01-2026'.

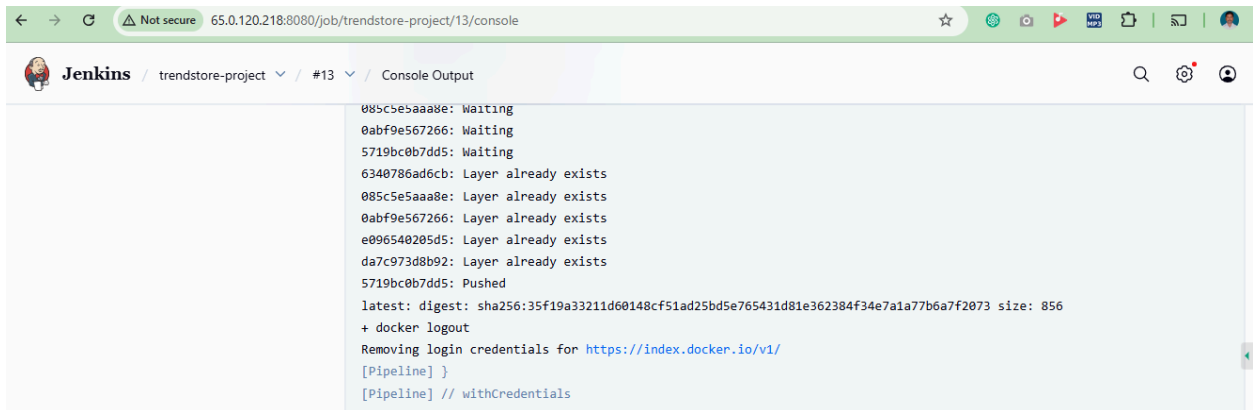
Jenkins Console Output : [URL](#)

Jenkins Build verification

Implemented a CI/CD pipeline using Jenkins to build Docker images, push them to Docker Hub, and deploy them to Kubernetes. The Docker Image ID was validated to ensure the same image was consistently used across the Jenkins build, Docker Hub registry, and Kubernetes deployment stages.

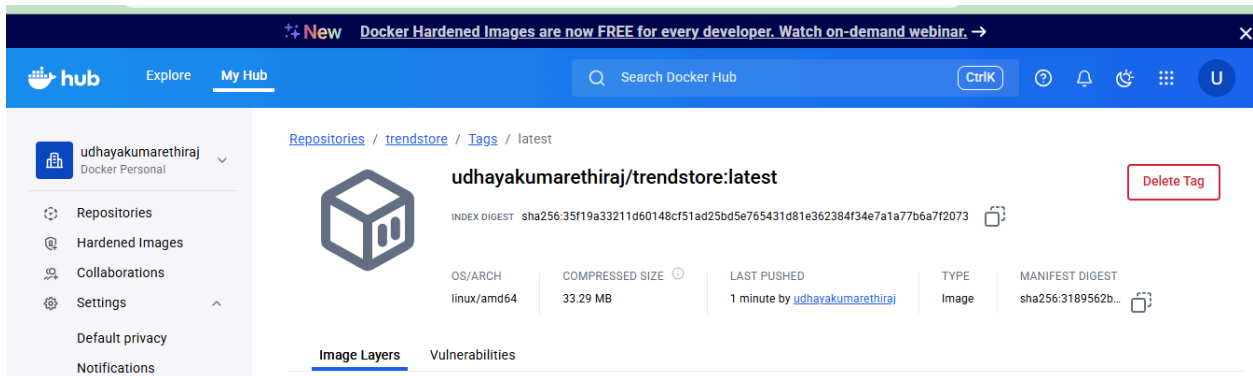
Jenkins build Docker Image pushed details from Jenkins log

Docker image Console output log



```
085c5e5aaa8e: Waiting
0abf9e567266: Waiting
5719bc0b7dd5: Waiting
6340786ad6cb: Layer already exists
085c5e5aaa8e: Layer already exists
0abf9e567266: Layer already exists
e096540205d5: Layer already exists
da7c973d8b92: Layer already exists
5719bc0b7dd5: Pushed
latest: digest: sha256:35f19a33211d60148cf51ad25bd5e765431d81e362384f34e7a1a77b6a7f2073 size: 856
+ docker logout
Removing login credentials for https://index.docker.io/v1/
[Pipeline] }
[Pipeline] // withCredentials
[Pipeline] }
```

Pushed Image details in Docker Hub



udhayakumarethiraj/trendstore:latest Delete Tag

INDEX DIGEST sha256:35f19a33211d60148cf51ad25bd5e765431d81e362384f34e7a1a77b6a7f2073

OS/ARCH	COMRESSED SIZE	LAST PUSHED	TYPE	MANIFEST DIGEST
linux/amd64	33.29 MB	1 minute by udhayakumarethiraj	Image	sha256:3189562b...

