Kubernetes 7

EFK -> Elasticsearch FluentD Kibana

One component we use to read Logs and store into Elasticsearch, it is FluentD

UI for elastic

Kibana

ElasticSearch

Logs

FluentD

Microservice 1

Logs

Microservice 2

Indexes and stores into ElasticSearch

Logs

Microservice 3

We need Kibana to visualize those Logs

Pod in the end of the day it is our application

Create K8s cluster

eksctl create cluster --name my-eks-cluster --region ca-central-1 --node-type t2.medium --zones ca-central-1a,ca-central-1b

ubuntu@ip-172-31-9-165:~/blue-green-model$ cat blue-deployment.yml

---

apiVersion: apps/v1

kind: Deployment

metadata:

name: javawebbluedeploy

spec:

replicas: 2

strategy:

type: RollingUpdate

selector:

matchLabels:

app: java-web-app

version: v1

color: blue

template:

metadata:

labels:

app: java-web-app

version: v1

color: blue

spec:

containers:

- name: javawebappcontainer

image: hacker123shiva/springbt-in-docker:latest

imagePullPolicy: Always

ports:

- containerPort: 8080

...

ubuntu@ip-172-31-9-165:~/ElasticSearch$ kubectl apply -f deployment.yml

deployment.apps/javawebbluedeploy created

service/javaappsvc created

---

apiVersion: apps/v1

kind: Deployment

metadata:

name: javawebbluedeploy

spec:

replicas: 2

strategy:

type: RollingUpdate

selector:

matchLabels:

app: java-web-app

version: v1

color: blue

template:

metadata:

labels:

app: java-web-app

version: v1

color: blue

spec:

containers:

- name: javawebappcontainer

image: hacker123shiva/springbt-in-docker:latest

imagePullPolicy: Always

ports:

- containerPort: 8080

---

apiVersion: v1

kind: Service

metadata:

name: javaappsvc

spec:

type: LoadBalancer

selector:

app: java-web-app

ports:

- port: 80

targetPort: 8080

...

ubuntu@ip-172-31-9-165:~/ElasticSearch$

ubuntu@ip-172-31-9-165:~/ElasticSearch$ kubectl get pods

NAME READY STATUS RESTARTS AGE

javawebbluedeploy-68fc6554d6-lb9tm 1/1 Running 0 82s

javawebbluedeploy-68fc6554d6-zsqkt 1/1 Running 0 82s

ubuntu@ip-172-31-9-165:~/ElasticSearch$ kubectl get all

NAME READY STATUS RESTARTS AGE

pod/javawebbluedeploy-68fc6554d6-lb9tm 1/1 Running 0 2m27s

pod/javawebbluedeploy-68fc6554d6-zsqkt 1/1 Running 0 2m27s

NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE

service/javaappsvc LoadBalancer 10.100.252.5 af3d90df73b354484b7f8be9d26c267e-1835006855.ca-central-1.elb.amazonaws.com 80:31177/TCP 2m27s

