BUOYANT'S SERVICE MESH ACADEMY

The creators of LINKERD

Hands on with Linkerd Enterprise

Jason Morgan, Technical Evangelist



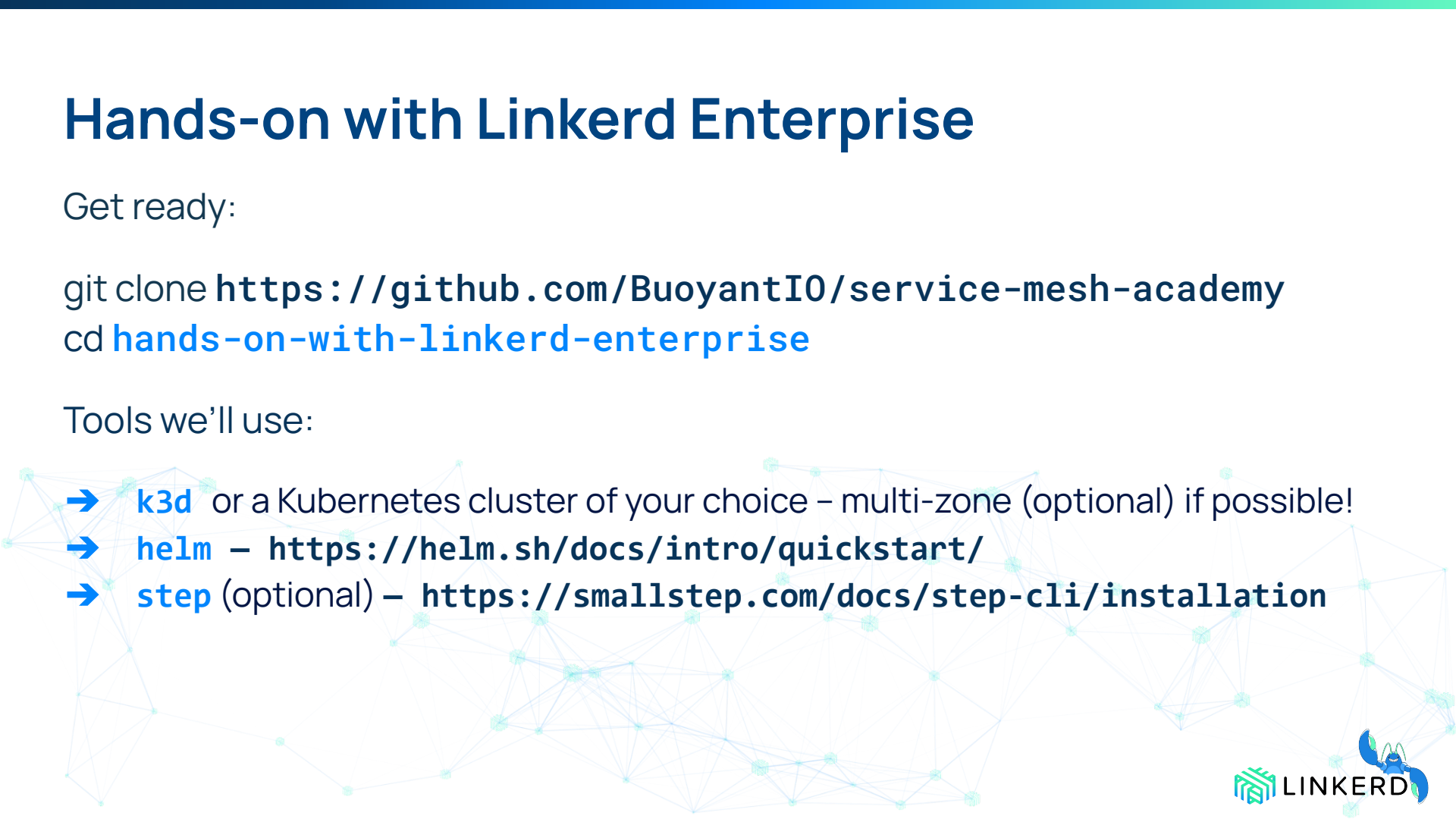
Hands-on with Linkerd Enterprise

Get ready:

git clone **`https://github.com/BuoyantIO/service-mesh-academy`**
cd **`hands-on-with-linkerd-enterprise`**

