



Emissary + Linkerd Resilience Patterns

Rate Limits, Retries & Timeouts



Flynn

Tech Evangelist @ Buoyant



✉️ flynn@buoyant.io
/github/ [@kflynn](#)
/slack/ [@flynn](#)

Daniel Bryant

Head of DevRel @ Ambassador Labs



✉️ db@datawire.io
/github/ [@danielbryantuk](#)
/slack/ [@danielbryantuk](#)

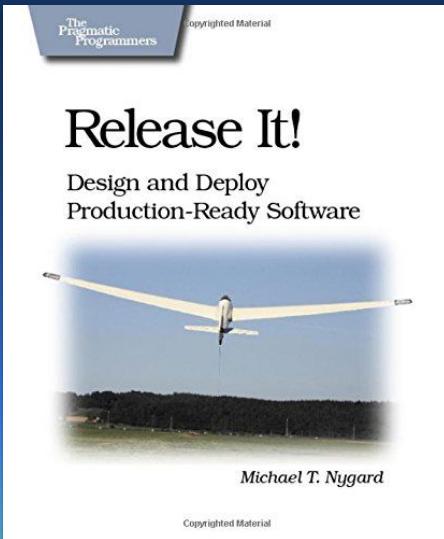
RESILIENCE



Requirements

- Users want a reliable experience
- Resilience must be end-to-end
- Must handle N/S and E/W Traffic
- Dev/Ops self-service is needed

