CLOUD NATIVE
COMPUTING FOUNDATION

Zero Trusting as a True Cloud Native Dev

Guilherme Cassolato

Principal Software Engineer @ Red Hat

Agenda

Cloud Native

Zero Trust

Authentication

Authorization

Demo

Bonus tip

Cloud Native

From [CNCF Cloud Native Definition v1.1]

> *“[…] develop, build, and deploy workloads in computing environments (public, private, hybrid cloud) […] at scale in a programmatic and repeatable manner. It is characterized by loosely coupled systems that interoperate in a manner that is secure, resilient, manageable, sustainable, and observable.*
>
> *“Cloud native technologies and architectures typically consist of some combination of* ***containers, service meshes, multi-tenancy, microservices, immutable infrastructure, serverless, and declarative APIs*** *[…].”*

Zero Trust

In a nutshell(*) – No request is safe by default

Straight to the remedy → External authz proxy(**)

Keeping ourselves honest:

- Proxy != sidecar
- Proxyless approaches do exist
- Proxies punish performance... kinda true
- Proxies add another point of failure - another half truth

Authentication

API keys

OpenID Connect (OAuth2) → JWTs

x509 certificates (mTLS)

Kubernetes TokenReview

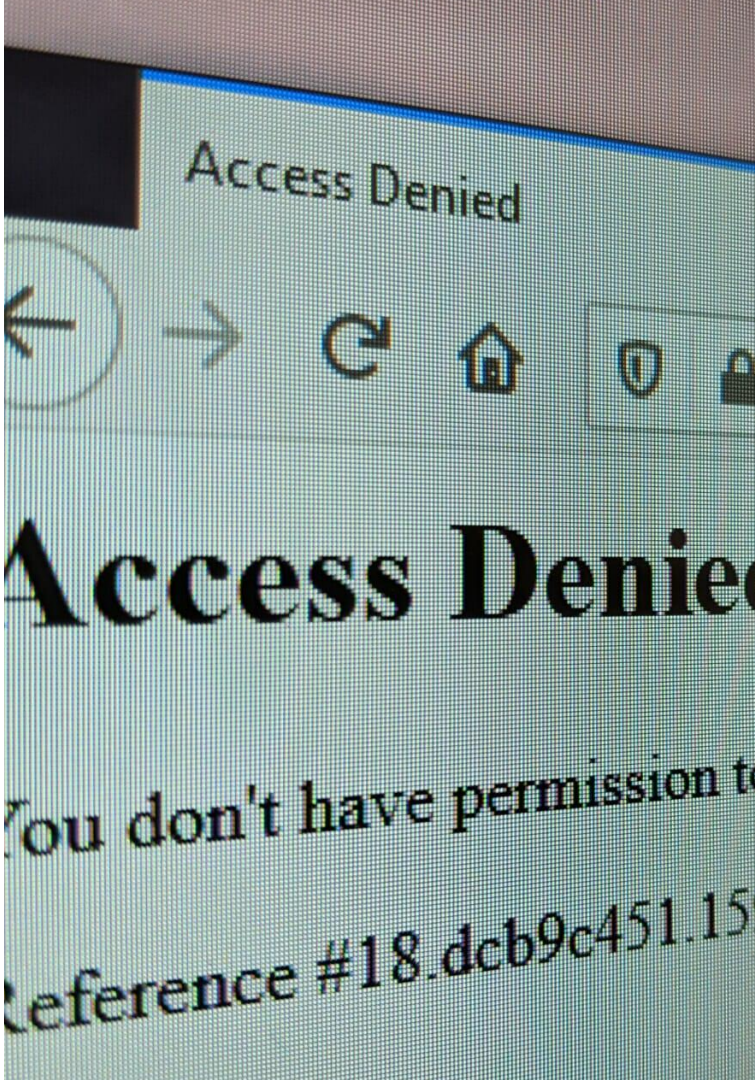
Authorization

JWT claims

ABAC / Policies (OPA *et al*)

ReBAC (OpenFGA, SpiceDB)

Kubernetes RBAC (SubjectAccessReview)



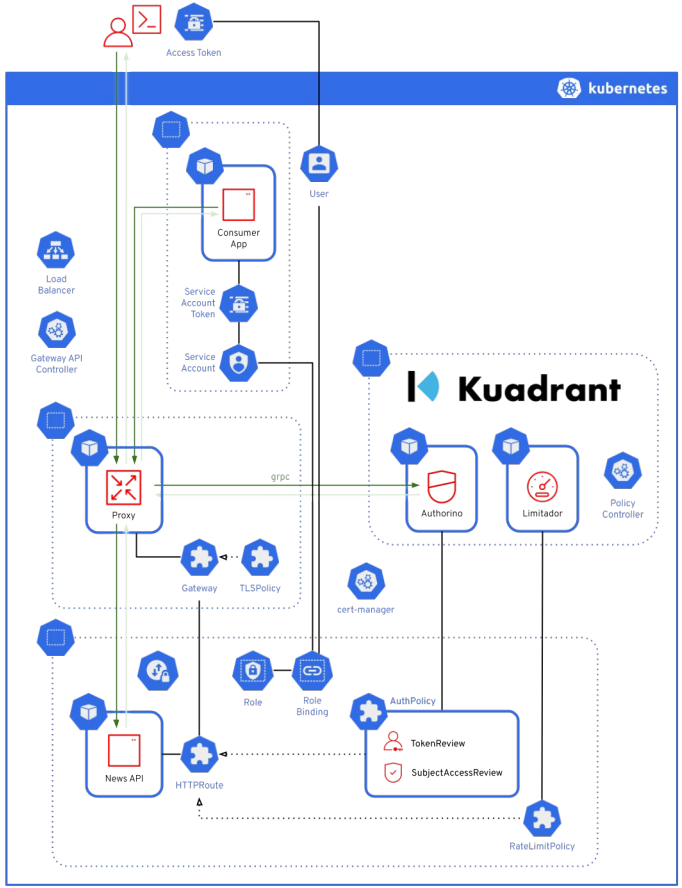
CNCF App Sec Landscape

Recap

Cloud Native → Kubernetes as platform & API language for configs and policies

Zero Trust → All requests checked for AuthN/Z via an ext_authz proxy

Leveraging Kube for auth → TokenReview and SubjectAccessReview APIs



Demo time

https://github.com/guicassolato/kube-auth-kuadrant

```yaml
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: test-network-policy
  namespace: default
spec:
  podSelector:
    matchLabels:
      role: db
  policyTypes:
  - Ingress
  - Egress
  ingress:
  - from:
    - ipBlock:
        cidr: 172.17.0.0/16
        except:
        - 172.17.1.0/24
    - namespaceSelector:
        matchLabels:
          project: myproject
    - podSelector:
        matchLabels:
          role: frontend
    ports:
    - protocol: TCP
      port: 6379
  egress:
  - to:
    - ipBlock:
        cidr: 10.0.0.0/24
    ports:
    - protocol: TCP
      port: 5978
```

NetworkPolicy API

See also:

- AdminNetworkPolicy (ANP)
- BaselineAdminNetworkPolicy (BANP)


CLOUD NATIVE
COMPUTING FOUNDATION