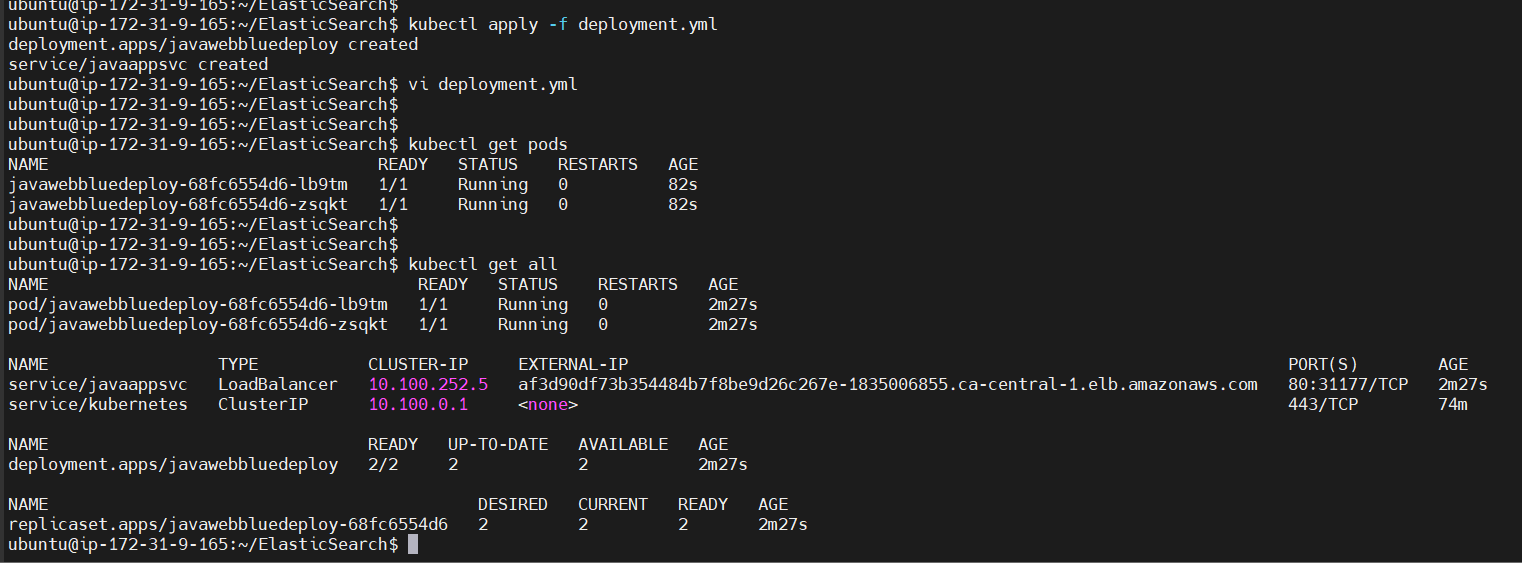
service/kubernetes ClusterIP 10.100.0.1 <none> 443/TCP 74m

NAME READY UP-TO-DATE AVAILABLE AGE

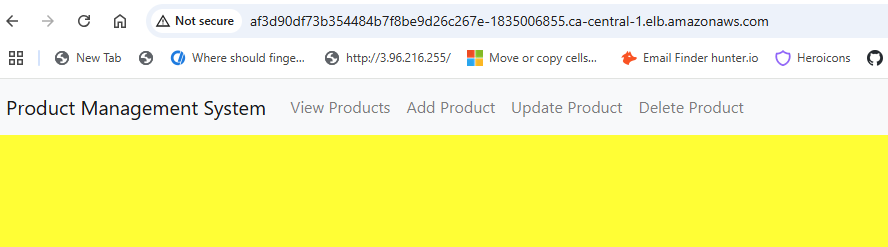
deployment.apps/javawebbluedeploy 2/2 2 2 2m27s

NAME DESIRED CURRENT READY AGE

replicaset.apps/javawebbluedeploy-68fc6554d6 2 2 2 2m27s



Application URL: af3d90df73b354484b7f8be9d26c267e-1835006855.ca-central-1.elb.amazonaws.com



Monitoring application is one part, monitoring cluster is another part

ubuntu@ip-172-31-9-165:~/ElasticSearch$ cat 02-ElasticSearch\_Service.yml

---

apiVersion: v1

kind: Service

metadata:

name: elasticsearch-logging

namespace: efklog

labels:

k8s-app: elasticsearch-logging

kubernetes.io/cluster-service: "true"

addonmanager.kubernetes.io/mode: Reconcile

kubernetes.io/name: "Elasticsearch"

spec:

ports:

- port: 9200

protocol: TCP

targetPort: db

selector:

k8s-app: elasticsearch-logging

...