RESILIENCE

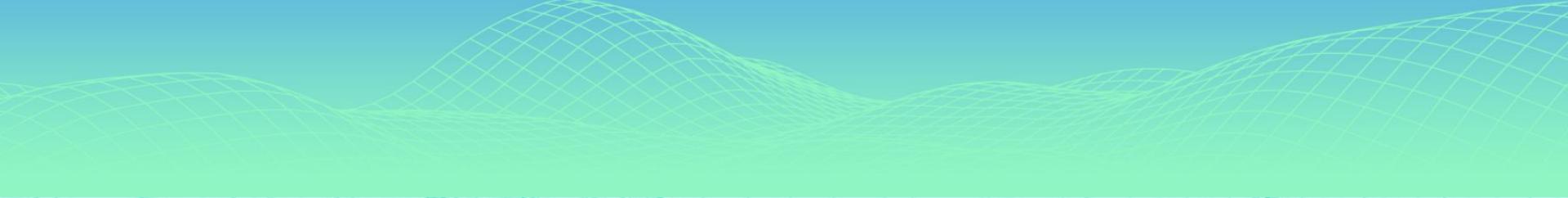


Patterns

- Retries
- Timeouts
- Rate limits

- Bulkheads
- Circuit-breakers

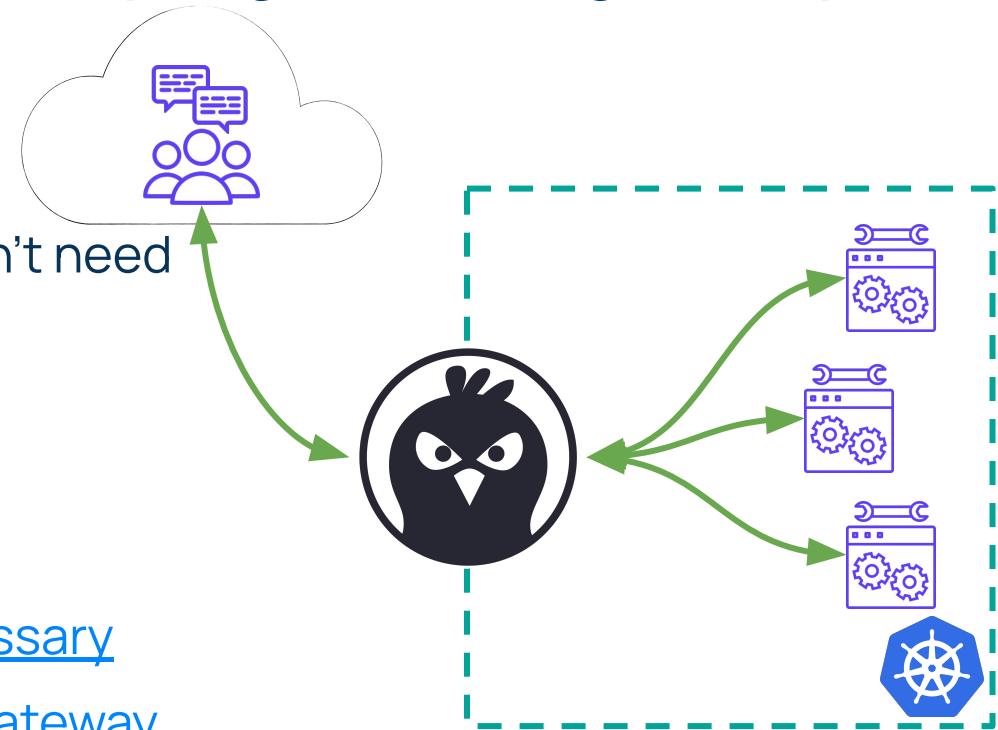
Emissary-ingress Intro



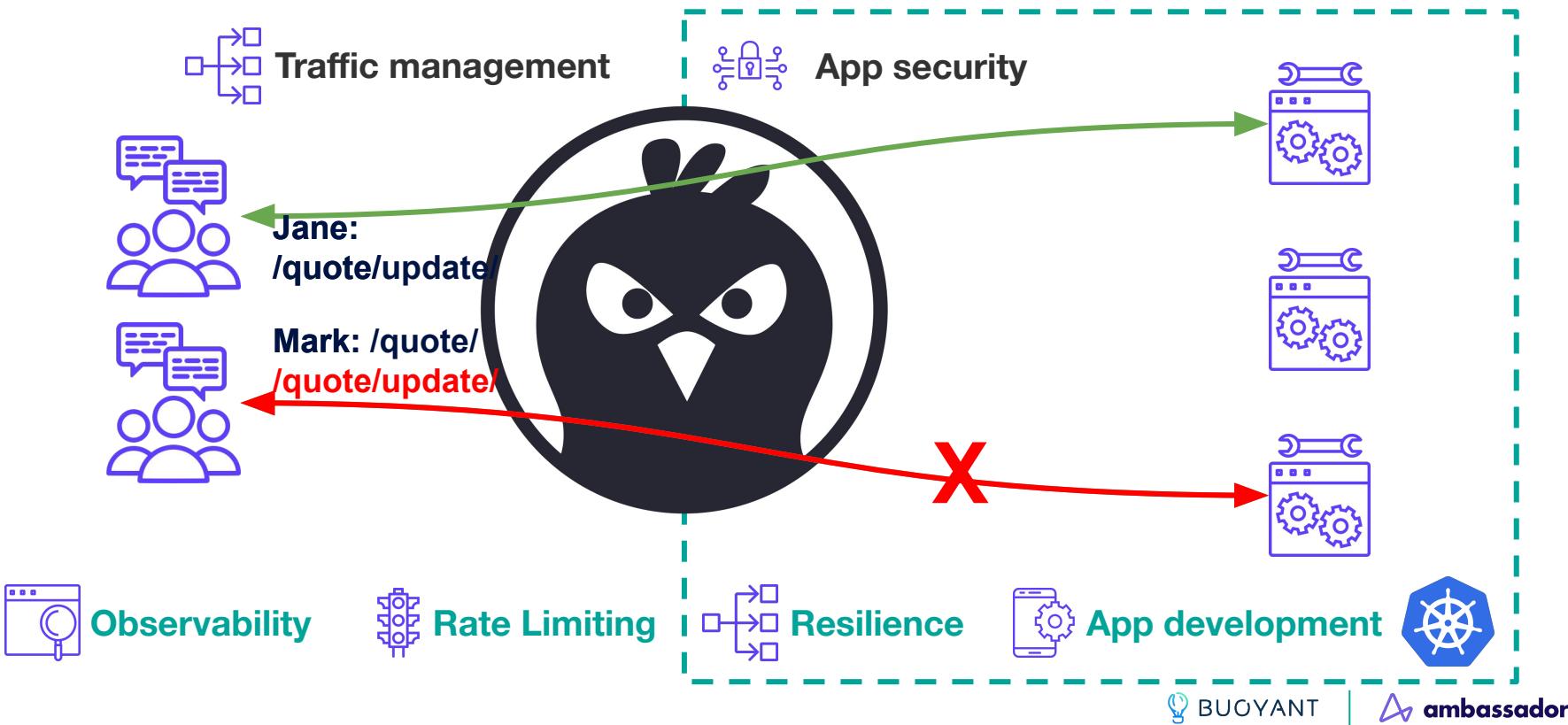
North-south traffic: Emissary-ingress (API gateway)