Tools we'll use:

- ➔ **`k3d`** or a Kubernetes cluster of your choice – multi-zone (optional) if possible!
- ➔ **`helm`** – **`https://helm.sh/docs/intro/quickstart/`**
- ➔ **`step`** (optional) – **`https://smallstep.com/docs/step-cli/installation`**



Today's agenda



1. Welcome!
2. Linkerd Enterprise vs Open Source
3. Feature overview
4. Hands-on time!

LINKERD

BUOYANT'S

SERVICE MESH ACADEMY

The creators of LINKERD

Monthly hands-on, engineer-focused training from the creators of the service mesh

February: Beyond Kubernetes with Linkerd 2.15 and SPIFFE

SIGN UP TODAY!
buoyant.io/sma



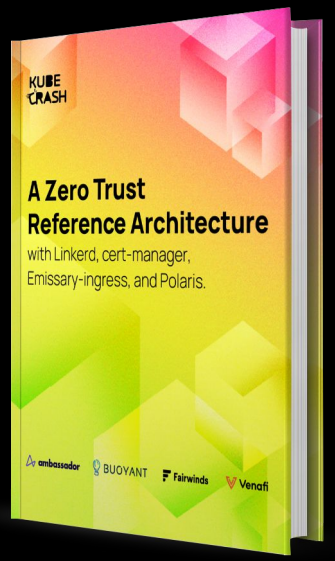
A Zero Trust Reference Architecture

with Linkerd, cert-manager, Emissary-ingress, and Polaris

60+ pages of technical, open source content!