ubuntu@ip-172-31-9-165:~/ElasticSearch$ ls -l

total 28

-rw-rw-r-- 1 ubuntu ubuntu 65 Jun 15 00:48 01-Namespace.yml

-rw-rw-r-- 1 ubuntu ubuntu 387 Jun 15 01:12 02-ElasticSearch\_Service.yml

-rw-rw-r-- 1 ubuntu ubuntu 1466 Jun 15 02:12 03-ElasticSearch\_StatefulSet.yml

-rw-rw-r-- 1 ubuntu ubuntu 890 Jun 15 02:18 04-Fluentd\_ConfigMap.yml

-rw-rw-r-- 1 ubuntu ubuntu 1495 Jun 15 02:16 05-Fluentd\_DaemonSet.yml

-rw-rw-r-- 1 ubuntu ubuntu 737 Jun 15 02:27 Kibana\_Deployment.yml

-rw-rw-r-- 1 ubuntu ubuntu 711 Jun 14 22:59 deployment.yml

ubuntu@ip-172-31-9-165:~/ElasticSearch$ mv Kibana\_Deployment.yml 06-kibana\_deployment.yml

ubuntu@ip-172-31-9-165:~/ElasticSearch$

ubuntu@ip-172-31-9-165:~/ElasticSearch$

ubuntu@ip-172-31-9-165:~/ElasticSearch$ kubectl apply -f 01-Namespace.yml

namespace/efklog created

ubuntu@ip-172-31-9-165:~/ElasticSearch$ kubectl apply -f 02-ElasticSearch\_Service.yml

service/elasticsearch-logging created

ubuntu@ip-172-31-9-165:~/ElasticSearch$ kubectl apply -f 03-ElasticSearch\_StatefulSet.yml

serviceaccount/elasticsearch-logging created

statefulset.apps/elasticsearch-logging created

ubuntu@ip-172-31-9-165:~/ElasticSearch$ kubectl apply -f 04-Fluentd\_ConfigMap.yml

configmap/fluentd-config created

ubuntu@ip-172-31-9-165:~/ElasticSearch$ kubectl apply -f 05-Fluentd\_DaemonSet.yml

serviceaccount/fluentd created

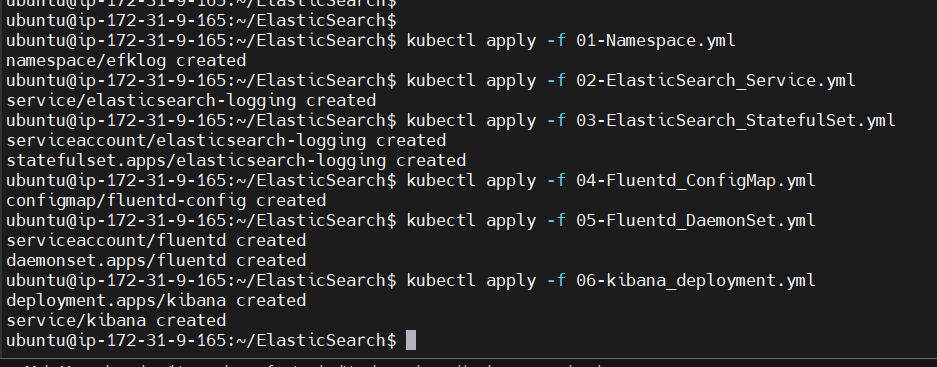
daemonset.apps/fluentd created

ubuntu@ip-172-31-9-165:~/ElasticSearch$ kubectl apply -f 06-kibana\_deployment.yml

