THE LINUX FOUNDATION
OPEN SOURCE SUMMIT
INDIA

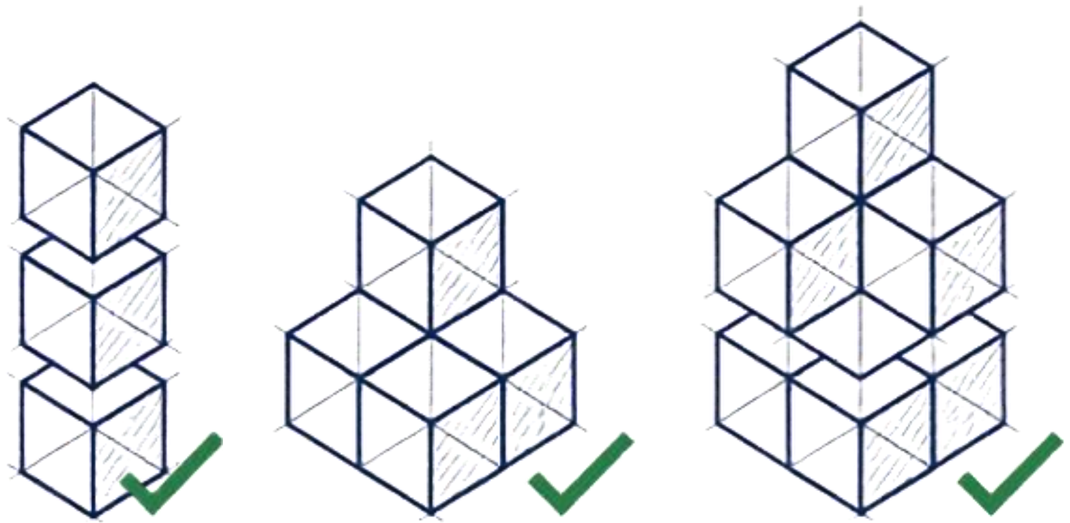
Kubernetes OIDC That Works in Practice: Keycloak + RBAC + Kubelogin Without Day-2 Pain