buoyant.io/**download/zero-trust-reference-architecture**

KUBE CRASH



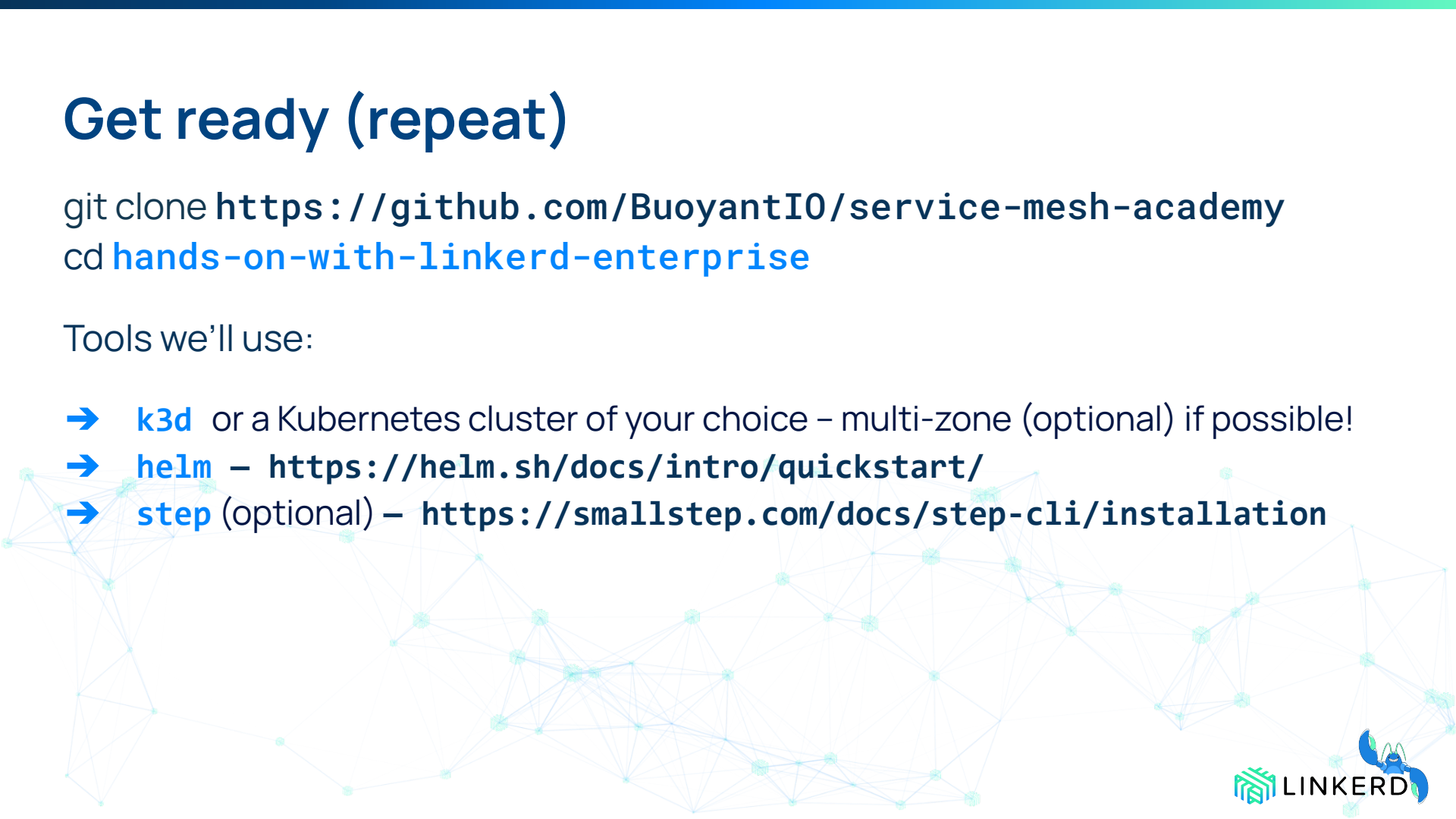
BUOYANT

Get ready (repeat)

git clone **`https://github.com/BuoyantIO/service-mesh-academy`**
cd **`hands-on-with-linkerd-enterprise`**

Tools we'll use:

- **`k3d`** or a Kubernetes cluster of your choice – multi-zone (optional) if possible!
- **`helm`** – **`https://helm.sh/docs/intro/quickstart/`**
- **`step`** (optional) – **`https://smallstep.com/docs/step-cli/installation`**



What is Linkerd Enterprise?

Linkerd Enterprise is everything in OSS Linkerd, plus:

- Enterprise-hardened images. Continuously scanned and tested at high scale.
- SBOMs. Software Bills of Materials detailing provenance of each component.
- Strict SLAs around CVE remediation. Guaranteed fixes with timelines.

And plus some enterprise-only features:

- FIPS-140-2/3 compliance. When enabled, conform to NIST specifications.
- Lifecycle automation. Fully automated installs, upgrades, and rollbacks.
- HAZL. High-availability zonal load balancing. Optimize routing to avoid cross-AZ charges, while still preserving reliability.
- Security policy generation. Generate zero-trust security policies from existing application traffic.

LINKERD

Free trial mechanics

In this trial you get full access to all Linkerd Enterprise features!

- Trial works on any Kubernetes cluster
- You get full* functionality of Buoyant Enterprise for Linkerd
- Trial lasts for two weeks (length can be extended, just contact us)
- Includes Buoyant Cloud, HAZL, policy generation, and more!

(* except FIPS compliance)



Enterprise feature: Lifecycle automation

Linkerd upgrades are easy, but they can be even easier! A Kubernetes Operator that:

- Automates Linkerd installations on new clusters
- Automated Linkerd upgrades, including across both control & and data planes
- Is fully GitOps compatible and works with Argo, Flux, and Helm

👉 Save developer time, reduce risk, and improve security by automating upgrades.