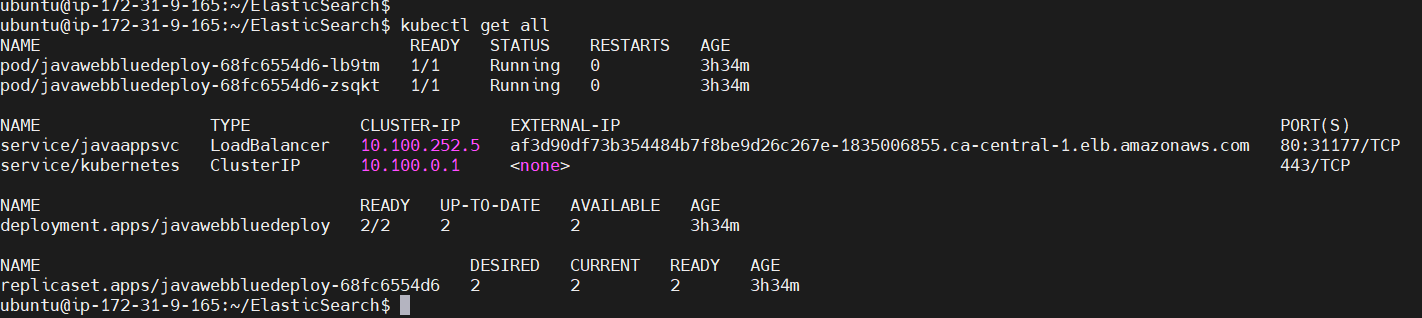
deployment.apps/kibana created

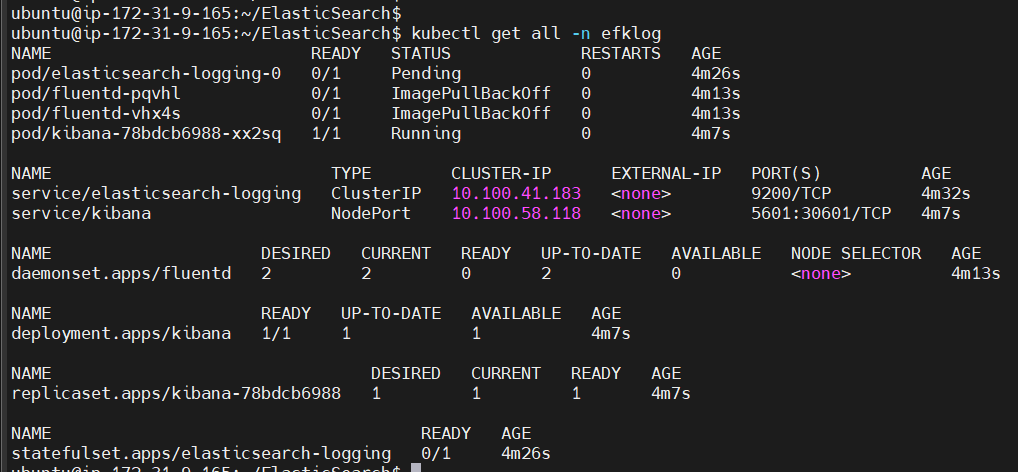
service/kibana created

ubuntu@ip-172-31-9-165:~/ElasticSearch$

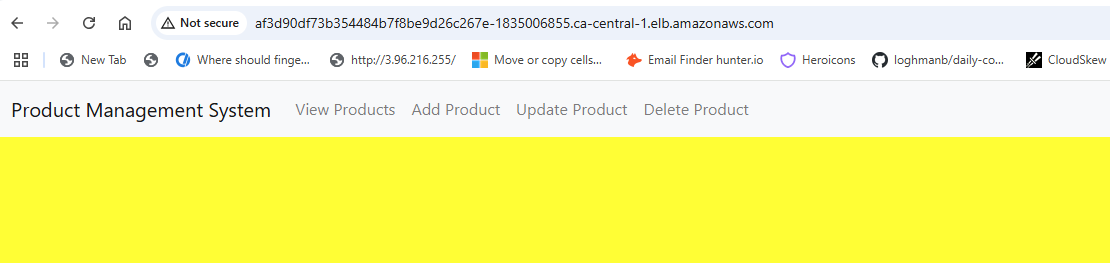


Alternative for EFK is Splunk



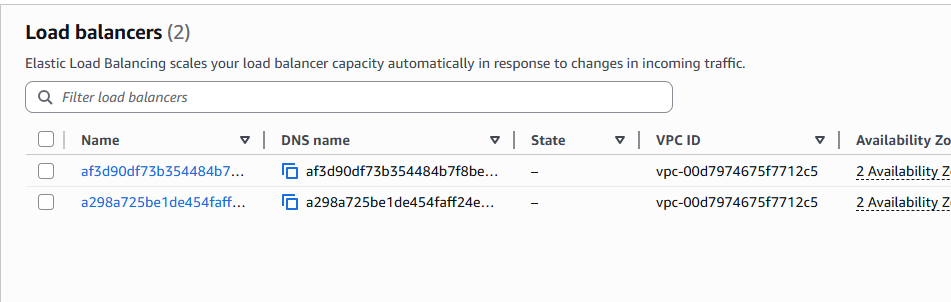