[Image Layers](#) [Vulnerabilities](#)

The same Image deployed in Kubernetes

pod image details

```
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ kubectl get pods -A
NAMESPACE      NAME                                     READY   STATUS    RESTARTS   AGE
default         trendstore-868b788886-fh5jf           1/1     Running   0           31s
default         trendstore-868b788886-qlfvh           1/1     Running   0           38s
kube-system     aws-node-86m9l                         2/2     Running   0           3h20m
kube-system     aws-node-sbjv9                         2/2     Running   0           3h20m
kube-system     coredns-6f9885fdb6-6sc4q              1/1     Running   0           3h20m
kube-system     coredns-6f9885fdb6-bztf5              1/1     Running   0           3h20m
kube-system     eks-pod-identity-agent-5z6xh           1/1     Running   0           3h20m
kube-system     eks-pod-identity-agent-9ddpl           1/1     Running   0           3h20m
kube-system     kube-proxy-stx2p                       1/1     Running   0           3h20m
kube-system     kube-proxy-z9w9c                       1/1     Running   0           3h20m
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ kubectl describe pod trendstore-868b788886-fh5jf | grep -E "Image|Image ID"
    Image:      udhayakumarethiraj/trendstore:latest
    Image ID:   docker.io/udhayakumarethiraj/trendstore@sha256:35f19a33211d60148cf51ad25bd5e765431d81e362384f34e7a1a77b6a7f2073
    Normal Pulled    50s   kubelet          Successfully pulled image "udhayakumarethiraj/trendstore:latest" in 8.317s (8.317s including waiting). Image size: 34919963 bytes.
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ ls
```

Application output :

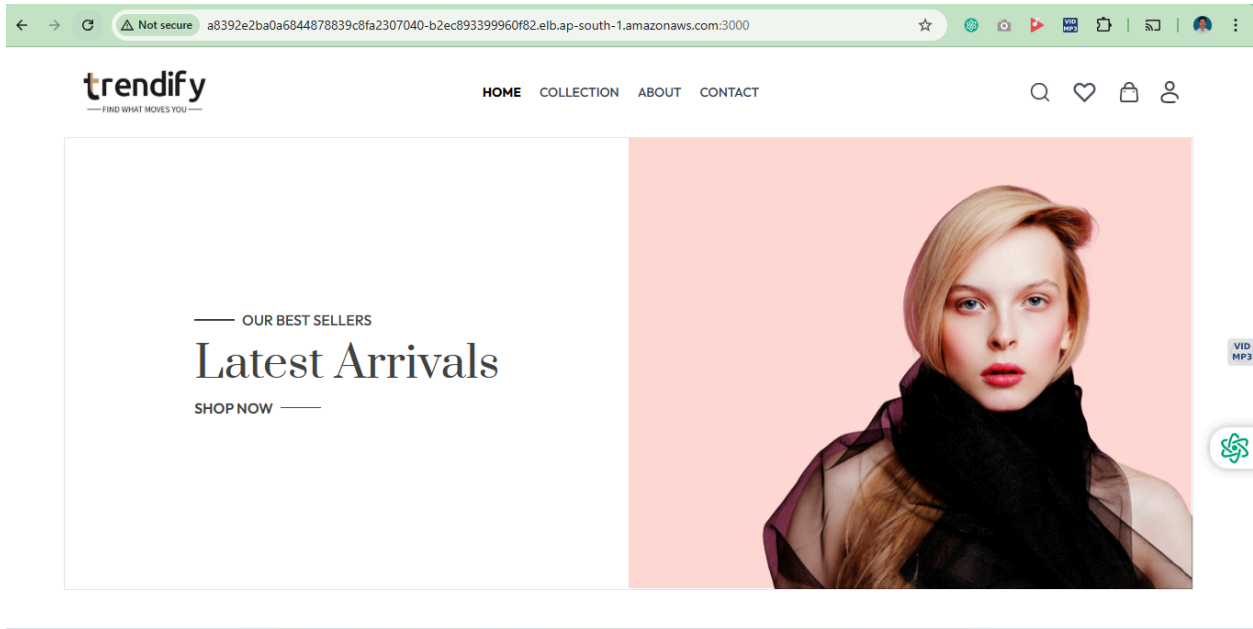
<http://a8392e2ba0a6844878839c8fa2307040-b2ec893399960f82.elb.ap-south-1.amazonaws.com:3000/>