Manik Bindlish

Orange Business Services

Kubernetes knows infrastructure. It knows nothing about people

WHAT KUBERNETES KNOWS

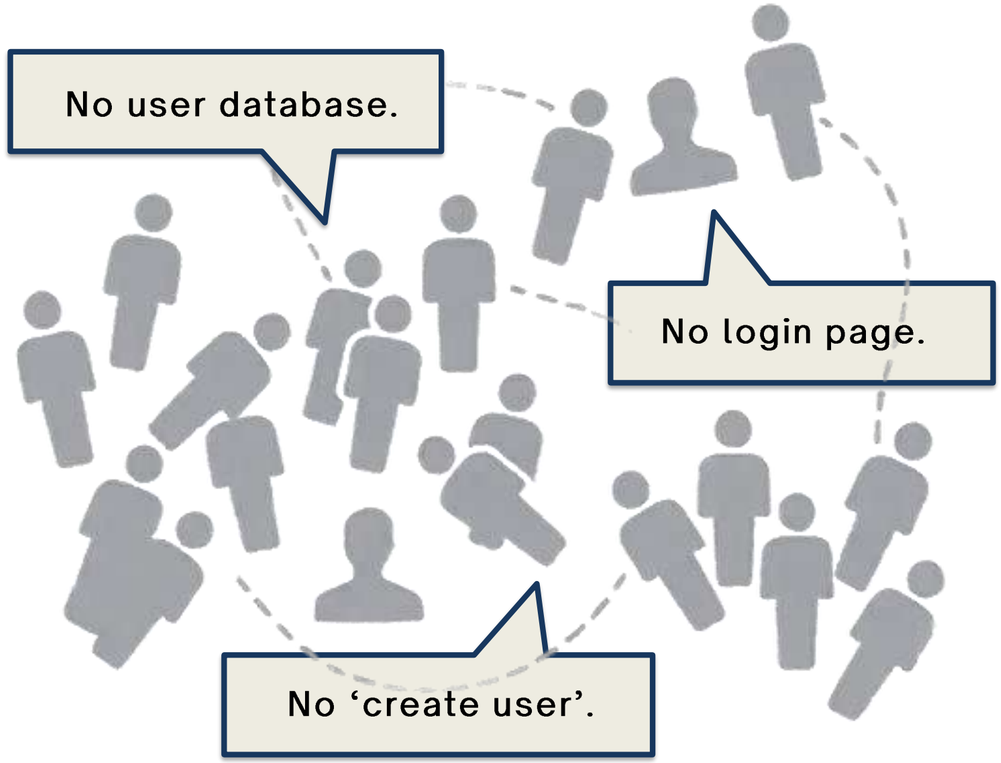


Pods

Deployment

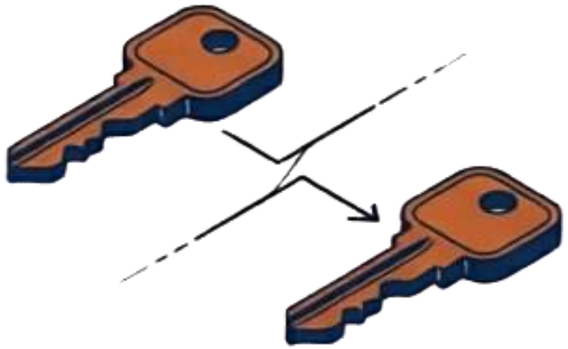
Namespaces

WHAT KUBERNETES DOESN'T KNOW



The Reality of sharing Kubeconfigs

New Engineer Joins



Copy kubeconfig

↓

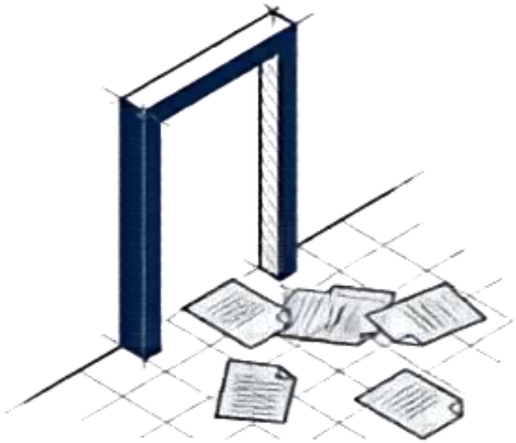
Email file

↓

Token NEVER expires

Now 2 people have god-mode access

Engineer Leaves



Who has which config?

↓

Rotate every service account?

↓

Old token still works 6 months later

Total Access Chaos

Security Audit Hits

Who accessed Production?

