
KubeCon
CloudNativeCon
North America 2019


KubeCon
CloudNativeCon
North America 2019
Enforcing Automatic mTLS with Linkerd and OPA Gatekeeper
Ivan Sim, Buoyant & Rita Zhang, Microsoft
LINKERD
Open Policy Agent

Agenda

- What this talk is about
- About us
- Why use Linkerd for mTLS
- Automatic mTLS with Linkerd
- Policies enforcement with Gatekeeper during workload admission
- Closing Thoughts
- Q&A

Problem Statement

How do I encrypt and authenticate east-west traffic between my services?

Security Factors To Consider

- Image Scanning
- Network Policies
- DNS Rebinding
- **Mutual Transport Layer Security**

About Us

Ivan Sim (@ihcsim)

Software Engineer, Buoyant

Linkerd Contributor

Rita Zhang (@ritazzhang)

Software Engineer, Microsoft

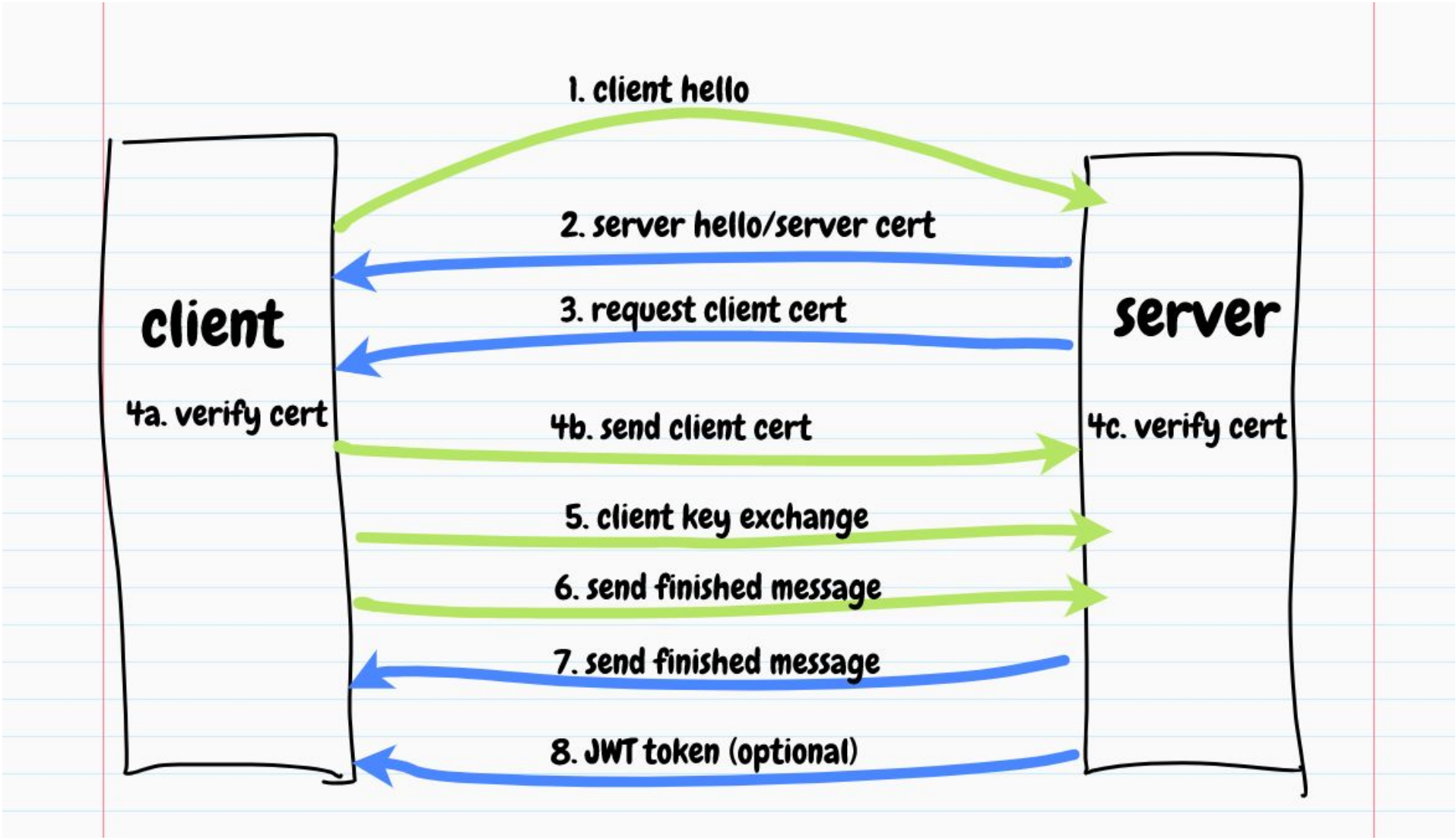
Gatekeeper Maintainer

What is mTLS?