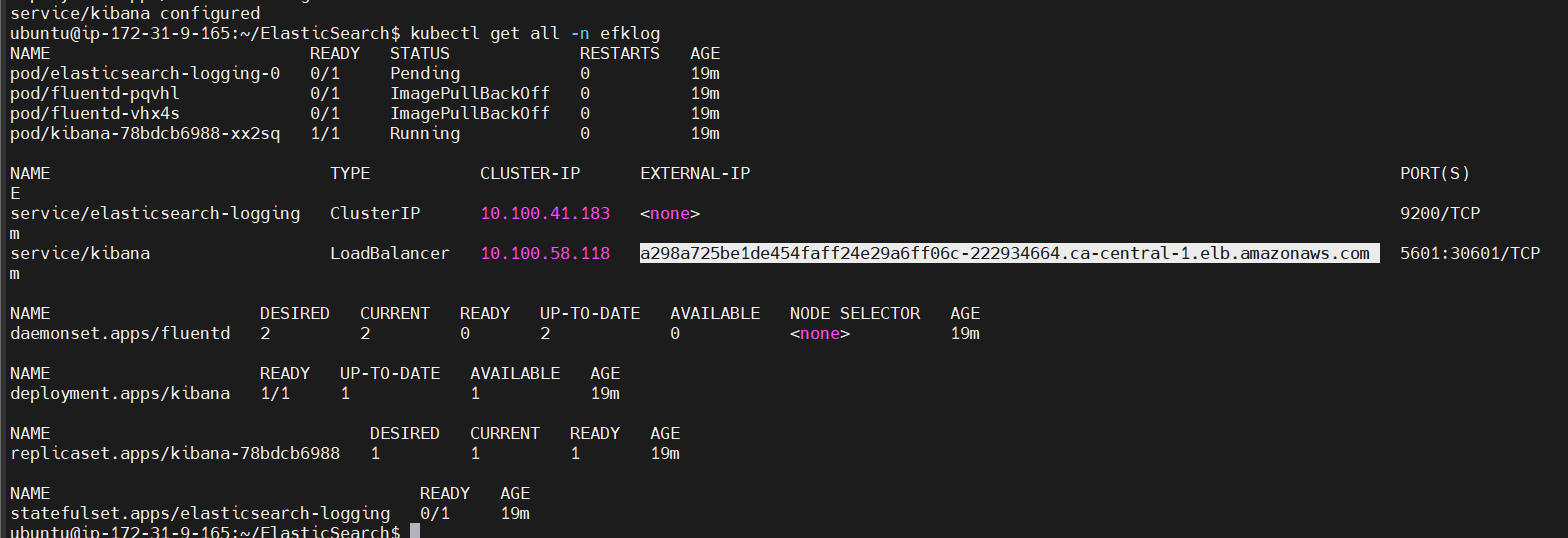
<http://af3d90df73b354484b7f8be9d26c267e-1835006855.ca-central-1.elb.amazonaws.com/>



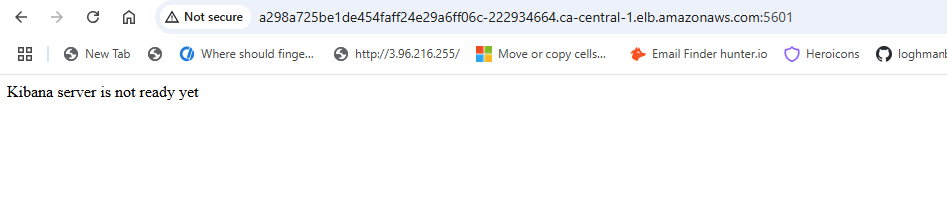
FluentD must be available in every Worker Node and if you want something to be available in all WorkerNodes then DaemonSet comes into picture

