

ENPM818N Final Project

EKS and Monitoring with OpenTelemetry



University of Maryland

7950 Baltimore Avenue

College Park, MD 20742

10-29-2024

Phase 1: Environment and Initial Application Setup

Sreelekshmy Rengith

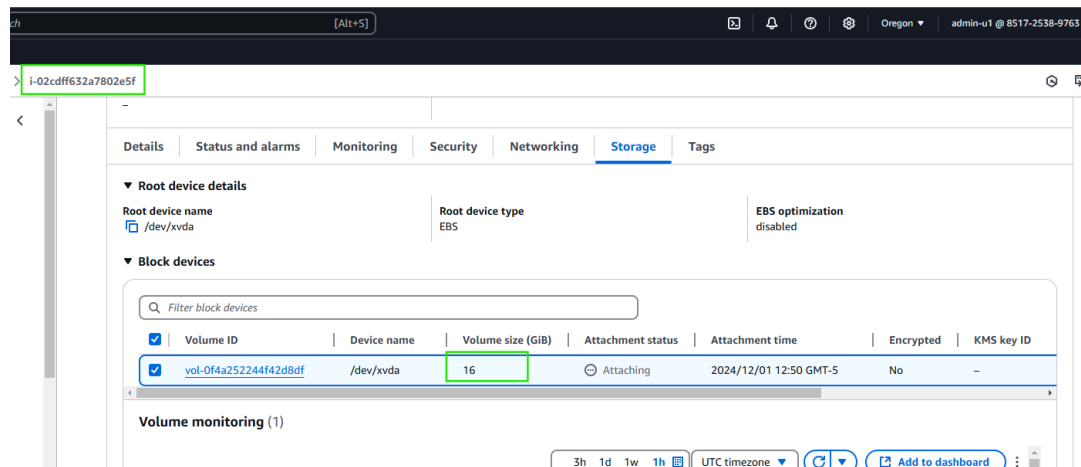
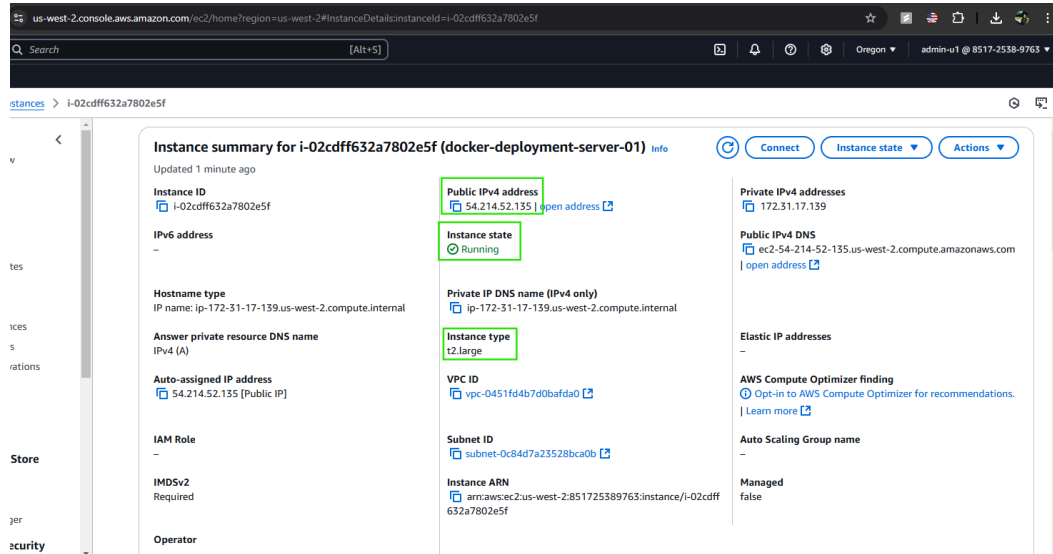
UID: 119458815

Phase 1 : Environment and Initial Application Setup

1. Docker Deployment on EC2:

Create an EC2 instance and connect securely using SSH and your private key.

Connecting to the EC2 instance vi SSH and installing Docker inside that instance:



Installing Docker on the EC2 Instance

To set up Docker on the EC2 instance, we followed these steps:

1. Update System Packages:

Ensures all existing packages are up-to-date.

2. Install Docker:

Installs Docker to enable containerization.

3. Start Docker Service:

Initiates the Docker service to allow running containers.

4. Enable Docker on Boot:

Configures Docker to automatically start on system reboot.

```
C:\Windows\system32\cmd.exe X ec2-user@ip-172-31-17-139:~ X Windows PowerShell X + X
```

```
PS C:\Users\rengi\Downloads> ssh -i "docker-deployment-keypair.pem" ec2-user@54.214.52.135
The authenticity of host '54.214.52.135 (54.214.52.135)' can't be established.
ED25519 key fingerprint is SHA256:8Wf8nMXsr0Lwadr9pG82woxd3H/felanhLiCvUdEdg.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '54.214.52.135' (ED25519) to the list of known hosts.
```

```
#
_#_##### Amazon Linux 2
###\#####
###\###| AL2 End of Life is 2025-06-30.
###\#/
#####V__t-->
#####
#####_/_/_/ Amazon Linux 2023, GA and supported until 2028-03-15.
_m/' n/ https://aws.amazon.com/linux/amazon-linux-2023/
```

```
[ec2-user@ip-172-31-17-139 ~]$ sudo yum update -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
No packages marked for update
[ec2-user@ip-172-31-17-139 ~]$ sudo yum install docker -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
--> Package docker.x86_64 0:25.0.6-1.amzn2.0.2 will be installed
--> Processing Dependency: containerd >= 1.3.2 for package: docker-25.0.6-1.amzn2.0.2.x86_64
--> Processing Dependency: libcgroupp >= 0.40.rc1-5.15 for package: docker-25.0.6-1.amzn2.0.2.x86_64
--> Processing Dependency: runc >= 1.0.0 for package: docker-25.0.6-1.amzn2.0.2.x86_64
--> Processing Dependency: pigz for package: docker-25.0.6-1.amzn2.0.2.x86_64
--> Running transaction check
--> Package containerd.x86_64 0:1.7.23-1.amzn2.0.2 will be installed
--> Package libcgroupp.x86_64 0:0.41-21.amzn2 will be installed
--> Package pigz.x86_64 0:2.3.4-1.amzn2.0.1 will be installed
--> Package runc.x86_64 0:1.1.14-1.amzn2 will be installed
--> Finished Dependency Resolution

Dependencies Resolved
```

Package	Arch	Version	Repository
---------	------	---------	------------

```
Complete!  
[ec2-user@ip-172-31-17-139 ~]$ sudo systemctl start docker  
[ec2-user@ip-172-31-17-139 ~]$ sudo systemctl enable docker  
Created symlink from /etc/systemd/system/multi-user.target.wants/docker.service to /usr/lib/systemd/system/docker.service.
```

3. Installing Docker Compose

Download and install the latest version of Docker Compose, then make it executable.

Installing Docker Compose:

```
sudo curl -L "https://github.com/docker/compose/releases/download/$(curl -s  
https://api.github.com/repos/docker/compose/releases/latest | grep tag name | cut -d '"' -f
```

4. Copying the `docker-compose.yml` File to the EC2 Instance

```
docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
sudo chmod +x /usr/local/bin/docker-compose
```

```
[ec2-user@ip-172-31-17-139 ~]$ sudo curl -L "https://github.com/docker/compose/releases/download/$(curl -s https://api.github.com/repos/docker/compose/releases/latest | grep tag_name | cut -d '"' -f 4)/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
0 0 0 0 0 0 0 0 --:--:-- --:--:-- --:--:-- 0
100 61.2M 100 61.2M 0 0 62.0M 0 --:--:-- --:--:-- --:--:-- 88.5M
[ec2-user@ip-172-31-17-139 ~]$ sudo chmod +x /usr/local/bin/docker-compose
[ec2-user@ip-172-31-17-139 ~]$ docker-compose --version
Docker Compose version v2.31.0
[ec2-user@ip-172-31-17-139 ~]$ pwd
/home/ec2-user
```

