Linkerd in a Minikube environment

- Prerequisites
- Set Up Minikube
  - Start Minikube
  - Verify Minikube Status
- Deploy a Sample Application
  - Deploy Emojivoto
  - Verify the Application
- Install Linkerd
  - Install the Linkerd CLI
  - Pre-Installation Check
  - Install Linkerd on Your Minikube Cluster
  - Verify Linkerd Installation
  - Inject Linkerd Proxies
- Explore Linkerd Features
  - Dashboard
  - Traffic Management
  - Security
- Monitoring and Debugging
  - Linkerd Tap
  - Linkerd Top
- Cleaning Up
- Troubleshooting Common Errors
- Additional Resources

Prerequisites

- Minikube: Ensure you have Minikube installed and running. You can install it from the official Minikube website.
- kubectl: Make sure you have kubectl installed and configured to work with your Minikube cluster. Kubernetes Tools website
- Linkerd: Familiarize yourself with the basics of Linkerd, a service mesh. linkerd getting started

Set Up Minikube

Start Minikube

```
minikube start
```

Verify Minikube Status

```
minikube status
```

Deploy a Sample Application

Deploy Emojivoto

To see Linkerd in action, use the emojivoto application provided by Linkerd.

```
curl --proto '=https' --tlsv1.2 -sSfL https://run.linkerd.io/emojivoto.yml | kubectl apply -f -
```

Expose the pod

```
kubectl -n emojivoto port-forward svc/web-svc 8080:80
```

You should now be able to view the website at localhost:8080

Verify the Application

```
kubectl get pods -n emojivoto
```

Install Linkerd

Install the Linkerd CLI

You can install the Linkerd CLI using the following command:

```
curl --proto '=https' --tlsv1.2 -sSfL https://run.linkerd.io/install | sh
```

```
export PATH=$PATH:$HOME/.linkerd2/bin
```

```
linkerd version
```

Alternatively, you can download the binary directly from the Linkerd releases page.

Pre-Installation Check

```
linkerd check --pre
```

Install Linkerd on Your Minikube Cluster

`linkerd install --crds | kubectl apply -f -`

`linkerd install --set proxyInit.runAsRoot=true | kubectl apply -f -`

Verify Linkerd Installation

`linkerd check`

Inject Linkerd Proxies

To enable Linkerd for your application, inject the Linkerd proxies into your pods.

`kubectl get deployments -n emojivoto -o yaml | linkerd inject - | kubectl apply -f -`

This command retrieves the deployments in YAML format, injects the Linkerd sidecar, and applies the modified configuration to the Kubernetes cluster
source k8s , source linkerd.

The command `curl -sL run.linkerd.io/emojivoto.yml | linkerd inject -` scans the emojivoto manifest file, skips the rest of the configurations in the manifest, and then injects linkerd-proxy proxies into each deployment in the pod. With the `kubectl apply -f -` command, the emojivoto configuration was re-applied in our cluster and the sidecars were successfully injected.

Explore Linkerd Features

Dashboard

For additional observability features, install the Linkerd Viz extension:

`linkerd viz install | kubectl apply -f -`

`linkerd viz check`

`linkerd viz dashboard`

This sets up the visualization tools, including Prometheus, and launches the Linkerd dashboard
source k8s, source linkerd.

Traffic Management

Linkerd allows you to manage traffic between services. Here's an example of how to split traffic between two versions of a service:

bash

# Create a new deployment for the v2 version of the web service

```
kubectl get deployments web -n emojivoto -o yaml > web-deployment.yaml ; sed -i 's/name: web/name: web-v2/' web-deployment.yaml sed -i 's/image: emojivoto-web:v1/image: emojivoto-web:v2/' web-deployment.yaml ; kubectl apply -f web-deployment.yaml ;rm web-deployment.yaml
```

# Inject Linkerd proxies into the new deployment

`kubectl get deployments web-v2 -n emojivoto -o yaml | linkerd inject - | kubectl apply -f -`