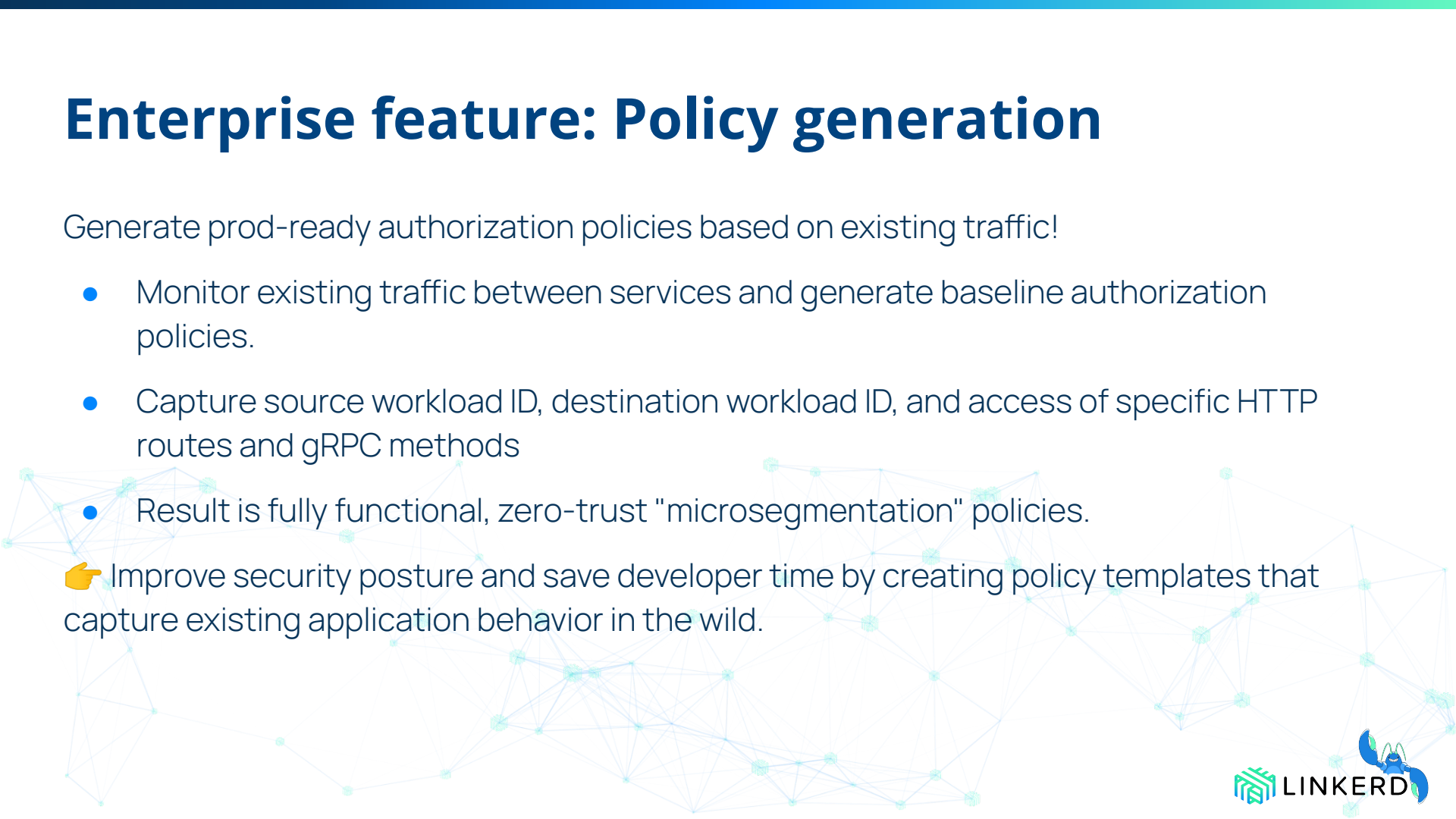
LINKERD

Enterprise feature: Policy generation

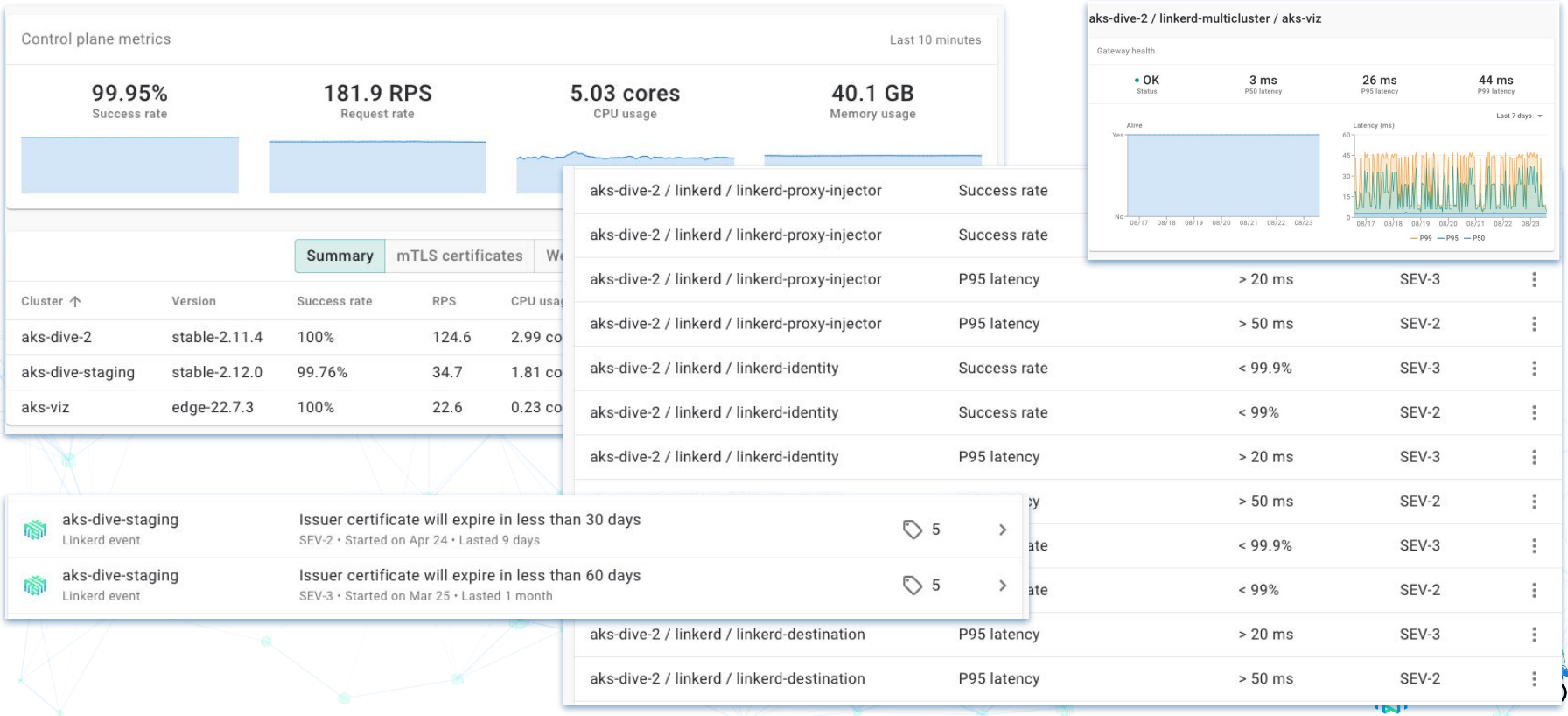
Generate prod-ready authorization policies based on existing traffic!

- Monitor existing traffic between services and generate baseline authorization policies.
- Capture source workload ID, destination workload ID, and access of specific HTTP routes and gRPC methods
- Result is fully functional, zero-trust "microsegmentation" policies.

👉 Improve security posture and save developer time by creating policy templates that capture existing application behavior in the wild.