- Open source & cloud-native
- CNCF Incubating project
- Powered by Envoy, but you don't need to be an Envoy expert!
- Wide adoption since 2017
- Focus on self-service

github.com/emissary-ingress/emissary
getambassador.io/products/api-gateway



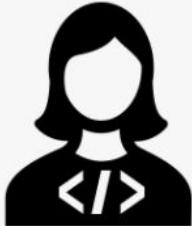
North-south traffic: Emissary-ingress



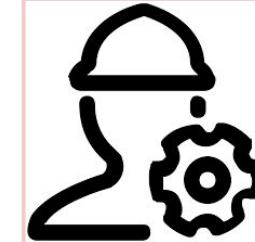


Focus on Self-Service

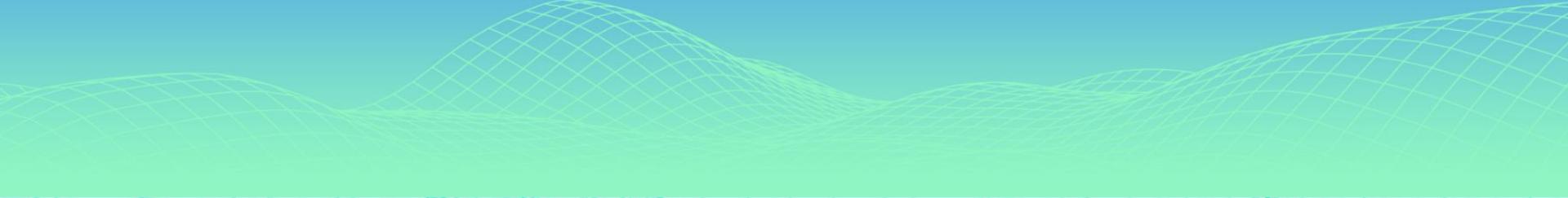
```
apiVersion: getambassador.io/v3alpha1
kind: Mapping
metadata:
  apiVersion: getambassador.io/v3alpha1
spec kind: Mapping
metadata:
  n  apiVersion: getambassador.io/v3alpha1
spe  kind: Mapping
p  metadata:
  s    name: rate-limited-mapping
w  spec:
t    host: high-traffic.example.com
    prefix: /high-traffic/
    service: popular-service
  - bels:
      "emissary":
        - "group1":
          - remote_address:
              key: remote_address
```



```
apiVersion: getambassador.io/v3alpha1
kind: Listener
metadata:
  apiVersion: getambassador.io/v3alpha1
spec kind: Host
metadata:
  n  ---
  l  apiVersion: getambassador.io/v3alpha1
     kind: RateLimitService
  spe  metadata:
    h    name: ratelimit
    t  spec:
      service: 'rate-limiter.default:5000'
      protocol_version: v3
      re_mode_deny: false
```



Linkerd Intro





Ultralight, ultrafast, security-first **service mesh** for Kubernetes.

- ★ Created by **Buoyant**
- ★ 5+ years in production
- ★ 7,000+ Slack channel members
- ★ 10,000+ GitHub stars
- ★ 200+ contributors
- ★ Weekly edge releases
- ★ Open governance, neutral home
- ★ The only **CNCF graduated** mesh



ATTEST



COMCAST

Clover

ebay

EVERQUOTE



GoSpotCheck



Ladbrokes



Mercedes-Benz



NORDSTROM

OfferUp

Ottawa

planet.

STRAVA

S U B S P A C E

transit

Walmart



and many more...





Ultralight, ultrafast
mesh for Kuber

- ★ Create
- ★ 5+
- ★ 7,
- ★ 10
- ★ 200
- ★ Weekly
- ★ Open govern
- ★ The only CNCF grad

Linkerd is a service mesh.

Its purpose in life is to give every developer the tools they need to create **secure, reliable, observable** cloud-native applications.