↓

Log:system:serviveaccount:admin

↓

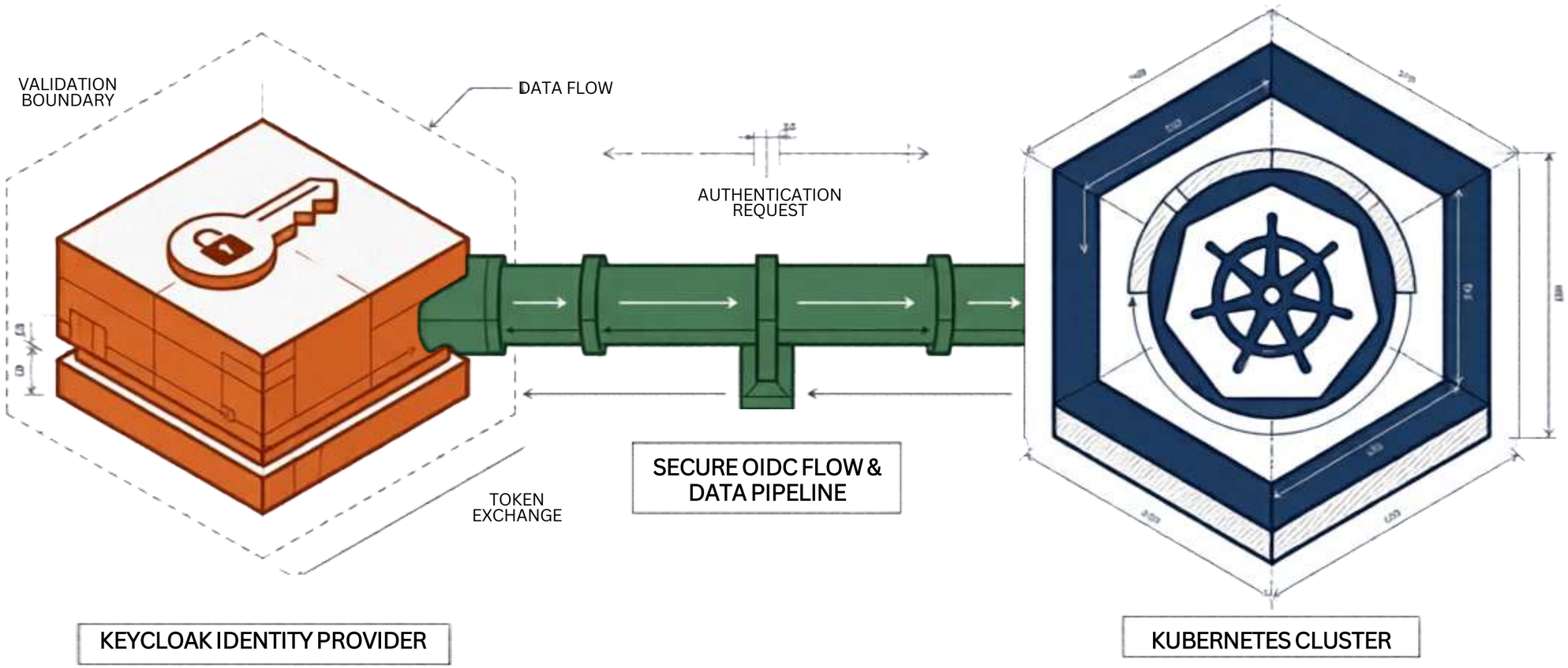
Answer

Zero Accountability

Root Cause: Everyone shares the same token.

OIDC THAT ACTUALLY WORKS

A standard protocol built on top of OAuth 2.0 that answers one question: Who are you?

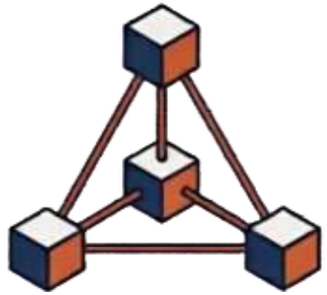


Keycloak: Open-source Identity Provider (IdP) that speaks OIDC

Keycloak fills the identity gap.

IDENTITY

Real people with emails, passwords, and MFA.
No more service accounts for humans.



GROUPS

Structured teams (infra admins, dev-team-a’). Maps directly to Kubernetes RBAC.

LIFECYCLE

`Onboard in 60 seconds.`
`Offboard in 5 seconds.`
`Group membership equals access.`

TOKENS

Short-lived JWTs. Issued, signed securely, and strictly time-bound.

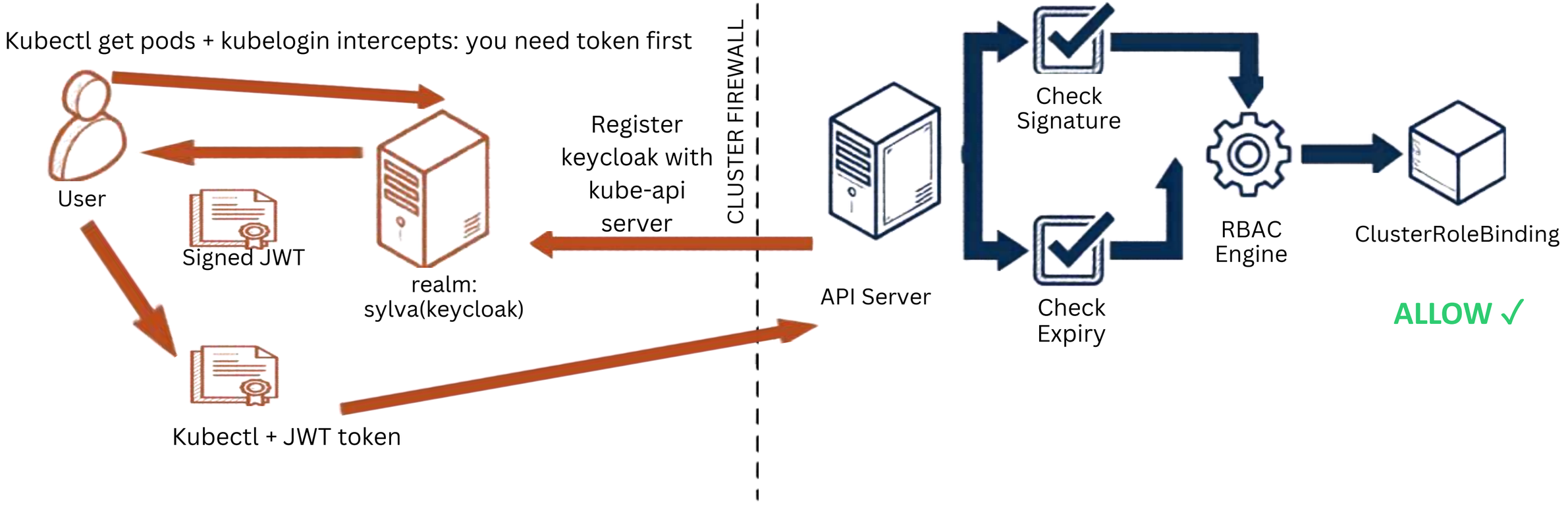
Authentication vs. Authorization

KEYCLOAK

Proves WHO you are.

