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` **`viz check`**
Verify the viz extension

`linkerd` **`version`**
Client and server versions

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

DASHBOARD

`linkerd` **`viz dashboard`**
Open the web dashboard

METRICS

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

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

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

LIVE TRAFFIC

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

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

UNINSTALL

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

GLOBAL FLAGS

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