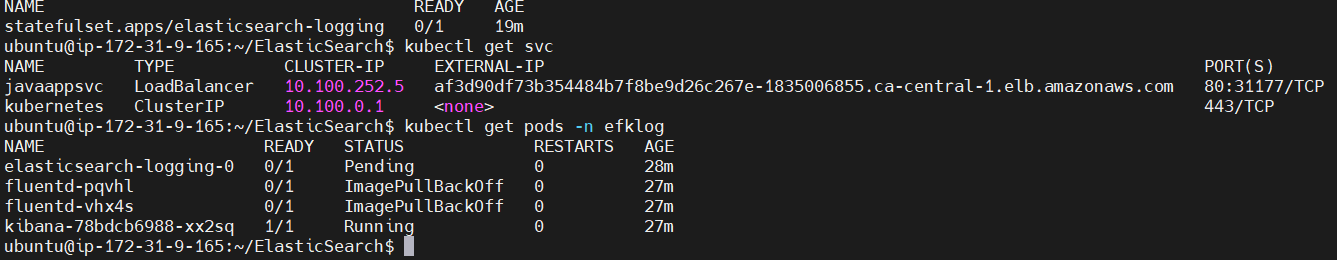
We see 2 loadbalancers, second one is for Kibana





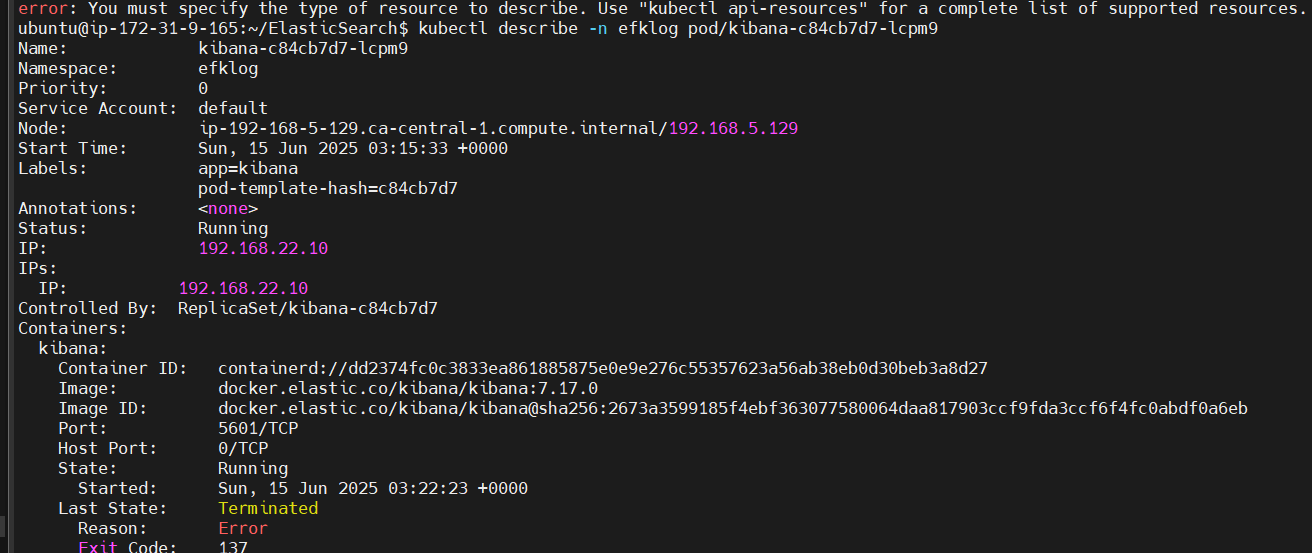
<http://a298a725be1de454faff24e29a6ff06c-222934664.ca-central-1.elb.amazonaws.com:5601/>