ATTEST

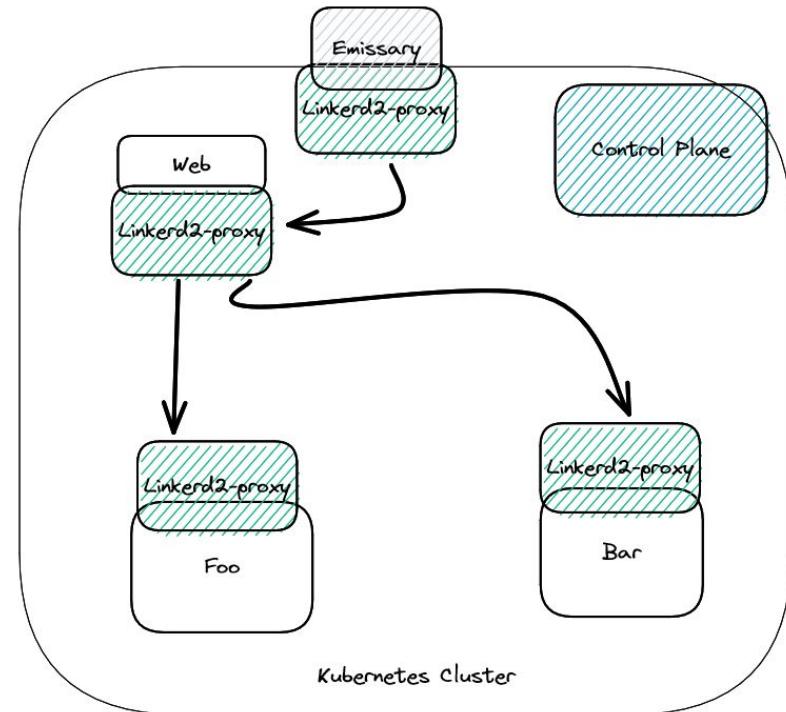


and many more...



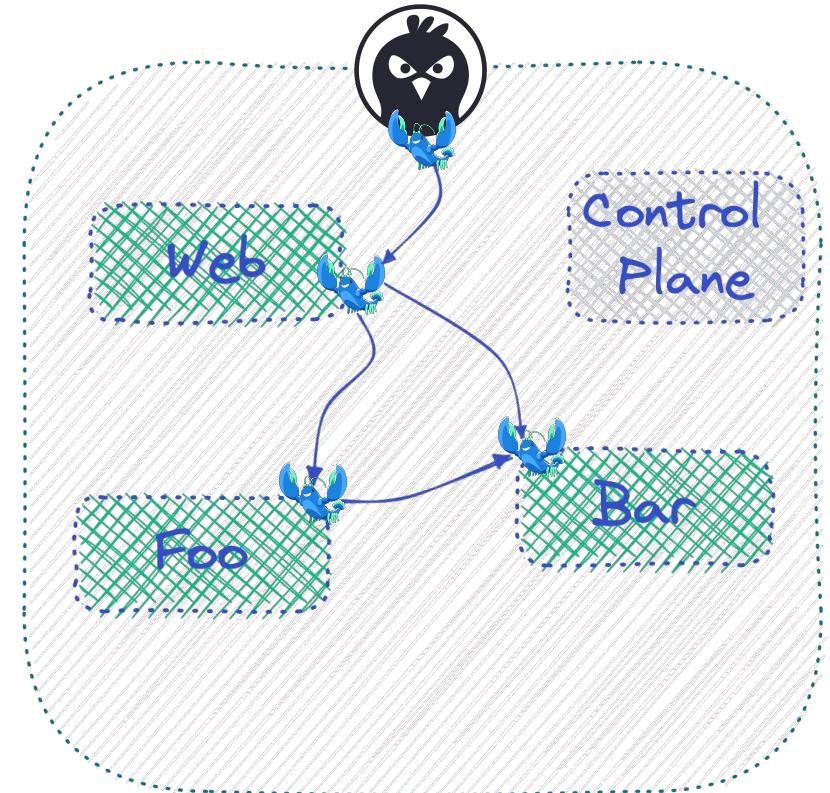
How do they work together

- Easily!
 - ◆ Emissary behavior mirrors k8s behavior
 - ◆ Makes the integration simple
- Via Proxies
- Let's check it out



How do they work together

- Easily!
 - ◆ Linkerd injects proxies next to each workload
 - ◆ Emissary is just another workload to Linkerd



Linkerd solves production problems in Kubernetes

"It solved our problems **including some of those we didn't realize we had.**" — Steve Gray, Head of Trading Solutions, Entain

"We desperately needed insights into what was happening in the cluster and the new microservices architecture. **Without the observability that Linkerd gave us, it would have been difficult, if not impossible, to find the source of the problem.**" — Henry Hagnäs, Cloud Solution Architect, Elkjøp

"Can't imagine building Kubernetes-based distributed systems without Linkerd! Nobody got time for building gRPC load balancing, mTLS, automatic retries and timeouts from scratch. Plus... IT. JUST. WORKS." — Jeremy Gordon, CTO, Projector

Linkerd fills gaps in *operation* and *security* of Kubernetes clusters:

- ★ **Observability**: no visibility into traffic on the cluster.
- ★ **Reliability**: no handling of partial or temporary failures.
- ★ **Security**: no real security of data in transit.