Load balancer ARN

arn:aws:elasticloadbalancing:ap-south-1:758234806674:loadbalancer/net/a8392e2ba0a6844878839c8fa2307040/b2ec893399960f82

Load balancer DNS

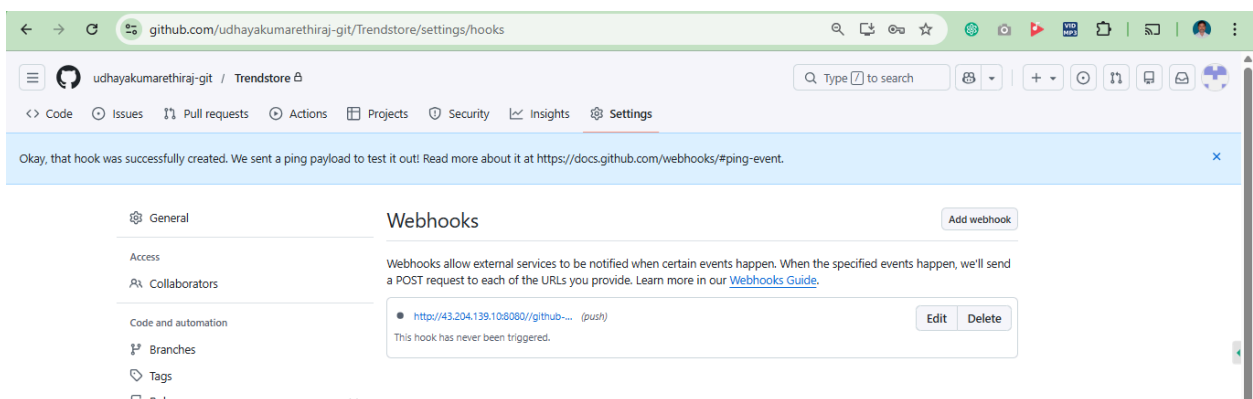
a8392e2ba0a6844878839c8fa2307040-b2ec893399960f82.elb.ap-south-1.amazonaws.com



GitHub and Jenkins Integration

Integrated GitHub with Jenkins using webhooks to automatically trigger builds on every commit, enabling continuous integration and eliminating manual build execution.

Webhook added for build auto trigger



Github webhook Auto Triggered

The screenshot shows the GitHub web interface for the repository 'udhayakumarethiraj-git / Trendstore'. The 'Settings' tab is selected, and the 'Webhooks / Manage webhook' section is active. A recent delivery is listed with a status of 'Completed' and a response code of '200'. The 'Request' tab is selected, displaying the following headers:

```
Request URL: http://43.204.139.10:8080/github-webhook/  
Request method: POST  
Accept: */*  
Content-Type: application/json  
User-Agent: GitHub-Hookshot/9b5ad09  
X-GitHub-Delivery: 953eb384-fb42-11f8-9a4c-0d07306300a5  
X-GitHub-Event: push  
X-GitHub-Hook-ID: 593508993  
X-GitHub-Hook-Installation-Target-ID: 1138008365  
X-GitHub-Hook-Installation-Target-Type: repository
```

Jenkins Build config

The screenshot shows the Jenkins Configuration page for the 'trendstore-project' job. The 'Configure' tab is selected, and the 'Triggers' section is active. The 'GitHub hook trigger for GITScm polling' option is checked, indicating that the build is triggered by GitHub webhooks.