Mutual Transport Layer Security (mTLS) - two entities confirm each other's identity using TLS server/client certificates

- Not to be confused with *multiplexed* TLS
- TLS RFC - *The TLS Handshake Protocol*

Mutual Transport Layer Security



Problem Background

- Managing the certificates
- Getting the configurations right
- Getting everyone onboard
- Maintaining home-grown internal Certificate Authority solution
  - Vulnerability within CA solution

Automatic mTLS with Linkerd

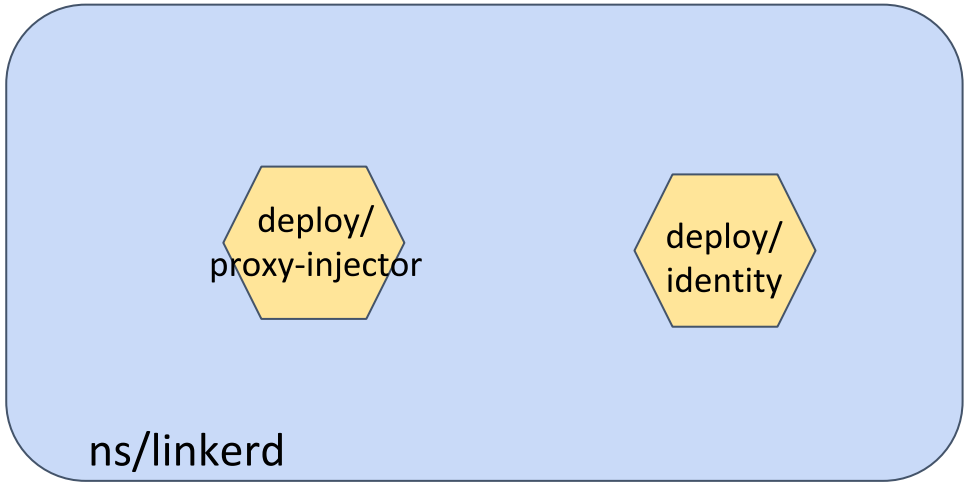
- **Secure**
  - Passed the PEN test and code security audit
  - https://github.com/linkerd/linkerd2/blob/master/SECURITY_AUDIT.pdf
- **Consistent**
  - mTLS enabled by default across the mesh
- **Scalable**
  - 3500 injected pods in a user's production environment

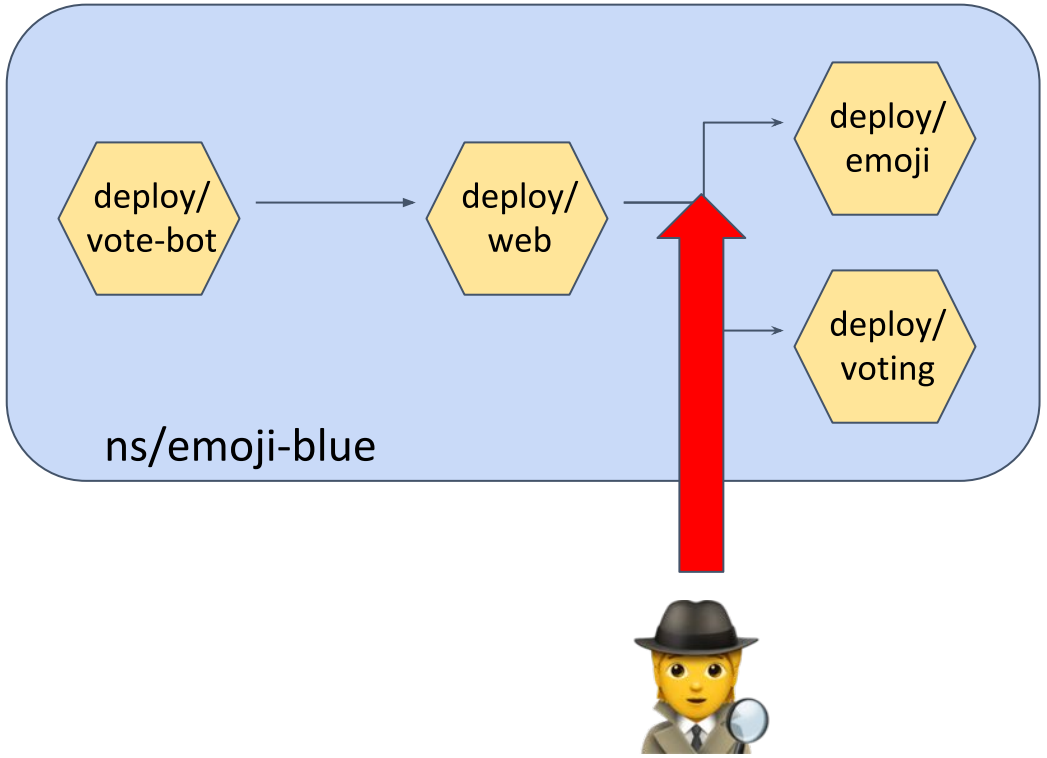
What is Linkerd?