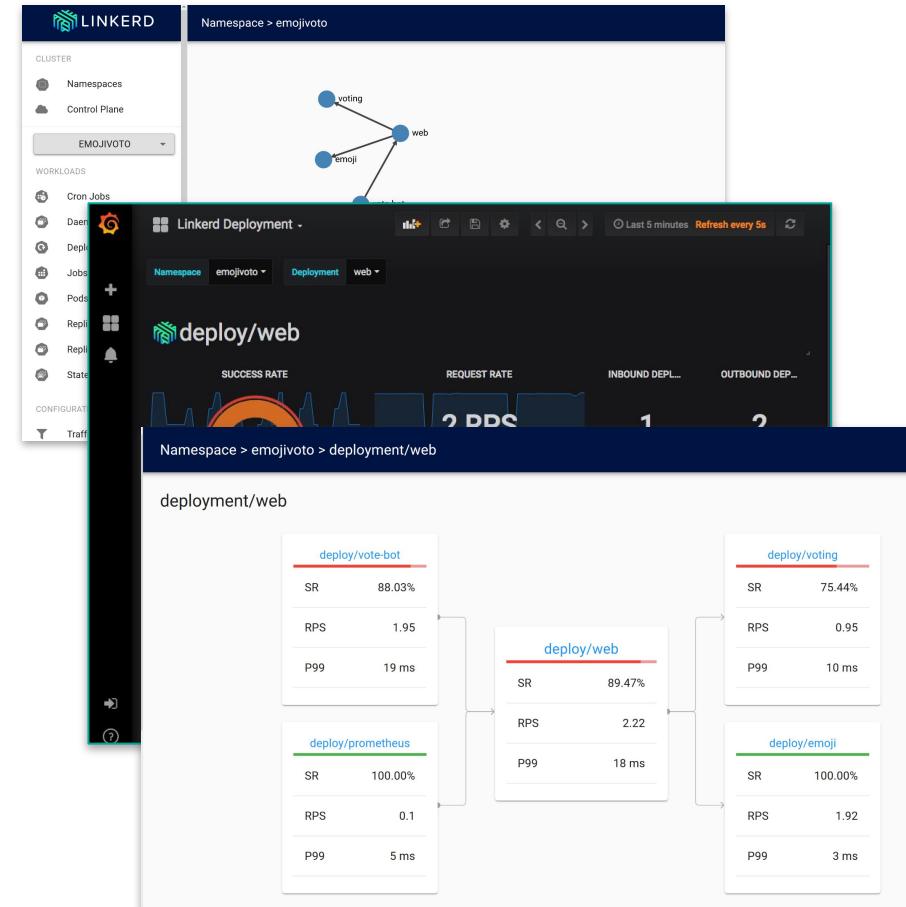
How does Linkerd do this?

→ **Observability:** Service-level *golden metrics*: success rates, latencies, throughput. Service topologies.