Configure

- General
- Triggers
- Pipeline
- Advanced

Triggers

Set up automated actions that start your build based on specific events, like code changes

- ☐ Throttle builds ?
- ☐ Build after other projects are built ?
- ☐ Build periodically ?
- ☒ GitHub hook trigger for GITScm polling ?
- ☐ Poll SCM ?
- ☐ Trigger builds remotely (e.g., from scripts) ?

Jenkins Build status

The screenshot shows the Jenkins web interface for a build named '#21' in the 'trendstore-project'. The build is successful, indicated by a green checkmark icon. The status bar shows the build number '#21' and the timestamp '(27-Jan-2026, 5:39:57 am)'. On the left, there is a sidebar with various links: Status, Changes, Console Output, Edit Build Information, Delete build '#21', Polling Log, Timings, Git Build Data, Pipeline Overview, Restart from Stage, Replay, Pipeline Steps, and Workspaces. The main content area displays the build details. It starts with a green checkmark icon and the text 'Started by GitHub push by udhayakumarethiraj-git'. Below this, it shows the build's timing: 'This run spent: 6.3 sec waiting; 28 sec build duration; 35 sec total from scheduled to completion.' The build is associated with a Git repository: 'Revision: a7969a8141f98de770b1cddd83356419694a0717' and 'Repository: git@github.com:udhayakumarethiraj-git/Trendstore.git'. A warning icon indicates that some steps may have insecure interpolation of sensitive variables, with a link to 'click here for an explanation'. The build steps are listed under the 'Changes' section, showing '1. Add files via upload (details / githubweb)'.

Stage : Monitoring

An open-source monitoring system was implemented using Prometheus to collect cluster and application metrics, and Grafana was configured with dashboards to monitor application health, application URL status, and real-time metrics data.

Install Node Exporter

```
helm repo add prometheus-community https://prometheus-community.github.io/helm-charts
helm repo update
```

```
helm install node-exporter prometheus-community/prometheus-node-exporter \
--namespace monitoring
```

Install Kube Metrics

```
helm install kube-state-metrics prometheus-community/kube-state-metrics \
--namespace monitoring
```

Verify

```
kubectl get pods -n monitoring
```

Expose it via LoadBalancer

Node Exporter LB

```
kubectl expose daemonset node-exporter-prometheus-node-exporter \
  -n monitoring \
  --name node-exporter-lb \
  --type LoadBalancer \
  --port 9100 \
  --target-port 9100
```

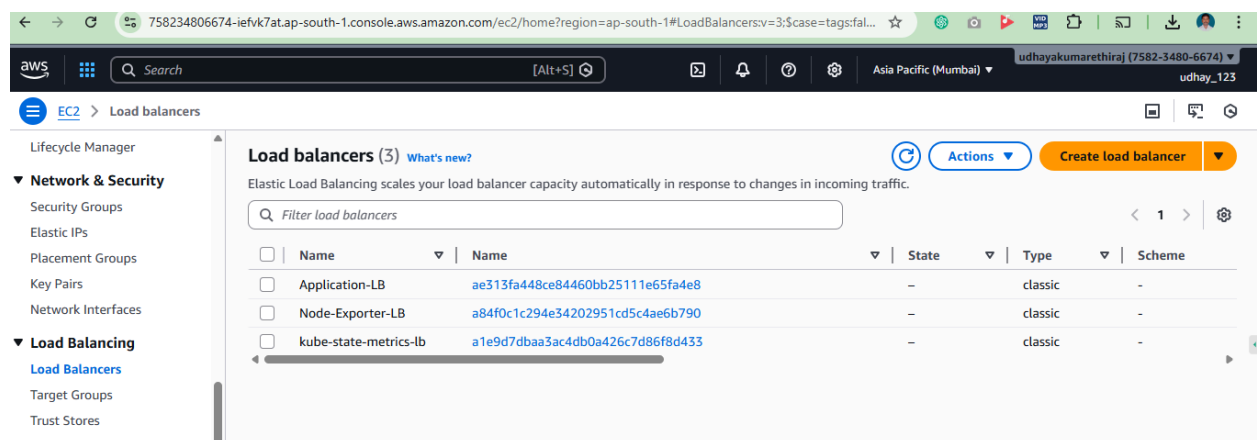
Kube State Metrics LB

```
kubectl expose deployment kube-state-metrics \
  -n monitoring \
  --name kube-state-metrics-lb \
  --type LoadBalancer \
  --port 8080 \
  --target-port 8080
```