Copying the docker-compose.yml file from local to the EC2 instance:

```
PS C:\Users\rengi\Downloads> scp -i "docker-deployment-keypair.pem" "docker-compose.yml" ec2-user@54.214.52.135:~
docker-compose.yml
PS C:\Users\rengi\Downloads>
```

```
[ec2-user@ip-172-31-17-139 ~]$ cd ~
[ec2-user@ip-172-31-17-139 ~]$ ls
docker-compose.yml
[ec2-user@ip-172-31-17-139 ~]$
```

```
[ec2-user@ip-172-31-25-154 ~]$ ls
opentelemetry-demo
[ec2-user@ip-172-31-25-154 ~]$ cd opentelemetry-demo/
[ec2-user@ip-172-31-25-154 opentelemetry-demo]$ ls
buildkitd.toml  docker-compose.minimal.yml  docker-compose.yml  internal  Makefile  pb  src
CHANGELOG.md   docker-compose-tests_include-override.yml  docker-gen-proto.sh  kubernetes  package.json  README.md  test
CONTRIBUTING.md  docker-compose-tests.yml  ide-gen-proto.sh  LICENSE  package-lock.json  renovate.json5
```

5. Troubleshooting **Permission Denied** Error

- **Issue:**
Error when running `docker-compose up -d`.
- **Solution:**
Add your user to the Docker group and restart the session:

Reconnect to the instance and retry `docker-compose up -d`.

```
[ec2-user@ip-172-31-17-139 ~]$ sudo docker-compose up -d
unable to get image 'ghcr.io/open-feature/flagd:v0.11.4': permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Get "http://%2Fvar%2Frun%2Fdocker.sock/v1.47/images/gchc.io/open-feature/flagd:v0.11.4/json": dial unix /var/run/docker.sock: connect: permission denied
[ec2-user@ip-172-31-17-139 ~]$ sudo docker-compose up -d
sudo: docker-compose: command not found
[ec2-user@ip-172-31-17-139 ~]$ docker-compose --version
Docker Compose version v2.31.0
[ec2-user@ip-172-31-17-139 ~]$ sudo systemctl start docker
[ec2-user@ip-172-31-17-139 ~]$ sudo systemctl enable docker
[ec2-user@ip-172-31-17-139 ~]$ docker-compose up -d
unable to get image 'ghcr.io/open-feature/flagd:v0.11.4': permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Get "http://%2Fvar%2Frun%2Fdocker.sock/v1.47/images/gchc.io/open-feature/flagd:v0.11.4/json": dial unix /var/run/docker.sock: connect: permission denied
[ec2-user@ip-172-31-17-139 ~]$ grep docker /etc/group
docker:x:992:
[ec2-user@ip-172-31-17-139 ~]$ sudo usermod -ag docker ec2-user
[ec2-user@ip-172-31-17-139 ~]$ exit
logout
Connection to 54.214.52.135 closed.
PS C:\Users\rengid\Downloads> ssh -i "docker-deployment-keypair.pem" ec2-user@54.214.52.135
Last login: Sun Dec 1 18:38:33 2024 from 129.2.89.2

_#_
 _L##### Amazon Linux 2
### \####|
    \|###| AL2 End of Life is 2025-06-30.
     \|##|   V#/\--->
      \|_/--/_ A newer version of Amazon Linux is available!
       \|_|_\--/_ Amazon Linux 2023, GA and supported until 2028-03-15.
        \|m/'----/_ https://aws.amazon.com/linux/amazon-linux-2023/

[ec2-user@ip-172-31-17-139 ~]$ groups ec2-user
ec2-user : ec2-user adm wheel system journal docker
[ec2-user@ip-172-31-17-139 ~]$ docker-compose --version
Docker Compose version v2.31.0
[ec2-user@ip-172-31-17-139 ~]$ sudo systemctl start docker
[ec2-user@ip-172-31-17-139 ~]$ sudo systemctl enable docker
[ec2-user@ip-172-31-17-139 ~]$ docker-compose up -d
[*] Running 241/25
✔ kafka Pulled 74.9s
✔ adservice Pulled 83.5s
✔ jaeger Pulled 70.1s
```

6. Running Docker Containers

Using the command: `docker-compose up -d`

Description: Deploy your applications in detached mode using Docker Compose.

Docker deployment successful:

```
opentelemetry-demo
[ec2-user@ip-172-31-25-154 ~]$ cd opentelemetry-demo/
[ec2-user@ip-172-31-25-154 opentelemetry-demo]$ docker-compose up -d
[*] Running 242/344
✓shippingservice Pulled 15.6s
✓emailservice Pulled 18.7s
✓kafka Pulled 91.2s
✓productcatalogservice Pulled 5.8s
✓jaeger Pulled 65.9s
✓frontend Pulled 110.4s
✓opensearch Pulled 124.9s
✓prometheus Pulled 96.5s
✓imageprovider Pulled 41.2s
✓flagd Pulled 99.4s
✓otelcol Pulled 89.9s
✓valkey-cart Pulled 59.6s
✓loadgenerator Pulled 117.5s
✓grafana Pulled 100.6s
✓currencyservice Pulled 53.5s
✓flagd Pulled 32.4s
✓accountingservice Pulled 25.9s
✓frauddetectionservice Pulled 95.2s
✓adservice Pulled 90.6s
✓frontendproxy Pulled 40.4s
✓cartservice Pulled 51.9s
✓checkoutservice Pulled 40.1s
✓paymentservice Pulled 70.4s
✓recommendationservice Pulled 36.9s
✓quoteservice Pulled 97.1s
```

```
[*] Running 26/26
✓Network opentelemetry-demo Created 0.3s
✓Container valkey-cart Started 4.0s
✓Container jaeger Started 3.3s
✓Container prometheus Started 3.9s
✓Container opensearch Healthy 44.8s
✓Container grafana Started 4.1s
✓Container flagd Started 4.2s
✓Container kafka Healthy 49.1s
✓Container otel-col Started 43.3s
✓Container cart-service Started 46.8s
✓Container imageprovider Started 44.8s
✓Container flagd-ui Started 44.8s
✓Container currency-service Started 46.1s
✓Container email-service Started 44.3s
✓Container shipping-service Started 45.5s
✓Container accounting-service Started 45.1s
✓Container quote-service Started 46.4s
✓Container payment-service Started 46.8s
✓Container product-catalog-service Started 46.6s
✓Container ad-service Started 46.1s
✓Container frauddetection-service Started 46.5s
✓Container recommendation-service Started 47.1s
✓Container checkout-service Started 47.9s
✓Container frontend Started 48.5s
✓Container load-generator Started 49.1s
✓Container frontend-proxy Started 49.7s
[ec2-user@ip-172-31-25-154 opentelemetry-demo]$
```