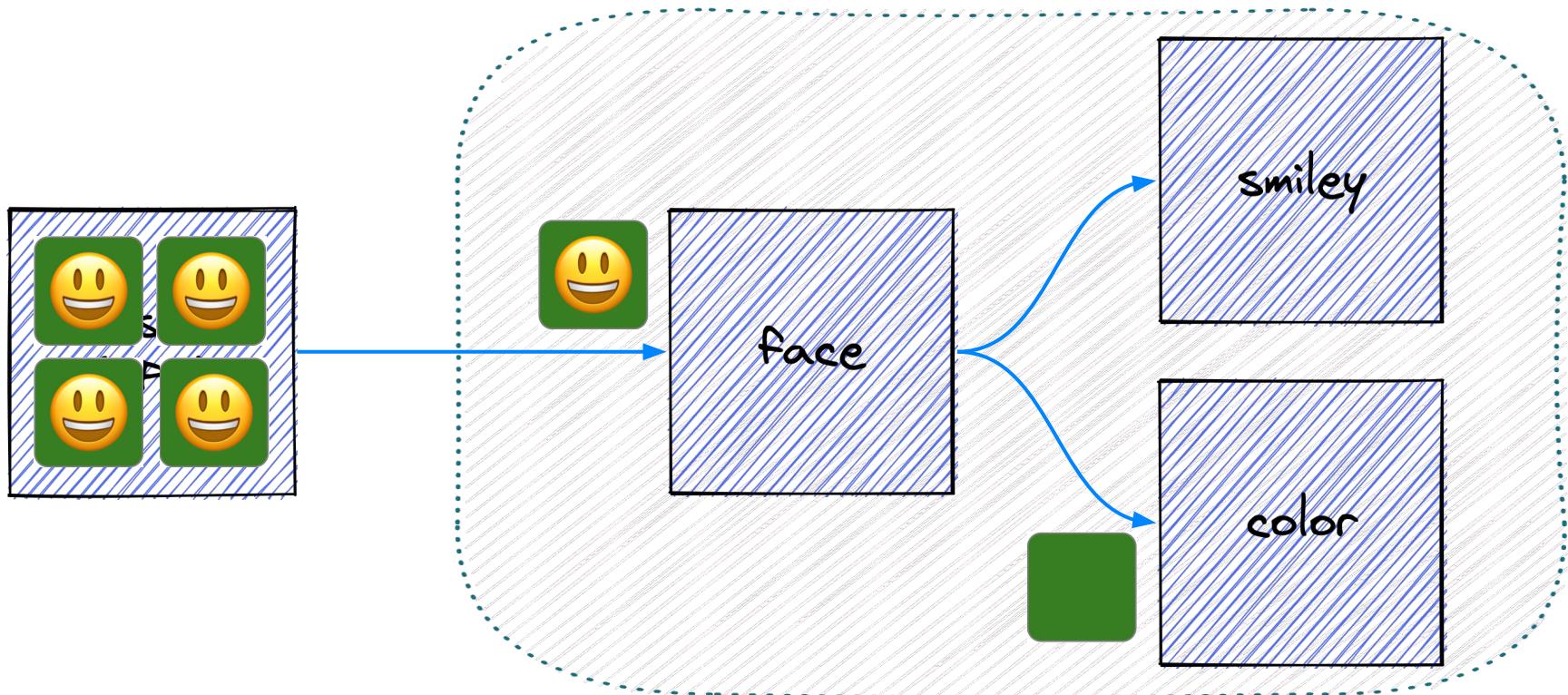
→ **Reliability:** Retries, timeouts, load balancing

→ **Security:** Transparent mTLS, cert management and rotation, policy

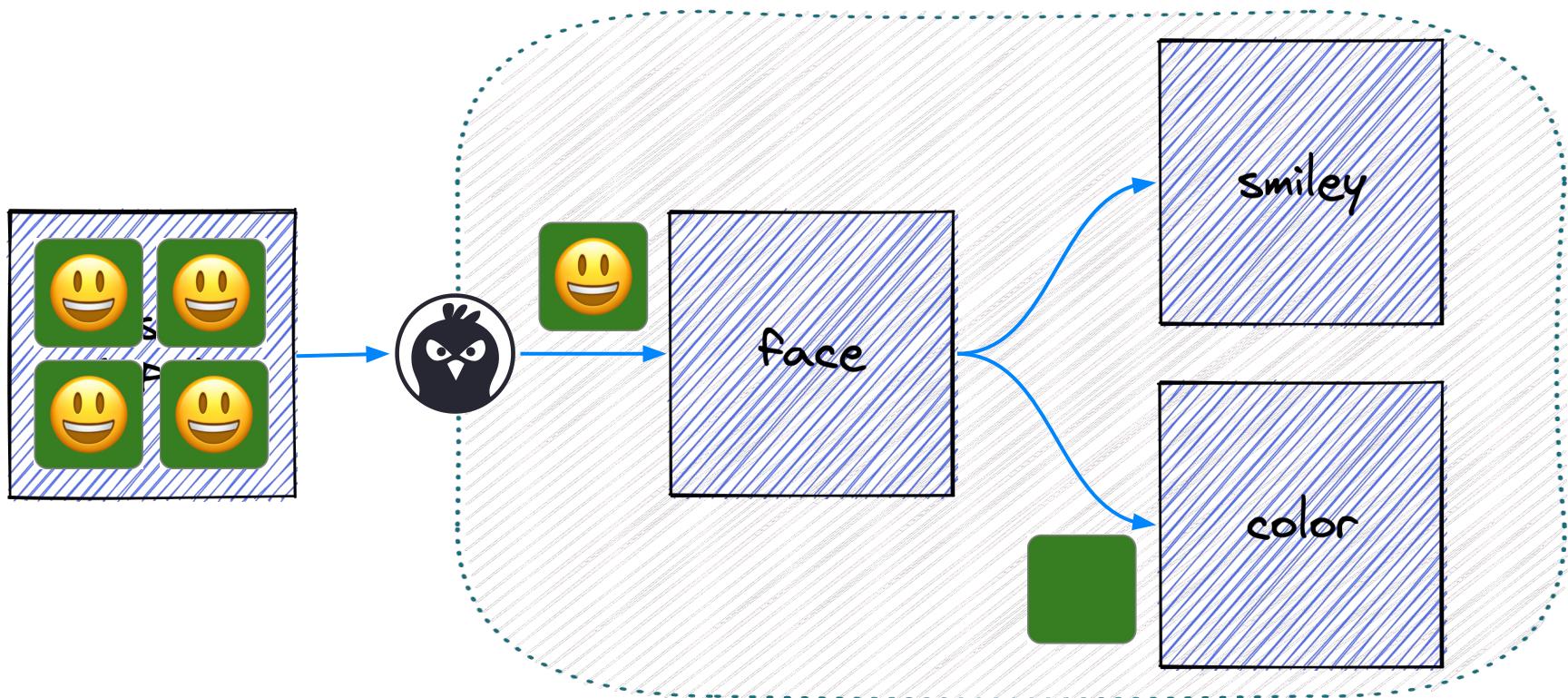
In an ultralight package focused on **simplicity** and **security** first.