Verify

```
kubectl get svc -n monitoring
```


Kubernetes metrics forwarded to Prometheus Server via loadbalancers



Prometheus Data from Kubernetes

Targets added to Prometheus

Query response


Prometheus
Alerts
Graph
Status
Help

☐ Use local time
 ☐ Enable query history
 ☒ Enable autocomplete
 ☒ Enable highlighting
 ☒ Enable linter

Execute

Table
Graph
Load time: 69ms
Resolution: 14s
Result series: 1

<
Evaluation time
>

```
probe_http_status_code(instance="http://ae313fa448ce84460bb25111e65fa4e8-228414783.ap-south-1.elb.amazonaws.com:3000", job="trendstore-app")
```

200

Remove Panel

Prometheus Alerts Graph Status Help

☐ Use local time ☐ Enable query history ☒ Enable autocomplete ☒ Enable highlighting ☒ Enable linter

Q probe_duration_seconds{job="trendstore-app"} Execute

Table Graph

Load time: 66ms Resolution: 14s Result series: 1

< Evaluation time >

probe_duration_seconds(instance="http://ae313fa448ce84460bb25111e65fa4e8-228414783.ap-south-1.elb.amazonaws.com:3000", job="trendstore-app")	0.005879886
--	-------------

```
ubuntu@ip-172-31-28-193:~/projects/trend/trend-repo$ kubectl get svc monitoring-kube-prometheus-prometheus -n monitoring -w
NAME                                TYPE        AGE      CLUSTER-IP      EXTERNAL-IP
monitoring-kube-prometheus-prometheus LoadBalancer 172.20.243.2   aaa4183ae50434bda8288b2d02cc0ac3-1358863498.ap-south-1.elb.amazonaws.com 9090:32226/TCP,8080:31249/TCP 112m
```

← → ↺ ⚠ Not secure 3.6.205.231:9090/graph?g0.expr=node_cpu_seconds_total&g0.tab=1&g0.stacked=0&g0.show_exemplars=0&g0.range_in... ☆

Prometheus Alerts Graph Status Help

☐ Use local time ☐ Enable query history ☒ Enable autocomplete ☒ Enable highlighting ☒ Enable linter

Q node_cpu_seconds_total

Table Graph

Load time: 70ms Resolution: 14s Result series: 1

< Evaluation time >

node_cpu_seconds_total(container="node-exporter", cpu="0", endpoint="http-metrics", instance="10.0.0.71:9100", job="node-exporter", mode="idle", namespace="monitoring", pod="monitoring-prometheus-node-exporter-vcbnw", prometheus="monitoring/monitoring-kube-prometheus-prometheus", prometheus_replica="prometheus-monitoring-kube-prometheus-prometheus-0", service="monitoring-prometheus-node-exporter")	20.000000000
node_cpu_seconds_total(container="node-exporter", cpu="0", endpoint="http-metrics", instance="10.0.0.71:9100", job="node-exporter", mode="iowait", namespace="monitoring", pod="monitoring-prometheus-node-exporter-vcbnw", prometheus="monitoring/monitoring-kube-prometheus-prometheus", prometheus_replica="prometheus-monitoring-kube-prometheus-prometheus-0", service="monitoring-prometheus-node-exporter")	4.000000000
node_cpu_seconds_total(container="node-exporter", cpu="0", endpoint="http-metrics", instance="10.0.0.71:9100", job="node-exporter", mode="irq", namespace="monitoring", pod="monitoring-prometheus-node-exporter-vcbnw", prometheus="monitoring/monitoring-kube-prometheus-prometheus", prometheus_replica="prometheus-monitoring-kube-prometheus-prometheus-0", service="monitoring-prometheus-node-exporter")	0.000000000
node_cpu_seconds_total(container="node-exporter", cpu="0", endpoint="http-metrics", instance="10.0.0.71:9100", job="node-exporter", mode="nice", namespace="monitoring", pod="monitoring-prometheus-node-exporter-vcbnw", prometheus="monitoring/monitoring-kube-prometheus-prometheus", prometheus_replica="prometheus-monitoring-kube-prometheus-prometheus-0", service="monitoring-prometheus-node-exporter")	0.000000000
node_cpu_seconds_total(container="node-exporter", cpu="0", endpoint="http-metrics", instance="10.0.0.71:9100", job="node-exporter", mode="softirq", namespace="monitoring", pod="monitoring-prometheus-node-exporter-vcbnw", prometheus="monitoring/monitoring-kube-prometheus-prometheus", prometheus_replica="prometheus-monitoring-kube-prometheus-prometheus-0", service="monitoring-prometheus-node-exporter")	9.000000000
node_cpu_seconds_total(container="node-exporter", cpu="0", endpoint="http-metrics", instance="10.0.0.71:9100", job="node-exporter", mode="steal", namespace="monitoring", pod="monitoring-prometheus-node-exporter-vcbnw", prometheus="monitoring/monitoring-kube-prometheus-prometheus", prometheus_replica="prometheus-monitoring-kube-prometheus-prometheus-0", service="monitoring-prometheus-node-exporter")	1.000000000