Running applications:

```
2024-12-08T14:45:42.181Z info file/filepath_sync.go:101 watching filepath: ./etc/flagd/demo.flagd.json {"component": "sync", "sync": "fileinfo"}
2024-12-08T14:45:42.184Z info flag-evaluation/connect_service.go:227 Flag IResolver listening at [:]:8013 {"component": "service"}
2024-12-08T14:45:42.197Z info flag-evaluation/connect_service.go:247 metrics and probes listening at 8014 {"component": "service"}
[ec2-user@ip-172-31-24-124 opentelemetry-demo]$ docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
4df217d34e6f	ghcr.io/open-telemetry/demo:latest-frontendproxy	"/bin/sh -c 'envsubst..."	13 minutes ago	Up 13 minutes	0.0.0.0:8080->8080/tcp, :::8080-
91a00c5924fe	ghcr.io/open-telemetry/demo:latest-loadgenerator	"locust --skip-log-s..."	13 minutes ago	Up 13 minutes	0.0.0.0:32789->8089/tcp, :::3278
62c488fbcdd9	ghcr.io/open-telemetry/demo:latest-frontend	"npm start"	13 minutes ago	Up 13 minutes	0.0.0.0:32788->8080/tcp, :::3278
bfbcb0aa4c6e5	ghcr.io/open-telemetry/demo:latest-recommendationservice	"opentelemetry-instr..."	13 minutes ago	Up 13 minutes	0.0.0.0:32787->9001/tcp, :::3278
1b823ee20628	ghcr.io/open-telemetry/demo:latest-checkoutservice	"./checkoutservice"	13 minutes ago	Up 13 minutes	0.0.0.0:32786->5050/tcp, :::3278
d01452b365fd	ghcr.io/open-telemetry/demo:latest-shippingservice	"/app/shippingservice"	13 minutes ago	Up 13 minutes	0.0.0.0:32785->50050/tcp, :::327
49d5f8bd45c2	ghcr.io/open-telemetry/demo:latest-paymentservice	"npm run start"	13 minutes ago	Up 13 minutes	0.0.0.0:32781->50051/tcp, :::327
ec8431092842	ghcr.io/open-telemetry/demo:latest-cartservice	"./cartservice"	13 minutes ago	Up 13 minutes	0.0.0.0:32782->7070/tcp, :::3278
47f54416d569	ghcr.io/open-telemetry/demo:latest-productcatalogservice	"./productcatalogser..."	13 minutes ago	Up 13 minutes	0.0.0.0:32784->3550/tcp, :::3278
92e70aeb6e1a	ghcr.io/open-telemetry/demo:latest-flagd-ui	"docker-entrypoint.s..."	13 minutes ago	Up 13 minutes	0.0.0.0:32780->4000/tcp, :::3278
5f585750ce10	ghcr.io/open-telemetry/demo:latest-emailservice	"bundle exec ruby em..."	13 minutes ago	Up 13 minutes	0.0.0.0:32783->6060/tcp, :::3278
1f8cc4df0c7	ghcr.io/open-telemetry/demo:latest-currencyservice	"sh -c './usr/local/..."	13 minutes ago	Up 13 minutes	0.0.0.0:32779->7001/tcp, :::3277
a080e2205b43	ghcr.io/open-telemetry/demo:latest-adservice	"./build/install/ope..."	13 minutes ago	Up 13 minutes	0.0.0.0:32778->9555/tcp, :::3277
3cbaffb0d0e1	ghcr.io/open-telemetry/demo:latest-frauddetectionservice	"java -jar frauddete..."	13 minutes ago	Up 13 minutes	0.0.0.0:32777->8090/tcp, :::3277
004613b9b3df	ghcr.io/open-telemetry/demo:latest-quoteservice	"docker-php-entrypoi..."	13 minutes ago	Up 13 minutes	0.0.0.0:32776->8081/tcp, :::3277
664cf59bcd14a	ghcr.io/open-telemetry/demo:latest-accountingservice	"./instrument.sh dot..."	13 minutes ago	Up 13 minutes	0.0.0.0:32775->55679/tcp, 0.0.0.0:32775->
ce2727cac095	ghcr.io/open-telemetry/demo:latest-imageprovider	"./docker-entrypoint..."	13 minutes ago	Up 13 minutes	0.0.0.0:32774->4318/tcp, :::32774->4318/tcp
1add7fc9ca17	otel/opentelemetry-collector-contrib:0.113.0	"/otelcol-contrib --..."	13 minutes ago	Up 13 minutes	0.0.0.0:32774->4318/tcp, :::32774->4318/tcp

7. Checking Application Logs

View the logs of your running applications to confirm they are up and running.

docker logs grafana

```
[ec2-user@ip-172-31-24-124 opentelemetry-demo]$ docker logs grafana
✓Downloaded and extracted grafana-opensearch-datasource v2.22.1 zip successfully to /var/lib/grafana/plugins/grafana-opensearch-datasource

Please restart Grafana after installing or removing plugins. Refer to Grafana documentation for instructions if necessary.

logger=settings t=2024-12-04T14:45:46.004112674Z level=info msg="Starting Grafana" version=11.3.0 commit=d9455ff7db73b694db7d412e49a68bec767f2b5a branch=HEAD compiled=2024-12-04T14:45:45Z
logger=settings t=2024-12-04T14:45:46.007407413Z level=info msg="Config loaded from" file=/usr/share/grafana/conf/defaults.ini
logger=settings t=2024-12-04T14:45:46.007426524Z level=info msg="Config loaded from" file=/etc/grafana/grafana.ini
logger=settings t=2024-12-04T14:45:46.007431765Z level=info msg="Config overridden from command line" arg="default.paths.data=/var/lib/grafana"
logger=settings t=2024-12-04T14:45:46.007436626Z level=info msg="Config overridden from command line" arg="default.paths.logs=/var/log/grafana"
logger=settings t=2024-12-04T14:45:46.007440924Z level=info msg="Config overridden from command line" arg="default.paths.plugins=/var/lib/grafana/plugins"
logger=settings t=2024-12-04T14:45:46.007444766Z level=info msg="Config overridden from command line" arg="default.paths.provisioning=/etc/grafana/provisioning"
logger=settings t=2024-12-04T14:45:46.007449018Z level=info msg="Config overridden from command line" arg="default.log.mode=console"
logger=settings t=2024-12-04T14:45:46.007453443Z level=info msg="Config overridden from Environment variable" var="GF_PATHS_DATA=/var/lib/grafana"
logger=settings t=2024-12-04T14:45:46.007457685Z level=info msg="Config overridden from Environment variable" var="GF_PATHS_LOGS=/var/log/grafana"
logger=settings t=2024-12-04T14:45:46.007462762Z level=info msg="Config overridden from Environment variable" var="GF_PATHS_PLUGINS=/var/lib/grafana/plugins"
logger=settings t=2024-12-04T14:45:46.007468432Z level=info msg="Config overridden from Environment variable" var="GF_PATHS_PROVISIONING=/etc/grafana/provisioning"
logger=settings t=2024-12-04T14:45:46.007493068Z level=info msg="Target target=[all]"
logger=settings t=2024-12-04T14:45:46.008809862 level=info msg="Path Home" path=/usr/share/grafana
logger=settings t=2024-12-04T14:45:46.008854797Z level=info msg="Path Data" path=/var/lib/grafana
logger=settings t=2024-12-04T14:45:46.008859885Z level=info msg="Path Logs" path=/var/log/grafana
logger=settings t=2024-12-04T14:45:46.008864496Z level=info msg="Path Plugins" path=/var/lib/grafana/plugins
logger=settings t=2024-12-04T14:45:46.008877446Z level=info msg="Path Provisioning" path=/etc/grafana/provisioning
logger=settings t=2024-12-04T14:45:46.008881537Z level=info msg="App mode production"
logger=featuremgmt t=2024-12-04T14:45:46.016628579Z level=info msg="FeatureToggles topnav=true logsExploreTableVisualisation=true transformationsVariableSupport=true tlsMemcached=true notificationBanner=true autoMigrateXYChartPanel=true accessControlOnCall=true alertingNoDataErrorExecution=true angularDeprecationUI=true dashboardSceneSolo=true logRowsPopoverMenu=true cloudWatchRoundUpEndTime=true formatString=true publicDashboards=true ssoSettingsApi=true exploreMetrics=true prometheusAzureOverrideAudience=true logsContextDataSourceUi=true managedPluginsInstall=true nestedFolders=true logsInfiniteScrolling=true recoveryThreshold=true transformationsRedesign=true lokiMetricDataPlane=true aNotificationPermissionUpdate=true dashboardSceneForViewers=true prometheusConfigOverhaulAuth=true openSearchBackendFlowEnabled=true promQLScope=true kubernetesPlaylists=true cloudWatchNewLabelPanning=true recordedQueriesMulti=true influxdbBackendMigration=true awsAsyncQueryCaching=true groupToNestedTableTransformation=true panelMonitoring=true pinNavItems=true prometheusMetricEncyclopedia=true dataPlaneFrontendFallback=true dashboardScene=true alertingSimplifiedRouting=true addFieldFromCalculationStatFunctions=true cloudWatchCrossAccountQuerying=true dashgpt=true lokiStructuredMetadata=true correlations=true lokiQuerySplitting=true lokiQueryHints=true alertingInsights=true publicDashboardsScene=true"
logger=sqlstore t=2024-12-04T14:45:46.016703854Z level=info msg="Connecting to DB" dbtype=sqlite3
logger=sqlstore t=2024-12-04T14:45:46.01764188Z level=info msg="Creating SQLite database file" path=/var/lib/grafana/grafana.db
logger=migrator t=2024-12-04T14:45:46.0085384493Z level=info msg="Locking database"
logger=migrator t=2024-12-04T14:45:46.0085421315Z level=info msg="Starting DB migrations"
logger=migrator t=2024-12-04T14:45:46.008716241Z level=info msg="Executing migration" id="create migration_log table"
```

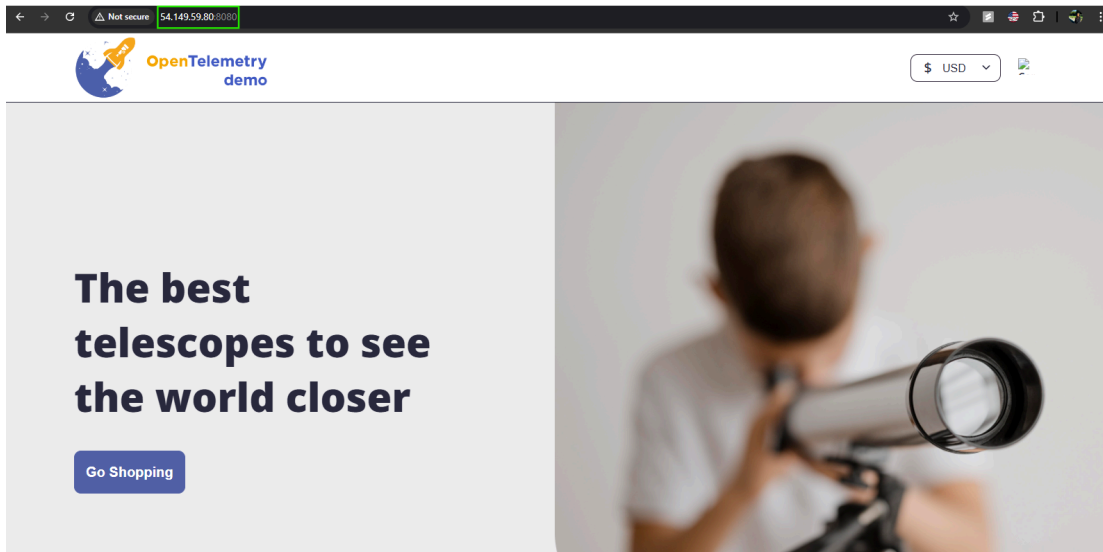
Docker logs flagd

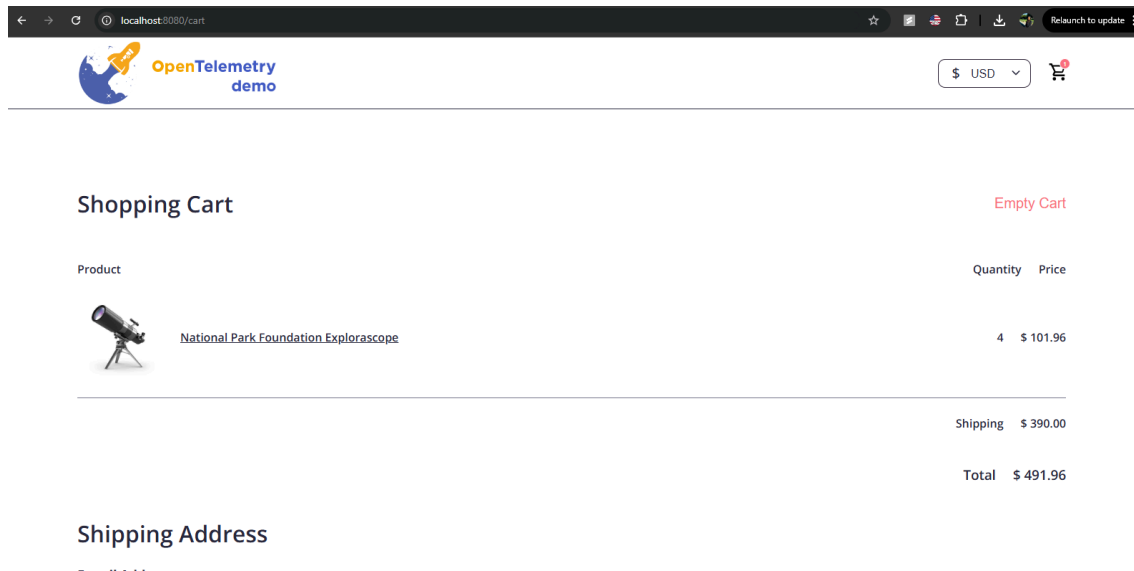
```
logger=infra.usagstats t=2024-12-04T14:47:48.762490371Z level=info msg="Usage stats are ready to report"
[ec2-user@ip-172-31-24-124 opentelemetry-demo]$ docker logs flagd