# Split traffic between v1 and v2

```
cat <<EOF | kubectl apply -f -
apiVersion: policy.linkerd.io/v1beta2
kind: HTTPRoute
metadata:
  name: web-split
  namespace: emojivoto
spec:
  parentRefs:
    - name: web-svc
      kind: Service
      group: core
      port: 80
  rules:
    - backendRefs:
        - name: web
          port: 80
          weight: 50
        - name: web-v2
          port: 80
          weight: 50
EOF
```

This may look complicated but essentially, cats the manifest and pipes this to the apply command

Security

Linkerd provides mTLS encryption out of the box. You can verify this by checking the Linkerd dashboard or using linkerd tap.

`linkerd viz tap -n emojivoto deploy/web`

This will now start to listen for traffic. If you click on one of the emojis on the website you will see traffic and here you will notice `tls=true`
ie: `rsp id=31:9 proxy=out src=10.244.0.60:59620 dst=10.244.0.58:8080 tls=true :status=200 latency=959µs`

Monitoring and Debugging

Linkerd Tap

We have just used this option linkerd viz tap to see the traffic flowing through your services in real-time.

```
linkerd viz tap -n emojivoto deploy/web
```

Linkerd Top

Use linkerd top to see the top-level metrics for your services.

```
linkerd viz top -n emojivoto deploy/web
```

```
press q to quit)
(press a/LeftArrowKey to scroll left, d/RightArrowKey to scroll right)

Source                Destination              Method      Path                                                       Count    Best   Worst    Last  Success Rate
web-85f6fb8564-cdrxg  emoji-788f84699-dzqfk    POST        /emojivoto.v1.EmojiService/FindByShortcode                    28   620µs     7ms     2ms       100.00%
web-85f6fb8564-cdrxg  emoji-788f84699-dzqfk    POST        /emojivoto.v1.EmojiService/ListAll                             3     1ms     4ms     4ms       100.00%
web-85f6fb8564-cdrxg  voting-7479ff64b6-qjlxw  POST        /emojivoto.v1.VotingService/VoteStuckOutTongueWinkingEye       2     2ms     4ms     2ms       100.00%
web-85f6fb8564-cdrxg  voting-7479ff64b6-qjlxw  POST        /emojivoto.v1.VotingService/Results                            2     2ms     3ms     3ms       100.00%
web-85f6fb8564-cdrxg  voting-7479ff64b6-qjlxw  POST        /emojivoto.v1.VotingService/VoteDoughnut                       1     1ms     1ms     1ms         0.00%
web-85f6fb8564-cdrxg  voting-7479ff64b6-qjlxw  POST        /emojivoto.v1.VotingService/VoteSunglasses                     1     2ms     2ms     2ms       100.00%
```

As you can see a great method to see the issues and traffic including issues.
Here you can see I have selected the VoteStuckOutTongueWinkingEye, VoteDoughnut and VoteSunglasses.

You will also see that the doughnut was not successful

Cleaning Up

When you're done, clean up the resources created:

# Delete the emojivoto application

```
curl --proto '=https' --tlsv1.2 -sSfL https://run.linkerd.io/emojivoto.yml | kubectl delete -f -
```

# Uninstall Linkerd

```
linkerd viz uninstall | kubectl delete -f -
```

```
linkerd uninstall | kubectl delete -f -
```

Troubleshooting Common Errors

Error: No Objects Passed to Apply: Ensure you run the following command first to install the CRDs:

```
linkerd install --crds | kubectl apply -f -
```

Then, proceed with the control plane installation
source k8s, source linkerd.

Additional Resources

- Linkerd Documentation: The official Linkerd documentation is a comprehensive resource.
- Linkerd Tutorials: The Linkerd tutorials provide hands-on guides for various scenarios.
- Minikube Documentation: The Minikube documentation can help you manage your local Kubernetes cluster.

5-Week Training Plan: Service Mesh, Kubernetes, and Related Technologies