Added Kubernetes Data source to Grafana

← → ↺ ⚠ Not secure 3.6.205.231:3000/connections/datasources/edit/dfaaqv5n3pcf ☆

Grafana Home > Connections > Data sources > prometheus

Traces Profiles Alerting Alert rules Contact points Notification policies Silences Active notifications Recently deleted Settings

Connections Add new connection Data sources Administration General Plugins and data

Type: Prometheus

Settings Dashboards Permissions Insights Cache

Name prometheus Default

Before you can use the Prometheus data source, you must configure it below or in the config file. For detailed instructions, [view the documentation](#).

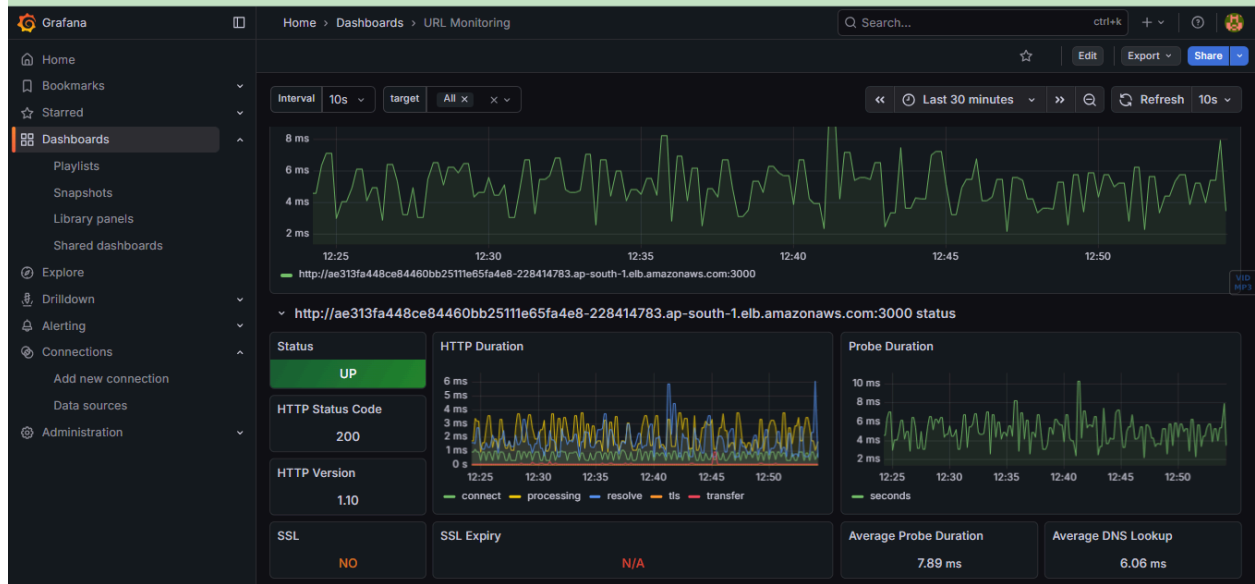
Fields marked with * are required

Connection

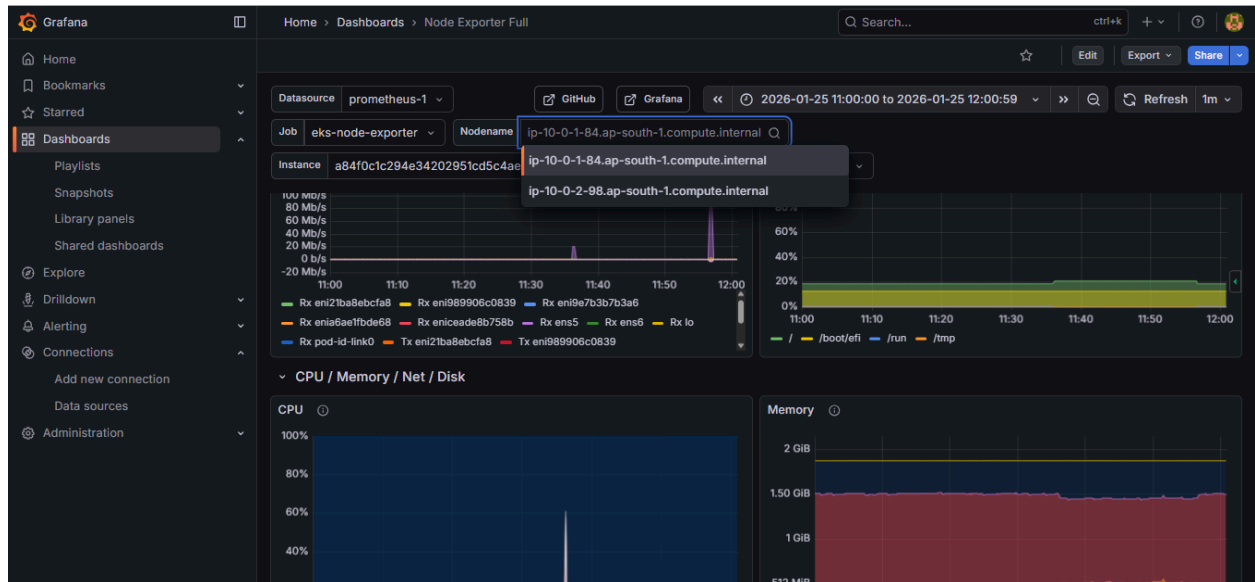
Prometheus server URL * http://aaa4183ae50434bda8288b2d02cc0ac3-1358863498.ap-south-1.elb.amazonaws.com:9090

Grafana Dashboards :