2024-12-04T14:45:42.0622Z    info    cmd/start.go:121      flagd version: v0.11.4 (df54b6612448609df642fa6fe644388558fd0843), built at: 2024-10-28 {"component": "start"}
2024-12-04T14:45:42.0812Z    info    flag-sync/sync_service.go:54    starting flag sync service on port 8015 {"component": "FlagSyncService"}
2024-12-04T14:45:42.0882Z    info    file/filepath_sync.go:62    Starting filepath sync notifier {"component": "sync", "sync": "fileinfo"}
2024-12-04T14:45:42.0962Z    info    ofrep/ofrep_service.go:56    ofrep service listening at 8016 {"component": "OFREPService"}
2024-12-04T14:45:42.1012Z    info    file/filepath_sync.go:101    watching filepath: /etc/flagd/demo.flagd.json {"component": "sync", "sync": "fileinfo"}
2024-12-04T14:45:42.1042Z    info    flag-evaluation/connect_service.go:227    Flag IResolver listening at [::]:8013 {"component": "service"}
2024-12-04T14:45:42.1072Z    info    flag-evaluation/connect_service.go:247    metrics and probes listening at 8014 {"component": "service"}
[ec2-user@ip-172-31-24-124 opentelemetry-demo]$
```

8. Confirming Applications are Running

Application running:



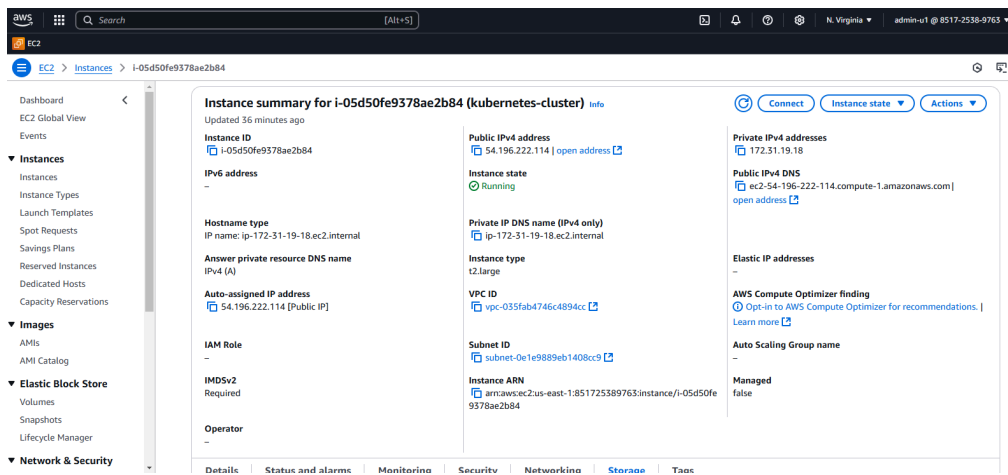


9. **Clean up** of the resources created.

2. Kubernetes Setup Tasks

1. EC2 Instance Setup

- **Provisioned a Dedicated EC2 Instance** for interacting with the EKS cluster.



- **Installed and Configured** the following tools:
 - **kubectl**: To manage Kubernetes clusters.
 - **AWS CLI**: For interacting with AWS services.
 - **eksctl**: To create and manage the EKS cluster.

2. Setting Up the EKS Cluster

- **Created an EKS Cluster** using the **eksctl** command with the following specifications:

- At least **2 worker nodes**.
- Worker node instance type: **large**.

```
eksctl create cluster --name my-eks-cluster --nodes 2 --node-type t3.large
```

3. Checking Cluster Health

- **Validated the EKS Cluster** to ensure it was created successfully and the nodes were running

```
[ec2-user@ip-172-31-19-18 kubernetes]$ kubectl get nodes
NAME                                STATUS    ROLES    AGE   VERSION
ip-192-168-27-195.ec2.internal      Ready    <none>    48m   v1.30.6-eks-94953ac
ip-192-168-33-229.ec2.internal      Ready    <none>    48m   v1.30.6-eks-94953ac
[ec2-user@ip-172-31-19-18 kubernetes]$
```

Verify the Cluster: After the cluster is created, check its details:

```
2024-12-06 07:48:07 [✓] saved kubeconfig as "/home/ec2-user/.kube/config"
2024-12-06 07:48:07 [✓] no tasks
2024-12-06 07:48:07 [✓] all EKS cluster resources for "otel-demo-cluster" have been created
2024-12-06 07:48:07 [✓] nodegroup "workers" has 2 node(s)
2024-12-06 07:48:07 [✓] node "ip-192-168-27-195.ec2.internal" is ready
2024-12-06 07:48:07 [✓] node "ip-192-168-33-229.ec2.internal" is ready
2024-12-06 07:48:07 [✓] waiting for at least 2 node(s) to become ready in "workers"
2024-12-06 07:48:07 [✓] nodegroup "workers" has 2 node(s)
2024-12-06 07:48:07 [✓] node "ip-192-168-27-195.ec2.internal" is ready
2024-12-06 07:48:07 [✓] node "ip-192-168-33-229.ec2.internal" is ready
2024-12-06 07:48:07 [✓] created 1 managed nodegroup(s) in cluster "otel-demo-cluster"
2024-12-06 07:48:08 [✓] kubectrl command should work with "/home/ec2-user/.kube/config", try 'kubectl get nodes'
2024-12-06 07:48:08 [✓] EKS cluster "otel-demo-cluster" in "us-east-1" region is ready
[ec2-user@ip-172-31-19-18 ~]$ eksctl get cluster --name otel-demo-cluster
NAME                                VERSION STATUS    CREATED              VPC                                SUBNETS
otel-demo-cluster                   1.30     ACTIVE  2024-12-06T07:36:05Z vpc-0cbf185e7ed32c3d3  subnet-05d855d755b8aca17, subnet-084bd0d0e385f75b8, subnet-0a0343bec3c
d07c8b, subnet-0fb095bc5230d27a0 sg-031250eeabc2d499f EKS
[ec2-user@ip-172-31-19-18 ~]$
```

4. Deploying the Application

- **Deployed the Application** to the EKS cluster using the provided **opentelemetry-demo.yaml** manifest file:

5. Verify the Deployment

- **Verified that all Pods and Services** were running as expected in the **otel-demo** namespace

```
ec2-user@ip-172-31-19-18:~/c X + v
deployment.apps/opentelemetry-demo-paymentsservice created
deployment.apps/opentelemetry-demo-productcatalogservice created
deployment.apps/opentelemetry-demo-quoteservice created
deployment.apps/opentelemetry-demo-recommendationservice created
deployment.apps/opentelemetry-demo-shippingservice created
deployment.apps/opentelemetry-demo-valkey created
statefulset.apps/otel-demo-opensearch created
[ec2-user@ip-172-31-19-18 ~]$ kubectl get all -n otel-demo
NAME                                READY    STATUS              RESTARTS   AGE
pod/opentelemetry-demo-accountingservice-845b86fbb7-4zmvp    0/1      Init:0/1            0           8s
pod/opentelemetry-demo-adservice-64c6fb9bc7-s8w2m           0/1      ContainerCreating   0           8s
pod/opentelemetry-demo-cartservice-799f5d8bf6-blkkr          0/1      Init:0/1            0           8s
pod/opentelemetry-demo-checkoutservice-586cd686c7-lmxd7       0/1      Init:0/1            0           8s
pod/opentelemetry-demo-currencyservice-6879f8777c-vpsd2      0/1      ContainerCreating   0           8s
pod/opentelemetry-demo-emailservice-546684d7cd-dk65x         0/1      ContainerCreating   0           8s
pod/opentelemetry-demo-flagd-64f8787cc7-gz9dj               0/2      Init:0/1            0           7s
pod/opentelemetry-demo-frauddetectionservice-687585d6c4-pk8sh 0/1      Init:0/1            0           7s
pod/opentelemetry-demo-frontend-6b856f6669-mq7z8            0/1      ContainerCreating   0           7s
pod/opentelemetry-demo-frontendproxy-5b6d4f87f7-kbjqp        0/1      ContainerCreating   0           7s
pod/opentelemetry-demo-grafana-6ffdbb54f7-cl2sb             0/1      ContainerCreating   0           8s
pod/opentelemetry-demo-imageprovider-5b7854975-4s2jj         0/1      ContainerCreating   0           7s
pod/opentelemetry-demo-jaeger-55b55c7687-vbldv              0/1      ContainerCreating   0           8s
pod/opentelemetry-demo-kafka-5cc647fdb5-vxg54              0/1      ContainerCreating   0           6s
pod/opentelemetry-demo-loadgenerator-5cb458c599-qsqbt        0/1      ContainerCreating   0           6s
pod/opentelemetry-demo-otelcol-6b4bcc9886-5rwwd             0/1      ContainerCreating   0           8s
pod/opentelemetry-demo-paymentsservice-868bd5b949-frk2z      0/1      ContainerCreating   0           6s
pod/opentelemetry-demo-productcatalogservice-676f795586-dh65g 0/1      ContainerCreating   0           6s
pod/opentelemetry-demo-prometheus-server-669f8c9d85-hbv5r    0/1      ContainerCreating   0           8s
pod/opentelemetry-demo-quoteservice-5d9fb4ff94-zc5q8        0/1      ContainerCreating   0           6s
pod/opentelemetry-demo-recommendationservice-798597fdd5-fbgfv 0/1      ContainerCreating   0           5s
pod/opentelemetry-demo-shippingservice-5c7bf8c68-mkq75      0/1      ContainerCreating   0           5s
pod/opentelemetry-demo-valkey-7d7bc94657-5mvsh              0/1      ContainerCreating   0           5s
pod/otel-demo-opensearch-0                                   0/1      Init:0/1            0           8s

NAME                                AGE                                TYPE                                CLUSTER-IP    EXTERNAL-IP    PORT(S)
service/opentelemetry-demo-adservice 9s                                ClusterIP      10.100.72.133 <none>         8080/TCP
service/opentelemetry-demo-cartservice 9s                                ClusterIP      10.100.228.176 <none>         8080/TCP
service/opentelemetry-demo-checkoutservice 9s                                ClusterIP      10.100.165.252 <none>         8080/TCP
```

NAME	AGE	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)
service/opentelemetry-demo-adservice	9s	ClusterIP	10.100.72.133	<none>	8080/TCP
service/opentelemetry-demo-cartservice	9s	ClusterIP	10.100.228.176	<none>	8080/TCP
service/opentelemetry-demo-checkoutservice	9s	ClusterIP	10.100.165.252	<none>	8080/TCP
service/opentelemetry-demo-currencyservice	9s	ClusterIP	10.100.247.236	<none>	8080/TCP
service/opentelemetry-demo-emailservice	9s	ClusterIP	10.100.169.221	<none>	8080/TCP
service/opentelemetry-demo-flagd	9s	ClusterIP	10.100.25.233	<none>	8013/TCP, 4000/TCP
service/opentelemetry-demo-frontend	9s	ClusterIP	10.100.65.167	<none>	8080/TCP
service/opentelemetry-demo-frontendproxy	9s	ClusterIP	10.100.183.71	<none>	8080/TCP
service/opentelemetry-demo-grafana	9s	ClusterIP	10.100.34.179	<none>	80/TCP
service/opentelemetry-demo-imageprovider	9s	ClusterIP	10.100.255.73	<none>	8081/TCP
service/opentelemetry-demo-jaeger-agent	9s	ClusterIP	None	<none>	5775/UDP, 5778/TCP, 6831/UDP, 6832/UDP
service/opentelemetry-demo-jaeger-collector	9s	ClusterIP	None	<none>	9411/TCP, 14250/TCP, 14267/TCP, 14268/TCP, 4317/TCP, 4318/TCP
service/opentelemetry-demo-jaeger-query	9s	ClusterIP	None	<none>	16686/TCP, 16685/TCP
service/opentelemetry-demo-kafka	9s	ClusterIP	10.100.152.156	<none>	9092/TCP, 9093/TCP
service/opentelemetry-demo-loadgenerator	9s	ClusterIP	10.100.224.193	<none>	8089/TCP
service/opentelemetry-demo-otelcol	9s	ClusterIP	10.100.189.179	<none>	6831/UDP, 14250/TCP, 14268/TCP, 8888/TCP, 4317/TCP, 4318/TCP, 9464/TCP
service/opentelemetry-demo-paymentsservice	9s	ClusterIP	10.100.197.201	<none>	8080/TCP
service/opentelemetry-demo-productcatalogservice	9s	ClusterIP	10.100.52.228	<none>	8080/TCP

NAME	READY	UP-TO-DATE	AVAILABLE	AGE
deployment.apps/opentelemetry-demo-accountingservice	1/1	1	1	31m
deployment.apps/opentelemetry-demo-adservice	1/1	1	1	31m
deployment.apps/opentelemetry-demo-cartservice	1/1	1	1	31m
deployment.apps/opentelemetry-demo-checkoutservice	1/1	1	1	31m
deployment.apps/opentelemetry-demo-currencyservice	1/1	1	1	31m
deployment.apps/opentelemetry-demo-emailservice	1/1	1	1	31m
deployment.apps/opentelemetry-demo-flagd	1/1	1	1	31m
deployment.apps/opentelemetry-demo-frauddetectionservice	1/1	1	1	31m
deployment.apps/opentelemetry-demo-frontend	1/1	1	1	31m
deployment.apps/opentelemetry-demo-frontendproxy	1/1	1	1	31m
deployment.apps/opentelemetry-demo-grafana	1/1	1	1	31m
deployment.apps/opentelemetry-demo-imageprovider	1/1	1	1	31m
deployment.apps/opentelemetry-demo-jaeger	1/1	1	1	31m
deployment.apps/opentelemetry-demo-kafka	1/1	1	1	31m
deployment.apps/opentelemetry-demo-loadgenerator	1/1	1	1	31m
deployment.apps/opentelemetry-demo-otelcol	1/1	1	1	31m
deployment.apps/opentelemetry-demo-paymentservice	1/1	1	1	31m
deployment.apps/opentelemetry-demo-productcatalogservice	1/1	1	1	31m
deployment.apps/opentelemetry-demo-prometheus-server	1/1	1	1	31m
deployment.apps/opentelemetry-demo-quoteservice	1/1	1	1	31m
deployment.apps/opentelemetry-demo-recommendationservice	1/1	1	1	31m
deployment.apps/opentelemetry-demo-shippingservice	1/1	1	1	31m
deployment.apps/opentelemetry-demo-valkey	1/1	1	1	31m

NAME	DESIRED	CURRENT	READY	AGE
replicaset.apps/opentelemetry-demo-accountingservice-845b86fbb7	1	1	1	31m
replicaset.apps/opentelemetry-demo-adservice-64c6fb9bc7	1	1	1	31m
replicaset.apps/opentelemetry-demo-cartservice-799f5d8bf6	1	1	1	31m
replicaset.apps/opentelemetry-demo-checkoutservice-586cd686c7	1	1	1	31m
replicaset.apps/opentelemetry-demo-currencyservice-6879f8777c	1	1	1	31m
replicaset.apps/opentelemetry-demo-emailservice-546684d7cd	1	1	1	31m
replicaset.apps/opentelemetry-demo-flagd-64f8787cc7	1	1	1	31m
replicaset.apps/opentelemetry-demo-frauddetectionservice-687585d6c4	1	1	1	31m
replicaset.apps/opentelemetry-demo-frontend-6b856f6669	1	1	1	31m
replicaset.apps/opentelemetry-demo-frontendproxy-5b6d4f87f7	1	1	1	31m
replicaset.apps/opentelemetry-demo-grafana-6ffddb54f7	1	1	1	31m
replicaset.apps/opentelemetry-demo-imageprovider-5b7854975	1	1	1	31m

7. Check Service Details

- **Checked Service Status** to ensure the services were up and running:

```
[ec2-user@ip-172-31-19-18 kubernetes]$ kubectl get svc -n otel-demo
```

NAME	AGE	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)
opentelemetry-demo-adservice	13m	ClusterIP	10.100.72.133	<none>	8080/TCP
opentelemetry-demo-cartservice	13m	ClusterIP	10.100.228.176	<none>	8080/TCP
opentelemetry-demo-checkoutservice	13m	ClusterIP	10.100.165.252	<none>	8080/TCP
opentelemetry-demo-currencyservice	13m	ClusterIP	10.100.247.236	<none>	8080/TCP
opentelemetry-demo-emailservice	13m	ClusterIP	10.100.169.221	<none>	8080/TCP
opentelemetry-demo-flagd	13m	ClusterIP	10.100.25.233	<none>	8013/TCP,4000/TCP
opentelemetry-demo-frontend	13m	ClusterIP	10.100.65.167	<none>	8080/TCP
opentelemetry-demo-frontendproxy	13m	ClusterIP	10.100.183.71	<none>	8080/TCP
opentelemetry-demo-grafana	13m	ClusterIP	10.100.34.179	<none>	80/TCP
opentelemetry-demo-imageprovider	13m	ClusterIP	10.100.255.73	<none>	8081/TCP
opentelemetry-demo-jaeger-agent	13m	ClusterIP	None	<none>	5775/UDP,5778/TCP,6831/UDP,6832/UDP
opentelemetry-demo-jaeger-collector	13m	ClusterIP	None	<none>	9411/TCP,14250/TCP,14267/TCP,14268/TCP,4317/TCP,4318/TCP
opentelemetry-demo-jaeger-query	13m	ClusterIP	None	<none>	16686/TCP,16685/TCP
opentelemetry-demo-kafka	13m	ClusterIP	10.100.152.156	<none>	9092/TCP,9093/TCP
opentelemetry-demo-loadgenerator	13m	ClusterIP	10.100.224.193	<none>	8089/TCP
opentelemetry-demo-otelcol	TCP 13m	ClusterIP	10.100.189.179	<none>	6831/UDP,14250/TCP,14268/TCP,8888/TCP,4317/TCP,4318/TCP,9464/TCP
opentelemetry-demo-paymentservice		ClusterIP	10.100.197.201	<none>	8080/TCP

Pod Logs

To troubleshoot and monitor the application running in the `otel-demo` namespace, we retrieved the logs of specific pods using the following steps:

List Pod Names:

First, we obtained the list of pods in the `otel-demo` namespace:

```
kubectl get pods -n otel-demo
```

Retrieve Pod Logs:

To view the logs of the `opentelemetry-demo-frontend` pod, the following command was executed:

```
kubectl logs opentelemetry-demo-frontend-6b856f6669-mq7z8 -n otel-demo
```

Description:

- **Command:** `kubectl logs <pod-name> -n <namespace>`
 - Fetches the logs of the specified pod in the given namespace.
- **Pod:** `opentelemetry-demo-frontend-6b856f6669-mq7z8`

- Namespace: **otel-demo**

```
[ec2-user@ip-172-31-19-18 ~]$ kubectl get pods -n otel-demo
NAME                                READY   STATUS    RESTARTS   AGE
opentelemetry-demo-accountingservice-845b86fbb7-4zmvp    1/1     Running   0           26m
opentelemetry-demo-adservice-64c6fb9bc7-s8w2m           1/1     Running   0           26m
opentelemetry-demo-cartservice-799f5d8bf6-blkkr         1/1     Running   0           26m
opentelemetry-demo-checkoutservice-586cd686c7-lmxd7      1/1     Running   0           26m
opentelemetry-demo-currencyservice-6879f8777c-vpsd2     1/1     Running   0           26m
opentelemetry-demo-emailservice-546684d7cd-dk65x        1/1     Running   0           26m
opentelemetry-demo-flagd-64f8787cc7-gz9dj              2/2     Running   0           26m
opentelemetry-demo-frauddetectionservice-687585d6c4-pk8sh 1/1     Running   0           26m
opentelemetry-demo-frontend-6b856f6669-mq7z8           1/1     Running   0           26m
opentelemetry-demo-frontendproxy-5b6d4f87f7-kbjqp       1/1     Running   0           26m
opentelemetry-demo-grafana-6ffdbb54f7-cl2sb             1/1     Running   0           26m
opentelemetry-demo-imageprovider-5b7854975-4s2jj        1/1     Running   0           26m
opentelemetry-demo-jaeger-55b55c7687-vbldv             1/1     Running   0           26m
opentelemetry-demo-kafka-5cc647fdb5-vxg54              1/1     Running   0           26m
opentelemetry-demo-loadgenerator-5cb458c599-qsqbt       1/1     Running   0           26m
opentelemetry-demo-otelcol-6b4bcc9886-5rwfd             1/1     Running   0           26m
opentelemetry-demo-payment-service-868bd5b949-frk2z     1/1     Running   0           26m
opentelemetry-demo-productcatalogservice-676f795586-dh65g 1/1     Running   0           26m
opentelemetry-demo-prometheus-server-669f8c9d85-hbv5r   1/1     Running   0           26m
opentelemetry-demo-quoteservice-5d9fb4ff94-zc5q8       1/1     Running   0           26m
opentelemetry-demo-recommendationservice-798597fdd5-fbgfv 1/1     Running   0           26m
opentelemetry-demo-shippingservice-5c7bf8c68-mkq75     1/1     Running   0           26m
opentelemetry-demo-valkey-7d7bc94657-5mvsh             1/1     Running   0           26m
otel-demo-opensearch-0                                1/1     Running   0           26m
[ec2-user@ip-172-31-19-18 ~]$ kubectl logs opentelemetry-demo-frontend-6b856f6669-mq7z8 -n otel-demo

> frontend@0.1.0 start
> node --require ./Instrumentation.js server.js

  ▲ Next.js 14.2.5
  - Local:      http://opentelemetry-demo-frontend-6b856f6669-mq7z8:8080
  - Network:    http://192.168.56.177:8080