Enterprise feature: Buoyant Cloud



Enterprise feature: HAZL

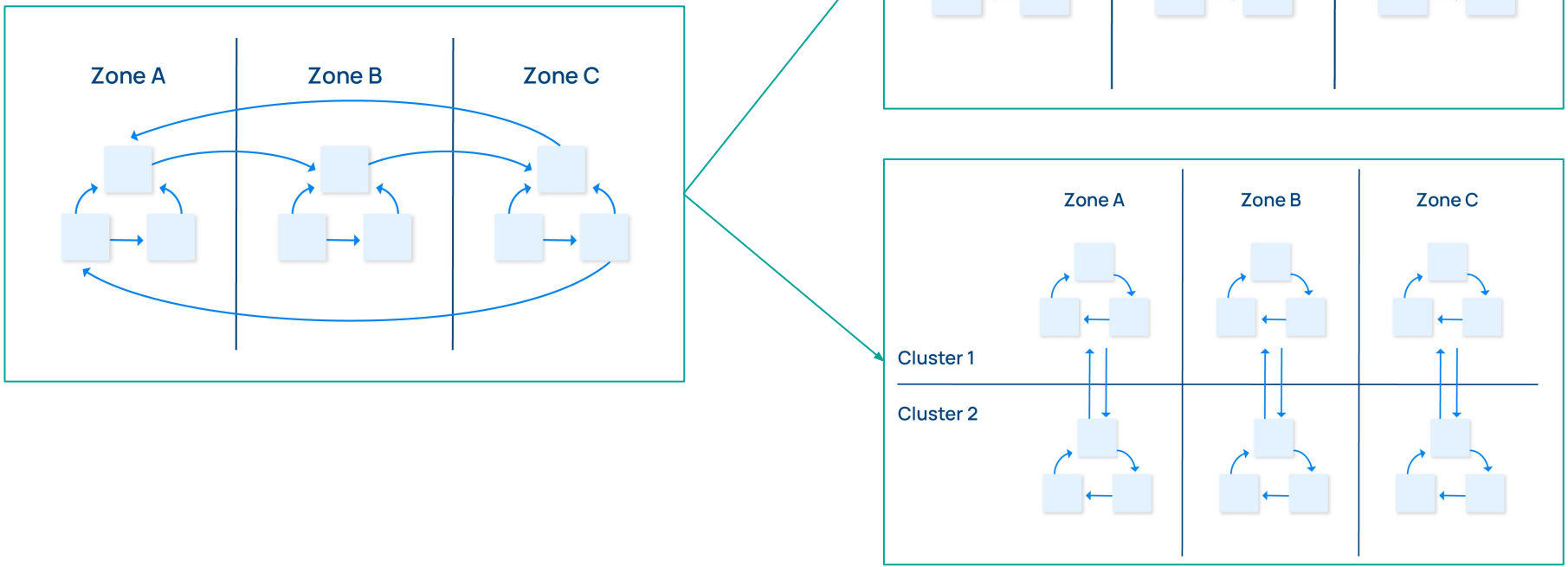
High **A**vailability **Z**onal **L**oad balancer: a powerful zone-aware load balancer that dynamically reduces traffic across zone/region boundaries while preserving HA guarantees.

Why?

- Many enterprises deploy multi-AZ (availability zone) clusters.
- Traffic across zones incurs non-trivial cloud spend!
- Topology Hints can reduce cost but sacrifices availability!

👉 Immediately cut 50% or more of your cross-AZ cloud spend. For some customers that is millions of dollars a month!




Zone A
Zone B
Zone C
Zone A
Zone B
Zone C
Zone A
Zone B
Zone C
Cluster 1
Cluster 2

HAZL vs Topology Hints

Kubernetes's Topology-aware routing (formerly Topology Hints) provides a simple mechanism for reducing cross-zone traffic. However, it is a *static* approach with significant drawbacks:

- Allocates endpoints at L4, the filters the endpoints returned to clients. This means cross-zone traffic is *prevented*—no effective HA capabilities.
- Cannot dynamically expand pool as traffic increases or as endpoints start to fail.
- Assumes even distribution of traffic. Services easily overloaded if that isn't the case.
- Does not work with tolerations, autoscaling, or other important K8s features.



Let's do it!

enterprise.buoyant.io/start_trial