An open source service mesh for Kubernetes

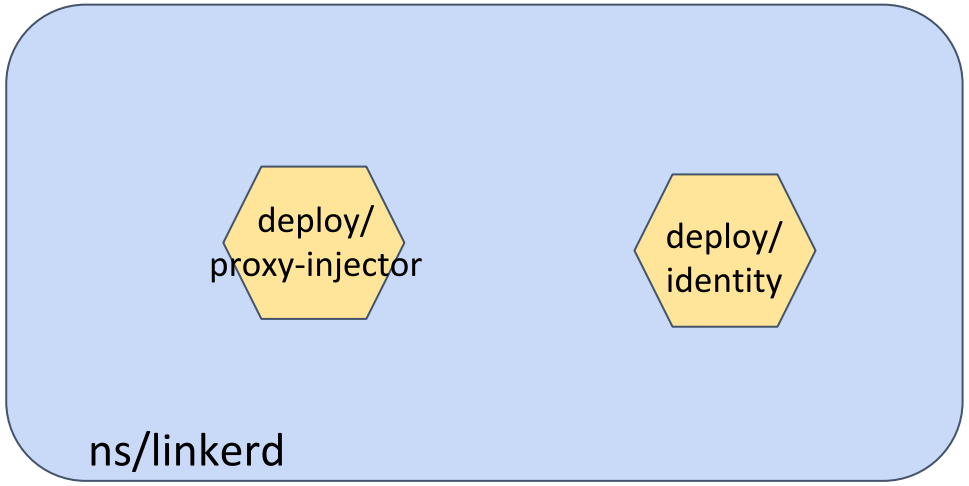
- **Observability**: Service-level golden metrics: success rates, latencies, throughput. Service topologies
- **Reliability**: Retries, timeouts, load balancing, traffic split
- **Security**: Transparent mTLS, cert management and rotation

