**Service Mesh Implementation**

**Step 1: Setup EKS Cluster** using Terraform <https://github.com/jaiswaladi246/Mega-Project-Terraform.git>

**Step 2: Install Istio using istioctl**

**Download and install Istio CLI:**

curl -L https://istio.io/downloadIstio | sh -

cd istio-1.26.0

export PATH=$PWD/bin:$PATH

**Install Istio control plane:**

istioctl install --set profile=demo -y

It installs:

* istiod (control plane)
* istio-ingressgateway (default gateway)

**Step 3: Label Namespace for Auto Sidecar Injection**

kubectl create ns webapps

kubectl label namespace webapps istio-injection=enabled

**Step 4: Deploy Sample App (Bookinfo)**

kubectl apply -f samples/bookinfo/platform/kube/bookinfo.yaml -n webapps

Deploys:

* productpage, details, reviews, ratings microservices

Check:

kubectl get pods -n webapps

**Step 5: Expose App via Istio Gateway**

kubectl apply -f samples/bookinfo/networking/bookinfo-gateway.yaml -n webapps

---

apiVersion: networking.istio.io/v1

kind: Gateway

metadata:

name: bookinfo-gateway

spec:

selector:

istio: ingressgateway

servers:

- port:

number: 8080

name: http

protocol: HTTP

hosts:

- "\*"

---

apiVersion: networking.istio.io/v1

kind: VirtualService

metadata:

name: bookinfo

spec:

hosts:

- "\*"

gateways:

- bookinfo-gateway

http:

- match:

- uri:

exact: /productpage

- uri:

prefix: /static

- uri:

exact: /login

- uri:

exact: /logout

- uri:

prefix: /api/v1/products

route:

- destination:

host: productpage

port:

number: 9080

Get EXTERNAL IP:

kubectl get svc istio-ingressgateway -n istio-system

**Step 6: Enable mTLS for the App**

Apply PeerAuthentication:

# strict-mtls.yaml

apiVersion: security.istio.io/v1beta1

kind: PeerAuthentication

metadata:

name: default

namespace: webapps

spec:

mtls:

mode: STRICT

kubectl apply -f strict-mtls.yaml

**Step 8: Enable Observability**

**Deploy addons:**

kubectl apply -f samples/addons

**Destination Rule:**

apiVersion: networking.istio.io/v1beta1

kind: DestinationRule

metadata:

name: reviews

spec:

host: reviews

subsets:

- name: v1

labels:

version: v1

- name: v2

labels:

version: v2

- name: v3

labels:

version: v3

Virtual Service:

---

apiVersion: networking.istio.io/v1beta1

kind: VirtualService

metadata:

name: reviews

namespace: webapps

spec:

hosts:

- reviews

http:

- route:

- destination:

host: reviews

subset: v1

weight: 40

- destination:

host: reviews

subset: v3

weight: 30

- destination:

host: reviews

subset: v2

weight: 30