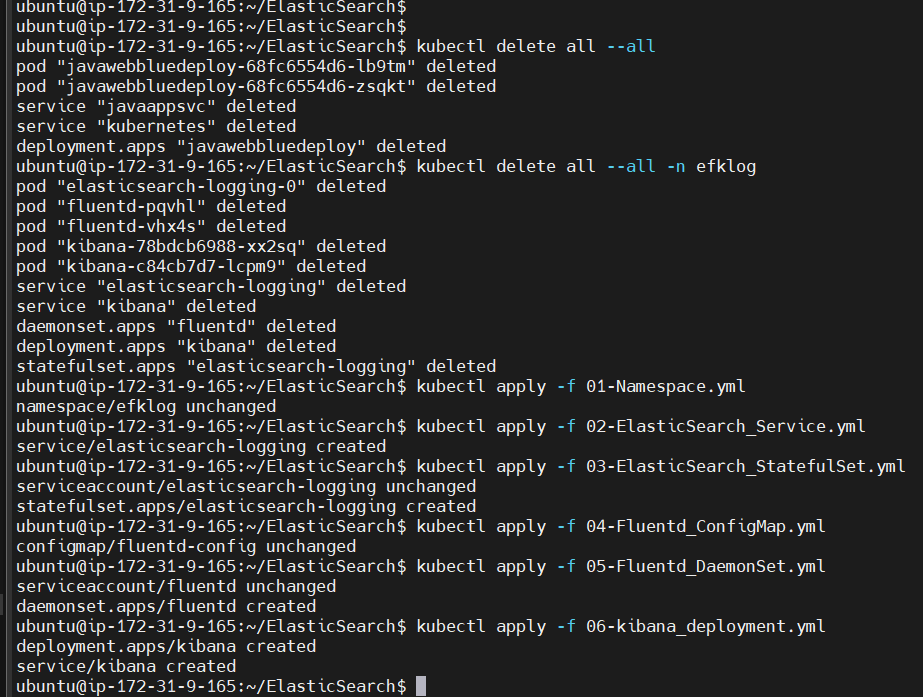


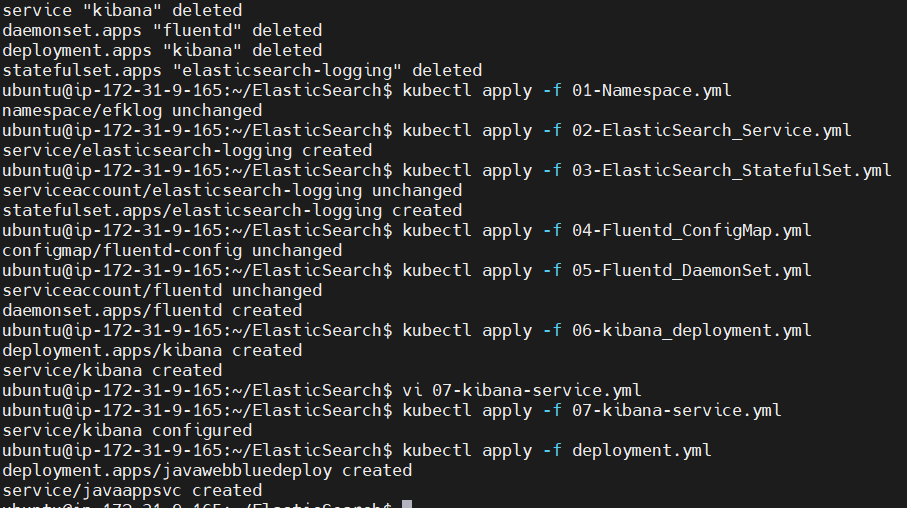


It is not able to connect to ElasticSearch

ubuntu@ip-172-31-9-165:~/ElasticSearch$ kubectl describe -n efklog pod/kibana-c84cb7d7-lcpm9







1:16

eksctl delete cluster --name my-eks-cluster --region ca-central-1

In a week we will have Kibana yml files