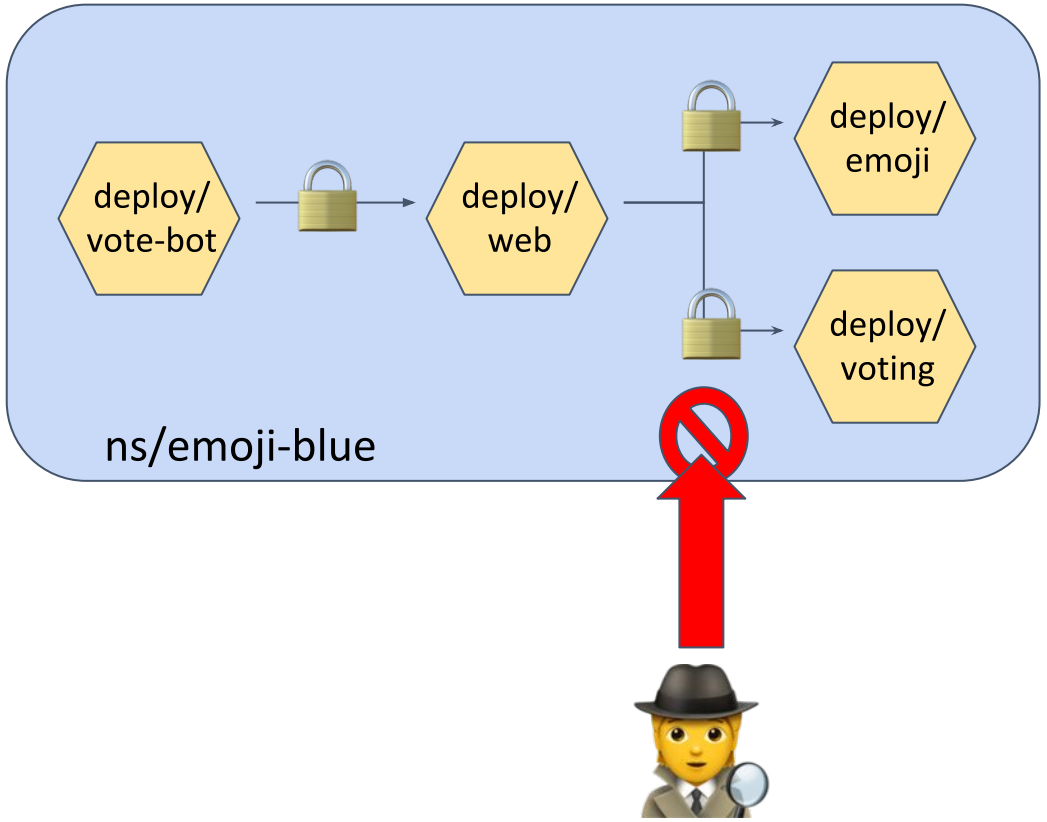
Demo 1 - Automatic mTLS with Linkerd





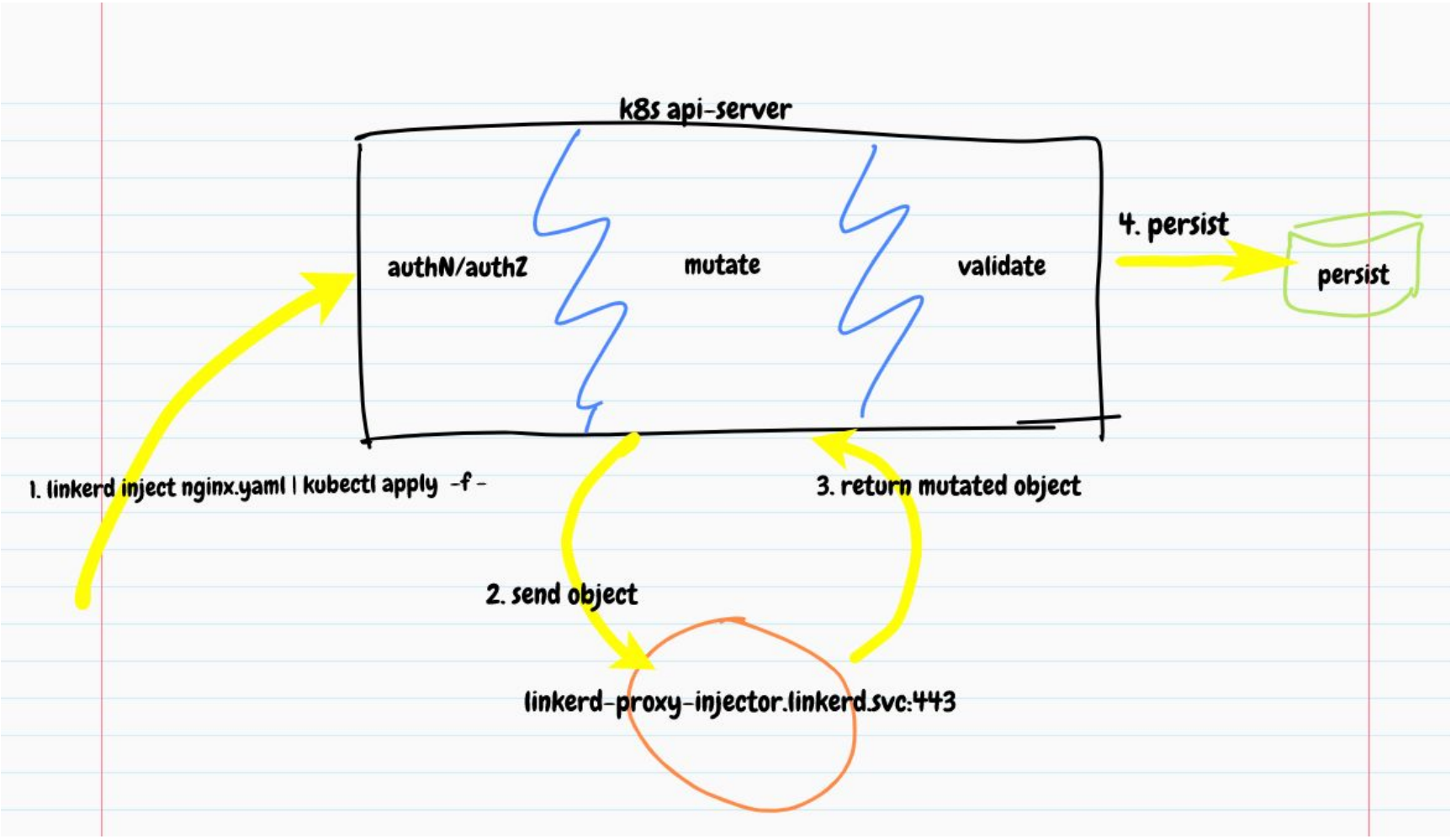
Demo 1 - Automatic mTLS with Linkerd





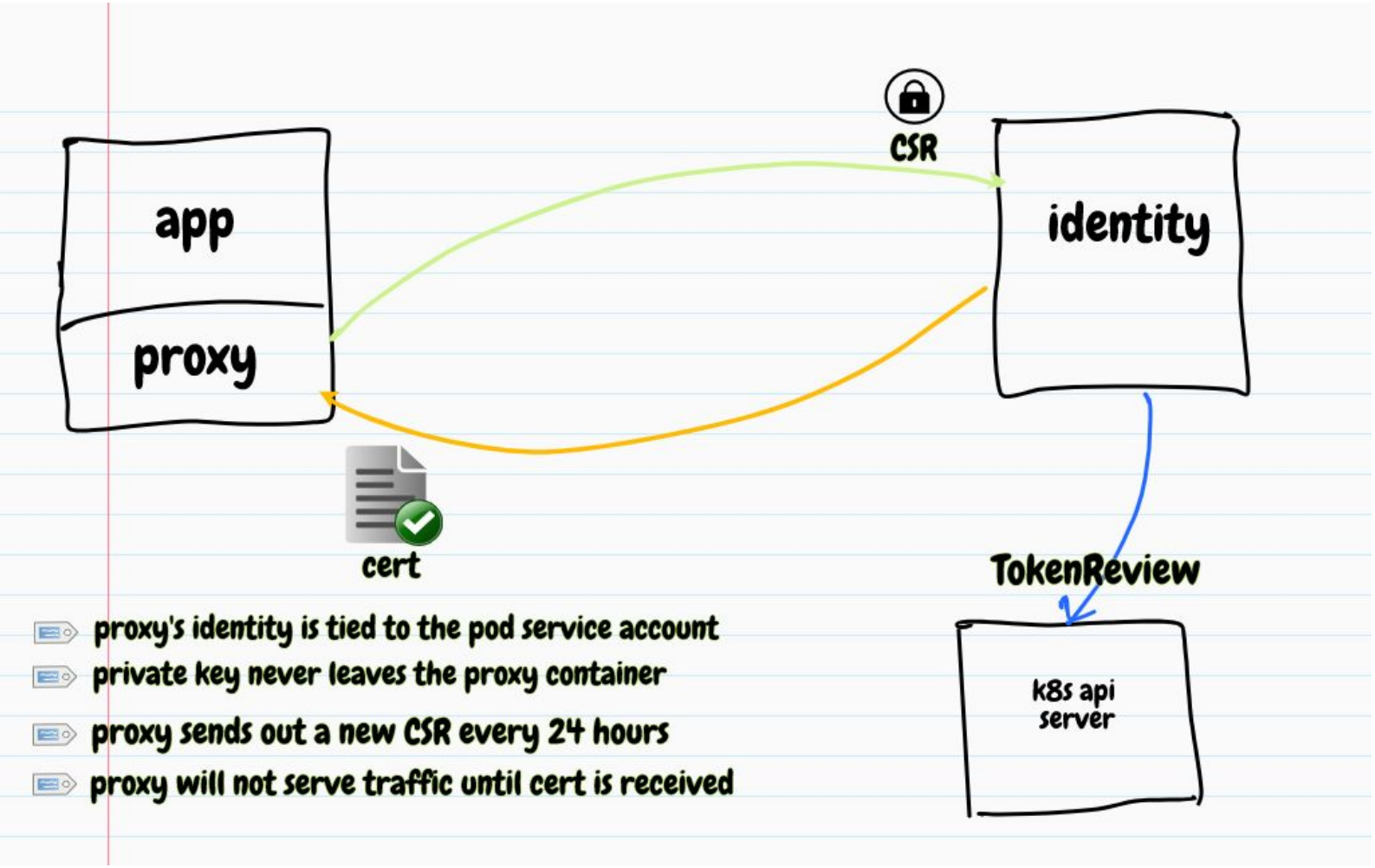
Inject Linkerd Proxy

The proxy injector injects the proxy into the workload



Signing Proxy CSR

The identity service issues TLS certificate to the proxy



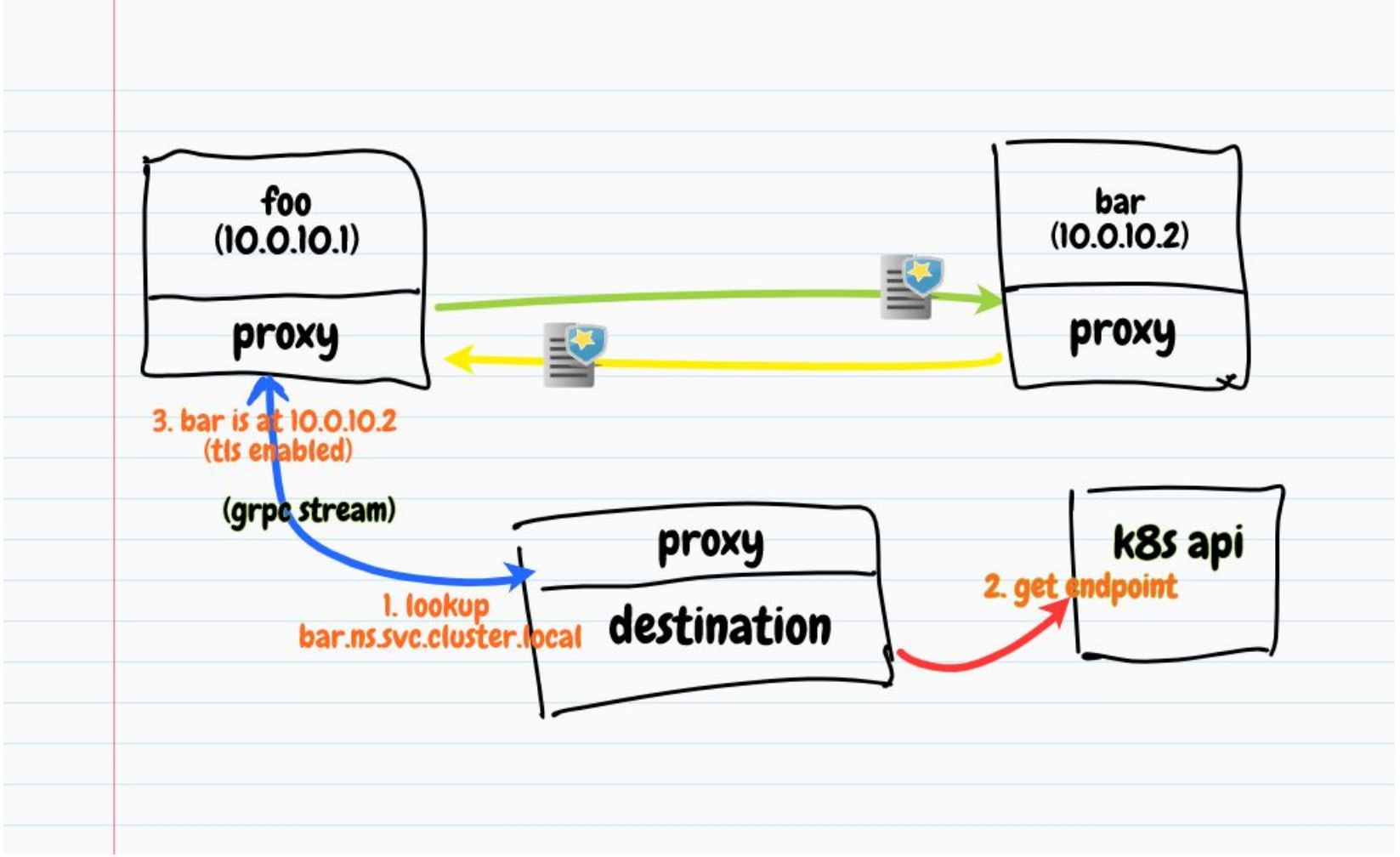
Issue Proxy Identity

Proxy identity information is read from env vars

```
$ linkerd inject nginx.yaml --manual | grep -A1 "LINKERD2_PROXY_IDENTITY_*
deployment "nginx" injected