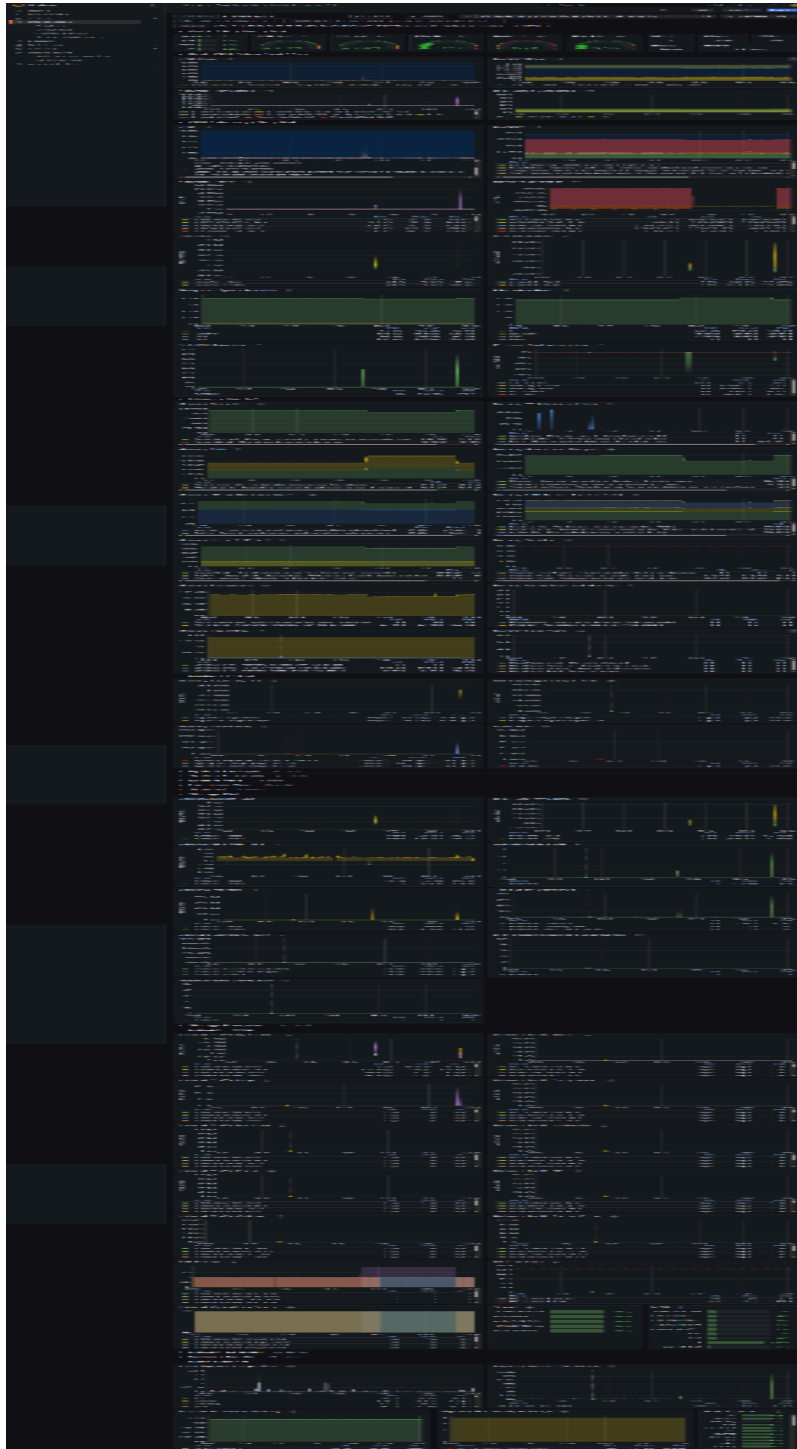
Application URL



Node Monitoring Data



Node Monitoring full page data



Cluster Monitoring Data

Grafana Home Dashboards > Kubernetes Cluster

Search... ctrl+k + - 🔍

Add ▾ Settings Exit edit Save dashboard

datasource prometheus-1 **cluster** **resolution** 30s ⏪ 🕒 2026-01-25 11:00:00 to 2026-01-25 12:00:59 ⏩ 🔍 🔄 Refresh 30s ▾

job eks-node-exporter ✕ ✕ **terraform** 🔵 **oncall** 🔵

Overview

Global CPU Usage
Real
1.53%

Global RAM Usage
Real
29.87%

Nodes
2
Nam...
5
Runn...
13

Kubernetes Resource Count

Name	Min	Max
Running Containers	15	15
Running Pods	13	13
Configmaps	12	12
Services	9	9
Secrets	3	3
Nodes	2	2

CPU Usage
Real 0.0207
Real 0
Average 0.502
Real 0
Real 0

RAM Usage
Real 572 MiB
Real 0
Average 140 MiB
Real 0
Limit 340 MiB
Real 0

Resources

Cluster CPU Utilization
150.00%
125.00%
100.00%
75.00%
50.00%
25.00%