  ✓ Starting...
  ✓ Ready in 579ms
[ec2-user@ip-172-31-19-18 ~]$ kubectl logs opentelemetry-demo-adservice-64c6fb9bc7-s8w2m -n otel-demo
```

```
[ec2-user@ip-172-31-19-18 ~]$ kubectl logs opentelemetry-demo-frontend-6b856f6669-mq7z8 -n otel-demo

> frontend@0.1.0 start
> node --require ./Instrumentation.js server.js

  ▲ Next.js 14.2.5
  - Local:      http://opentelemetry-demo-frontend-6b856f6669-mq7z8:8080
  - Network:    http://192.168.56.177:8080

  ✓ Starting...
  ✓ Ready in 579ms
[ec2-user@ip-172-31-19-18 ~]$ kubectl logs opentelemetry-demo-adservice-64c6fb9bc7-s8w2m -n otel-demo
Picked up JAVA_TOOL_OPTIONS: -javaagent:/usr/src/app/opentelemetry-javaagent.jar
OpenJDK 64-Bit Server VM warning: Sharing is only supported for boot loader classes because bootstrap classpath has been appended
[otel.javaagent 2024-12-06 07:59:41:862 +0000] [main] INFO io.opentelemetry.javaagent.tooling.VersionLogger - opentelemetry-javaagent - version: 2.6.0
2024-12-06 07:59:53 - oteldemo.AdService - Ad service starting. trace_id= span_id= trace_flags=
SLF4J(W): No SLF4J providers were found.
SLF4J(W): Defaulting to no-operation (NOP) logger implementation
SLF4J(W): See https://www.slf4j.org/codes.html#noProviders for further details.
2024-12-06 07:59:56 - oteldemo.AdService - Ad service started, listening on 8080 trace_id= span_id= trace_flags=
2024-12-06 08:00:42 - oteldemo.AdService - Targeted ad request received for [assembly] trace_id=3508508e813ce67bb0495a99f6857009 span_id=b301ee5419637565 tr
ace_flags=01
2024-12-06 08:00:43 - oteldemo.AdService - Non-targeted ad request received, preparing random response. trace_id=beb4e8bfa88fc4c525bb6645510acdd7 span_id=9a
8a02d77ac98f91 trace_flags=01
2024-12-06 08:01:01 - oteldemo.AdService - Targeted ad request received for [assembly] trace_id=a190bd9cf24e102887207c3f70645754 span_id=dda394a28e978e00 tr
ace_flags=01
2024-12-06 08:01:02 - oteldemo.AdService - Targeted ad request received for [assembly] trace_id=0d48aae2fda8ca93d6b7cd5f9c87f4de span_id=a420a2376641b77e tr
ace_flags=01
2024-12-06 08:01:30 - oteldemo.AdService - Targeted ad request received for [telescopes] trace_id=d4328a92d8f2e1b6da99379fa03edaa8 span_id=62b6e8cfba9e92c7
trace_flags=01
2024-12-06 08:01:30 - oteldemo.AdService - Targeted ad request received for [assembly] trace_id=3bd89d850fec48464973546a37c1259e span_id=726458832babb568 tr
ace_flags=01
2024-12-06 08:01:38 - oteldemo.AdService - Targeted ad request received for [books] trace_id=793379742e2a8ccc879a76b279738534 span_id=b04cf3af0cd1af60 trace
_flags=01
2024-12-06 08:01:56 - oteldemo.AdService - Targeted ad request received for [binoculars] trace_id=99cd3744ab15895a272506b8b810b5a9 span_id=bc5b57b8b8f92bf6e
trace_flags=01
2024-12-06 08:02:01 - oteldemo.AdService - Targeted ad request received for [telescopes] trace_id=4e6cd0d36f1d498cd1b6d73a87d6a5d5 span_id=83625612d09c8b2b
trace_flags=01
2024-12-06 08:02:09 - oteldemo.AdService - Targeted ad request received for [binoculars] trace_id=13558c3395c02d97b8edd25dd7f2a0eb span_id=872f6facde904a74
trace_flags=01
2024-12-06 08:02:09 - oteldemo.AdService - Targeted ad request received for [binoculars] trace_id=2450fc6b2dbdb191cf3e8c08e6ffe6c3 span_id=d3e736d2539ad254
```

Cluster Configuration Details:

To retrieve and verify the configuration details of the deployed EKS cluster, the following command was executed:

`eksctl get cluster --name otel-demo-cluster`

Description:

- Command: `eksctl get cluster`
 - Fetches details about the specified EKS cluster, including its name, region, version, and status.
- Cluster Name: `otel-demo-cluster`