…
    env:
          # trust root
          - name: LINKERD2_PROXY_IDENTITY_TRUST_ANCHORS
            value: |

          # proxy's identity
          - name: LINKERD2_PROXY_IDENTITY_LOCAL_NAME
            value: $(_pod_sa).$(_pod_ns).serviceaccount.identity.$(_l5d_ns).$(_l5d_trustdomain)

          # service account token file location
          - name: LINKERD2_PROXY_IDENTITY_TOKEN_FILE
            value: /var/run/secrets/kubernetes.io/serviceaccount/token

          # identity service endpoint
          - name: LINKERD2_PROXY_IDENTITY_SVC_ADDR
            value: linkerd-identity.linkerd.svc.cluster.local:8080

          # identity service's identity
          - name: LINKERD2_PROXY_IDENTITY_SVC_NAME
            value: linkerd-identity.$(_l5d_ns).serviceaccount.identity.$(_l5d_ns).$(_l5d_trustdomain)
```

Discover And Connect to other Services

The proxy discovers other services via the destination service



Enforce mTLS for Kubernetes Apps

How do we define and enforce mTLS policies

OPA Gatekeeper

A customizable Kubernetes admission webhook that

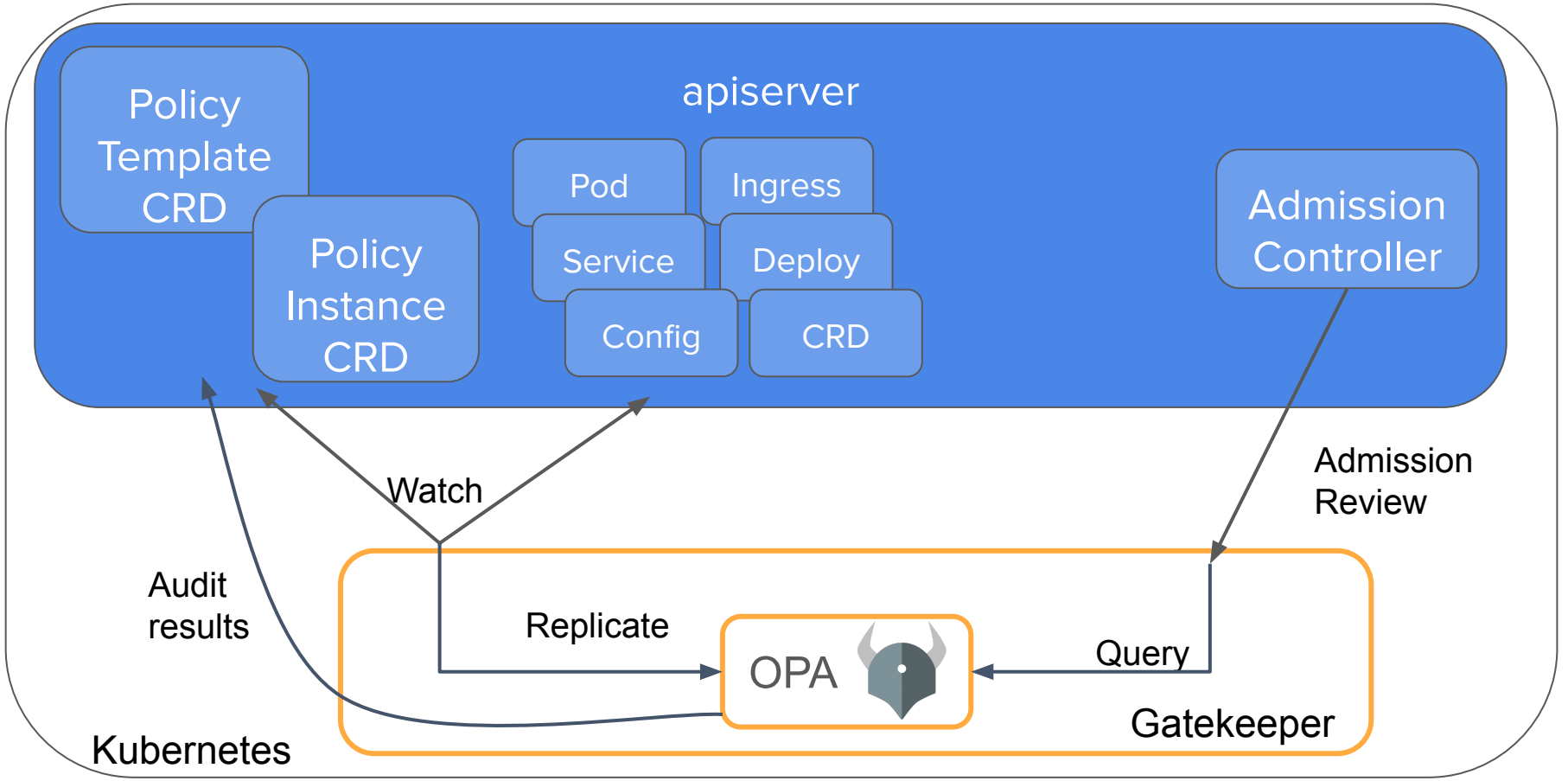
helps enforce policies and strengthen governance

