Linkerd Service mesh CLI

CONCEPTS

The CLI only prints YAML: pipe it to kubectl apply -f - to change the cluster.

linkerd install CRDs, then control plane
↓ linkerd check
Control plane identity, destination, policy
↓ inject
Meshed pods proxy sidecar per pod
↓ viz install
Viz extension stat, tap, top, dashboard

Install order matters: CRDs first, then the control plane, then extensions such as viz.

Meshing is done at pod creation: restart workloads after annotating them.

INSTALL & UPGRADE

`linkerd check --pre`
Pre-flight cluster checks

`linkerd install --crds | kubectl apply -f -`
Install the CRDs

`linkerd install | kubectl apply -f -`
Install the control plane

`linkerd viz install | kubectl apply -f -`
Install the viz extension

`linkerd install --ha`
High-availability control plane

`linkerd install --set <key>=<val>`
Override a Helm chart value

`linkerd install -f <values.yaml>`
Install with a values file

`linkerd install --identity-trust-anchors-file <ca.crt>`
Supply your own trust anchor

CHECK & VERSION

`linkerd check`
Verify the control plane

`linkerd check --proxy -n <ns>`
Verify data-plane proxies

CHECK & VERSION (cont.)

`linkerd check --crds`
Only check the CRDs

`linkerd viz check`
Verify the viz extension

`linkerd version`
Client and server versions

`linkerd version --client`
Client version only

INJECT

`linkerd inject <file|dir|URL⟶>`
Annotate manifests for meshing

`kubectl get deploy <name> -o yaml | linkerd inject - | kubectl apply -f -`
Mesh a live deployment

`kubectl annotate ns <ns> linkerd.io/inject=enabled`
Auto-inject a whole namespace

`linkerd uninject <file⟶>`
Remove the injection annotation

DASHBOARD

`linkerd viz dashboard`
Open the web dashboard

`linkerd viz dashboard --port <n>`
Pick the local port

`linkerd viz dashboard --show url`
Print the URL only

METRICS

`linkerd viz stat deploy -n <ns>`
Success rate, RPS, latency

`linkerd viz stat <kind>/<name> -n <ns>`
Stats for one resource

`linkerd viz stat ns`
Stats per namespace

`linkerd viz stat deploy -A`
All namespaces

`linkerd viz stat deploy/<name> --from deploy/<src>`
Traffic from a source

`linkerd viz stat deploy --to deploy/<dst>`
Traffic to a destination

METRICS (cont.)

`linkerd viz stat deploy -t <window>`
Time window, 15s minimum

`linkerd viz edges deploy -n <ns>`
Show who talks to whom (mTLS)

`linkerd viz routes svc/<name> -n <ns>`
Per-route metrics

LIVE TRAFFIC

`linkerd viz tap deploy/<name> -n <ns>`
Stream live requests

`linkerd viz tap deploy/<name> --path <prefix>`
Filter by path prefix

`linkerd viz tap deploy/<name> --method <verb>`
Filter by HTTP method

`linkerd viz top deploy/<name> -n <ns>`
Top-like live table of requests

DEBUG & IDENTITY

`linkerd identity -n <ns> <pod>`
Show a pod's mTLS certificate

UNINSTALL

`linkerd viz uninstall | kubectl delete -f -`
Remove the viz extension

`linkerd uninstall | kubectl delete -f -`
Remove the control plane

GLOBAL FLAGS

`-n, --namespace <ns>`
Namespace of the resource

`--context <name>`
Kubeconfig context

`--kubeconfig <path>`
Kubeconfig file

RECIPES

Mesh a namespace and verify

```
kubectl annotate ns <ns>
linkerd.io/inject=enabled
kubectl rollout restart deploy -n
<ns>
linkerd check --proxy -n <ns>
linkerd viz stat deploy -n <ns>
```