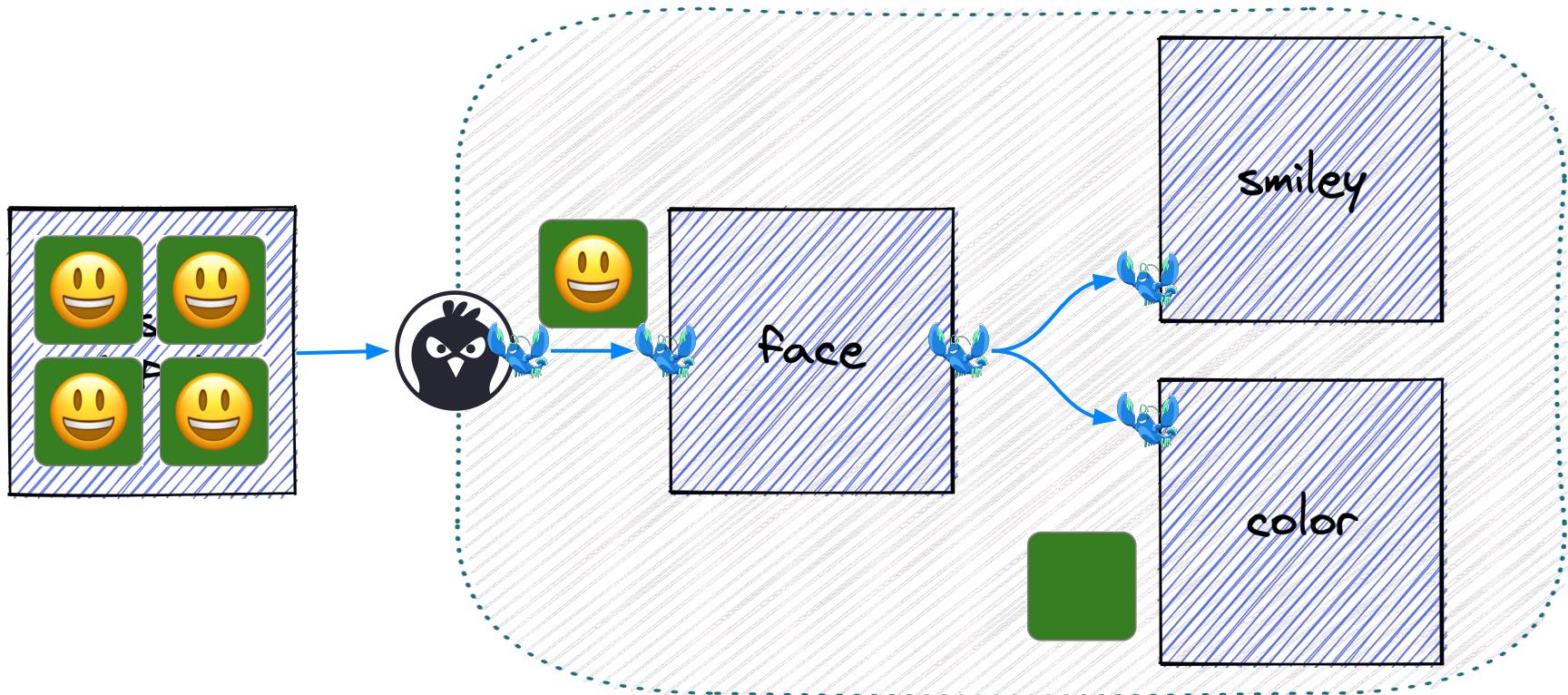
Demo Architecture



Demo Architecture



Demo Architecture



What you should always see



BUOYANT

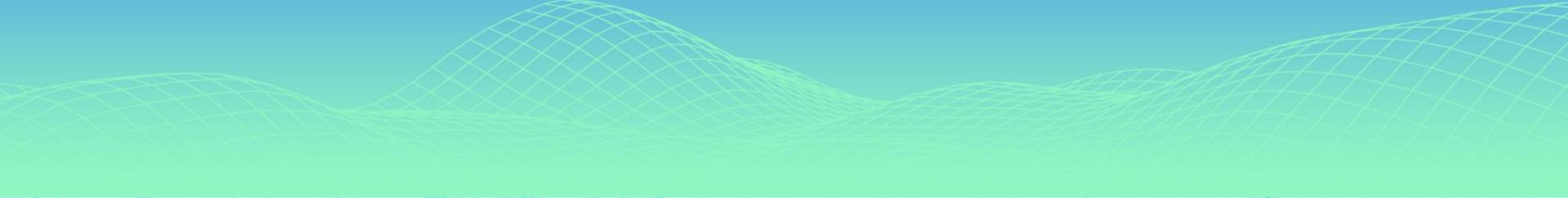


ambassador

What you might see



Demo Time!



Summary

- Users want reliable software - they don't care about the internal details
 - ◆ You need to think about end-to-end reliability and resilience
- Cloud native communications is a good place to start
 - ◆ Think north/south and east/west, or...
 - ◆ Think Emissary-ingress and Linkerd 😊
- A mix of retries, timeouts, and rate limits goes a long way
 - ◆ Make sure your solution is developer-focused and self-service

References

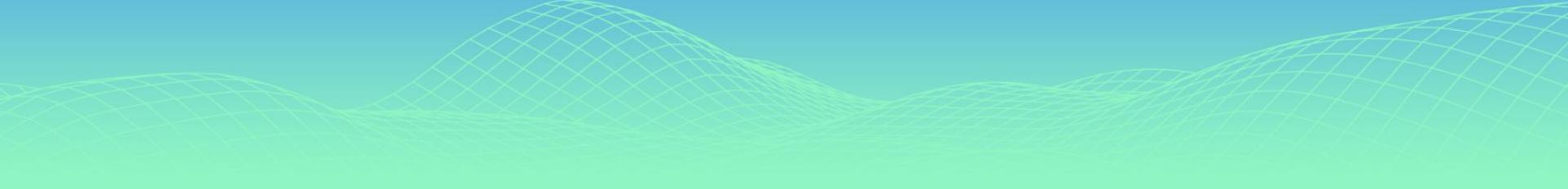
<https://github.com/BuoyantIO/faces-demo/tree/main/faces>

<https://github.com/ambassadorlabs/ratelimit-example>

<https://linkerd.io/2.12/overview/>

<https://www.getambassador.io/docs/emissary/>

Questions?





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

Nov 11-17: What really happens at startup: a deep dive into Linkerd, init
containers, CNI plugins, and more



SIGN UP TODAY!

buoyant.io/sma

K8s in 8

CHALLENGE



Take the challenge for a chance to win an iPhone 14, a Pluralsight subscription, and other prizes.

Complete **8 simple challenges** to learn how you can use Ambassador Cloud to Code, Ship, and Run your Kubernetes (K8s) services

- **Make a local code change** against a remote K8s cluster and share a preview with your team
- **Create an Edge Stack Mapping** to ship your code and publish your API externally
- **Add K8s annotations** to document critical run-time metadata within your team's Service Catalog
- **And everything else** needed to build faster and ship safer as a team with K8s

GET STARTED

getambassador.io/kubernetes-learning-center/k8s-challenge/

Fully managed LINKERD on any Kubernetes cluster

Buoyant Cloud automated upgrades, data plane version tracking, mesh health alerts, and much, much more.

BOOK A DEMO
buoyant.io/demo



Kubernetes Expert Interviews: Adopting K8s as a Team

Accelerating the Kubernetes Journey for Cloud Native Organizations

Perspectives on the platforms, paved paths, and developer experience for organizations adopting Kubernetes



Learn from Ambassador Experts

- [Use Your Own Tools: An Ambassador Engineering Manager's View](#)
- [Use Your Own Tools: An Ambassador VP of Engineering's View](#)
- [Use Your Own Tools: A Principal Engineer's Perspective](#)
- [Enabling and Measuring Developer Productivity at Ambassador Labs](#)
- And more will be published soon!

getambassador.io/kubernetes-expert-interviews/



Thanks so much!