0.00%
11:00 11:10 11:20 11:30 11:40 11:50 12:00

Cluster Memory Utilization
34%
33%
32%
31%
30%
29%
11:00 11:10 11:20 11:30 11:40 11:50 12:00

CPU Utilization by namespace
No data

Memory Utilization by namespace
No data

CPU Utilization by instance
150.00%
125.00%
100.00%
75.00%
50.00%
25.00%
0.00%
11:00 11:30 12:00
Name
a84f0c1c294e34202951cd5c4ae6b/90-1482954724

Memory Utilization by instance
656 MiB
640 MiB
624 MiB
608 MiB
592 MiB
576 MiB
560 MiB
11:00 11:30 12:00
Name
a84f0c1c294e34202951cd5c4ae6b/90-1482954724

CPU Throttled seconds by namespace
No data

CPU Core Throttled by Instance
No data

Kubernetes

Kubernetes Pods QoS classes

Name	Min	Max	Mean
Total pods	13	13	13
BestEffort pods	7	7	7
Burstable pods	6	6	6
Guaranteed pods	0	0	0

Kubernetes Pods Status Reason

Name	Min	Max
Evicted	0	0
NodeAffinity	0	0
NodeLost	0	0
Shutdown	0	0
UnexpectedAdmissionError	0	0