Motivation

- Control what end-users can do on the cluster
- Help ensure clusters are in conformance with company policies
- Preview the effect of policy changes in production clusters to prevent impacts on existing workloads

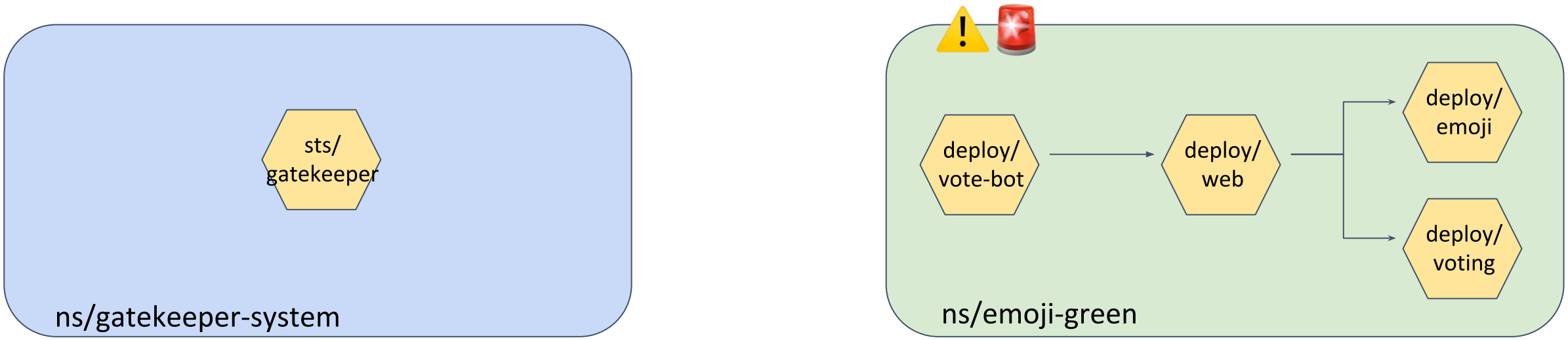
How do we help ensure conformance without sacrificing agility and autonomy?