```
2024-12-06 08:26:50 - oteldemo.AdService - Non-targeted ad request received, preparing random response. trace_id=586c77322dcef3b5f5c451df05f0ec32 span_id=27
f4851bc186440c trace_flags=01
[ec2-user@ip-172-31-19-18 kubernetes]$ eksctl get cluster --name otel-demo-cluster
NAME          VERSION STATUS  CREATED          VPC          SUBNETS          S
ECURITYGROUPS PROVIDER
otel-demo-cluster 1.30    ACTIVE  2024-12-06T07:36:05Z  vpc-0cbf185e7ed32c3d3  subnet-05d855d755b8aca17,subnet-084bd0d0e385f75b8,subnet-0a0343bec3c
d07c8b,subnet-0fb095bc5230d27a0 sg-031250eeabc2d499f  EKS
[ec2-user@ip-172-31-19-18 kubernetes]$
```

Port Forwarding for Frontend Server/Proxy, Grafana, and Prometheus

To access the frontend server/proxy, Grafana, and Prometheus services locally, we performed port forwarding using the following command:

`kubectrl port-forward svc/opentelemetry-demo-frontendproxy -n otel-demo 8080:80`

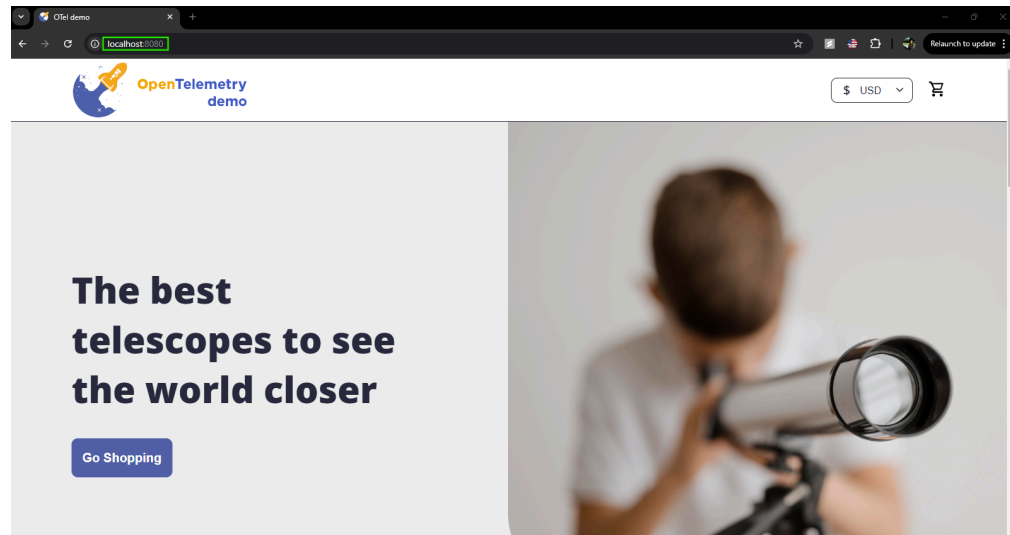
Description:

- Service: `opentelemetry-demo-frontendproxy`
- Namespace: `otel-demo`
- Port Mapping: Forwards traffic from port 8080 on the local machine to port 80 of the frontend proxy service within the cluster.

Accessible Endpoints: After executing the port-forwarding command, the following services became accessible via the respective URLs:

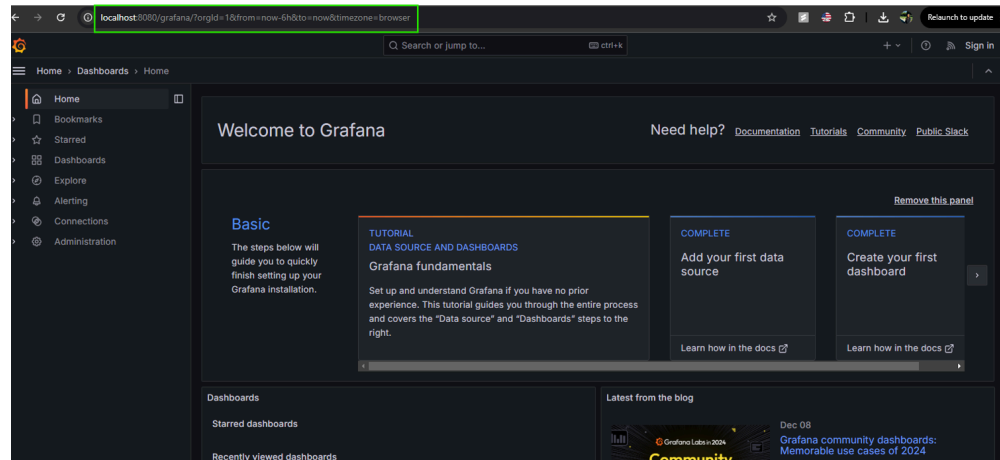
1. **Web Store** – <http://localhost:8080/>

- The main application interface for the deployed web store.



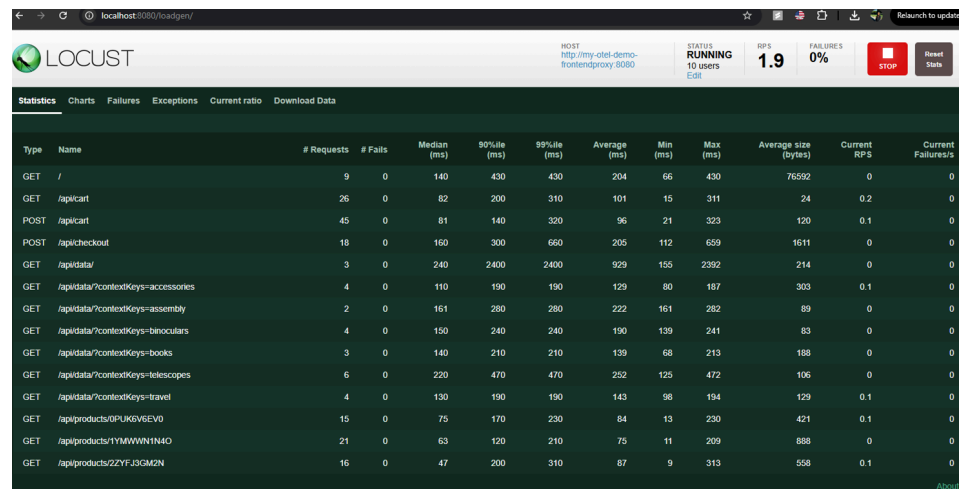
2. Grafana – <http://localhost:8080/grafana/>

- The dashboard for visualizing metrics and monitoring the application's performance.



3. Load Generator UI – <http://localhost:8080/loadgen/>

- An interface to generate synthetic load on the application for testing scalability and performance.

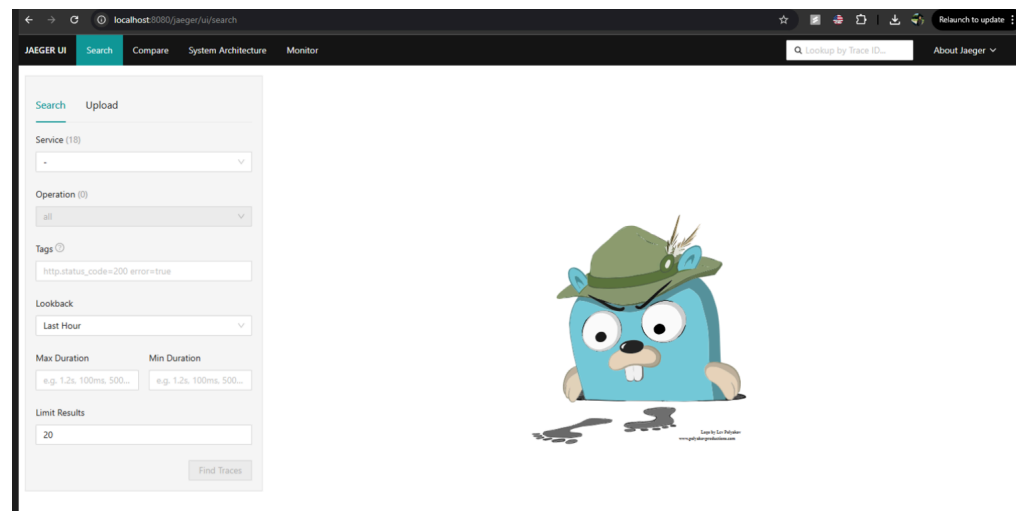


The screenshot shows the Locust web interface at <http://localhost:8080/locust/>. The interface includes a status bar at the top indicating the host is running, with 10 users and 1.9 RPS. Below this is a table of request statistics.

Type	Name	# Requests	# Fails	Median (ms)	90%ile (ms)	99%ile (ms)	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	Current RPS	Current Failures/s
GET	/	9	0	140	430	430	204	66	430	76592	0	0
GET	/api/cart	26	0	82	200	310	101	15	311	24	0.2	0
POST	/api/cart	45	0	81	140	320	96	21	323	120	0.1	0
POST	/api/checkout	18	0	160	300	560	205	112	659	1611	0	0
GET	/api/data/	3	0	240	2400	2400	929	155	2392	214	0	0
GET	/api/data/?contextKeys=accessories	4	0	110	190	190	129	80	187	303	0.1	0
GET	/api/data/?contextKeys=assembly	2	0	161	280	280	222	161	282	89	0	0
GET	/api/data/?contextKeys=binoculars	4	0	150	240	240	190	139	241	83	0	0
GET	/api/data/?contextKeys=books	3	0	140	210	210	139	68	213	188	0	0
GET	/api/data/?contextKeys=telescopes	6	0	220	470	470	252	125	472	106	0	0
GET	/api/data/?contextKeys=travel	4	0	130	190	190	143	98	194	129	0.1	0
GET	/api/products/0PUK6VSEV0	15	0	75	170	230	84	13	230	421	0.1	0
GET	/api/products/1YMWWN1N4O	21	0	63	120	210	75	11	209	888	0	0
GET	/api/products/2ZYFJ3QM2N	16	0	47	200	310	87	9	313	558	0.1	0

4. Jaeger UI – <http://localhost:8080/jaeger/ui/>

- A distributed tracing interface for tracking and analyzing requests through the application.



The screenshot shows the Jaeger UI search interface at <http://localhost:8080/jaeger/ui/search>. The interface includes a search bar and various filters for finding traces.

Search **Upload**

Service (18):


Operation (0):

Tags (0):

Lookback:

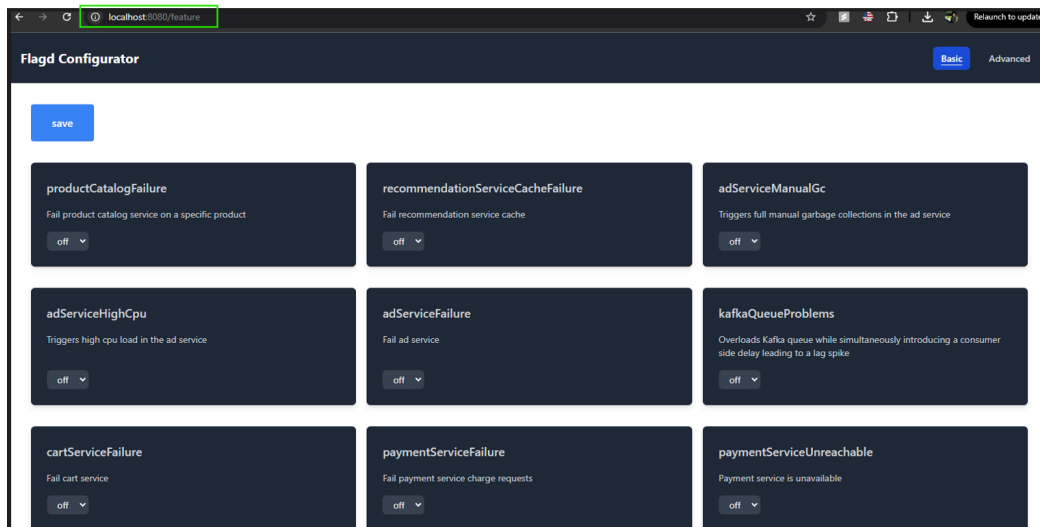
Max Duration: Min Duration:

Limit Results:



5. Flagd Configurator UI – <http://localhost:8080/feature>

- An interface for managing feature flags within the application.



These services were successfully accessed via **port forwarding**, enabling efficient monitoring, tracing, and load testing of the deployed Kubernetes application.