OPA Gatekeeper 3.0

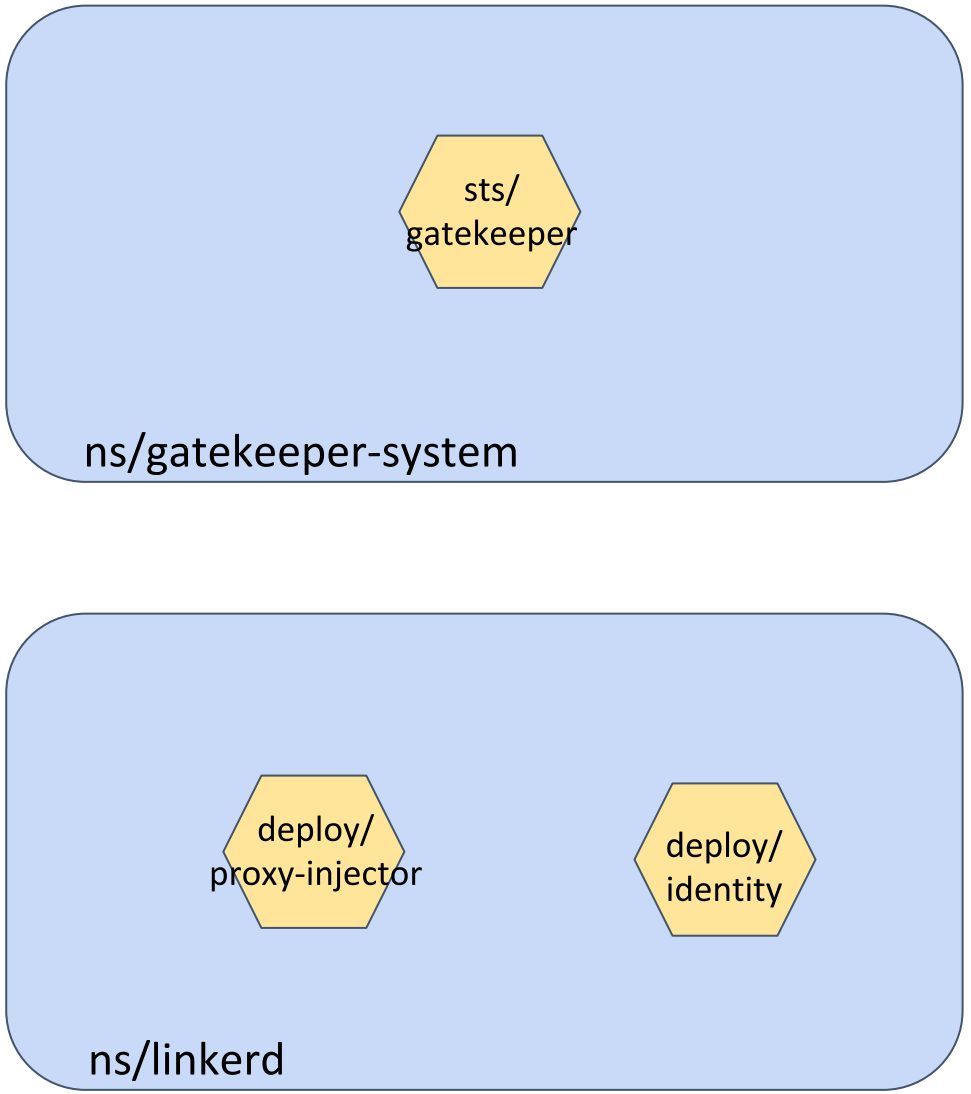


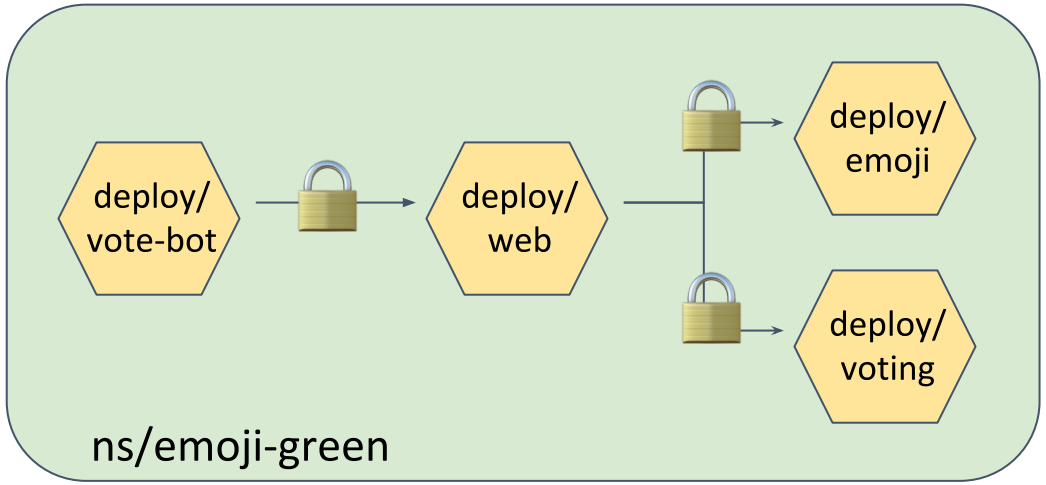
- Validating admission. CICD.
- Policy template defines Rego rules
- Policy instance parameterizes rules
- Policies stored as CRD
- Audit results stored on policy CRs
- Dry run to enable gradual rollout to build confidence
- Context-aware/referential policies
- Google, Microsoft, Redhat, CBA, Styra
- “Gatekeeper” donated by Replicated

Demo 2 - Enforce Policies with Gatekeeper



Demo 2 - Enforce Policies with Gatekeeper





Closing Thoughts

Additional features we did not have time to demo:

1. HTTPS only
2. Unique ingress hostname
3. Services must all have globally unique selector
4. Constraint the identity of the caller
5. Reject workload deployment right away
6. Raise non-TLS traffic alerts in alertmanager

Secure Your Services

Join Us!

LINKERD

- Development is all on GitHub
- Thriving community in the Slack
- Formal announcements on the CNCF mailing lists
- Monthly community calls
- Formal 3rd-party security audits

Linkerd has a friendly, welcoming community!
Join us!

Linkerd is 100% Apache v2 licensed, owned by a neutral foundation (CNCF), and is committed to open governance.

Open Policy Agent

openpolicyagent.org
github.com/open-policy-agent/opa

OPA Gatekeeper

github.com/open-policy-agent/gatekeeper

Community

slack.openpolicyagent.org
#kubernetes-policy

Meetings Tue @2p Pacific