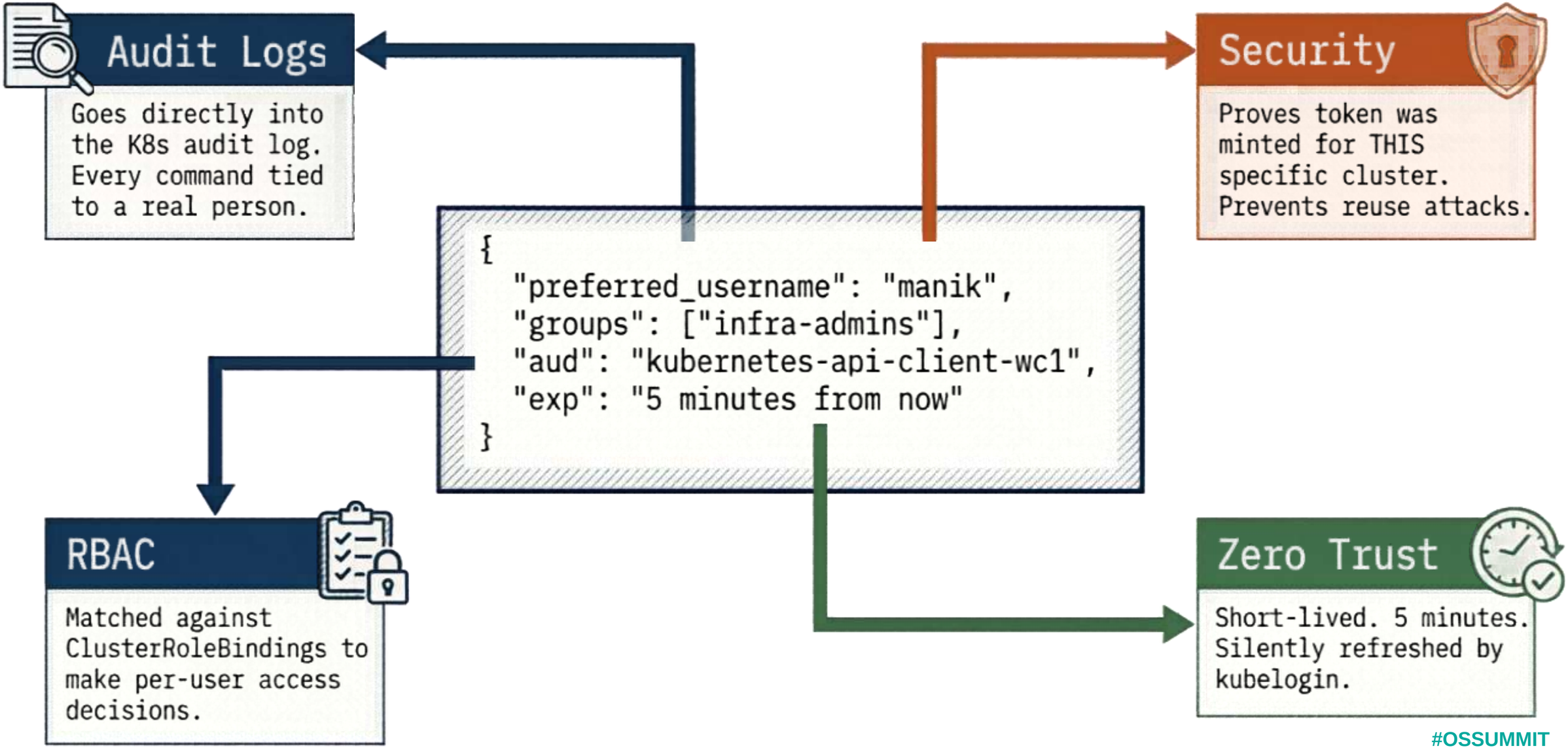
KUBERNETES API SERVER

Decides WHAT you can do.

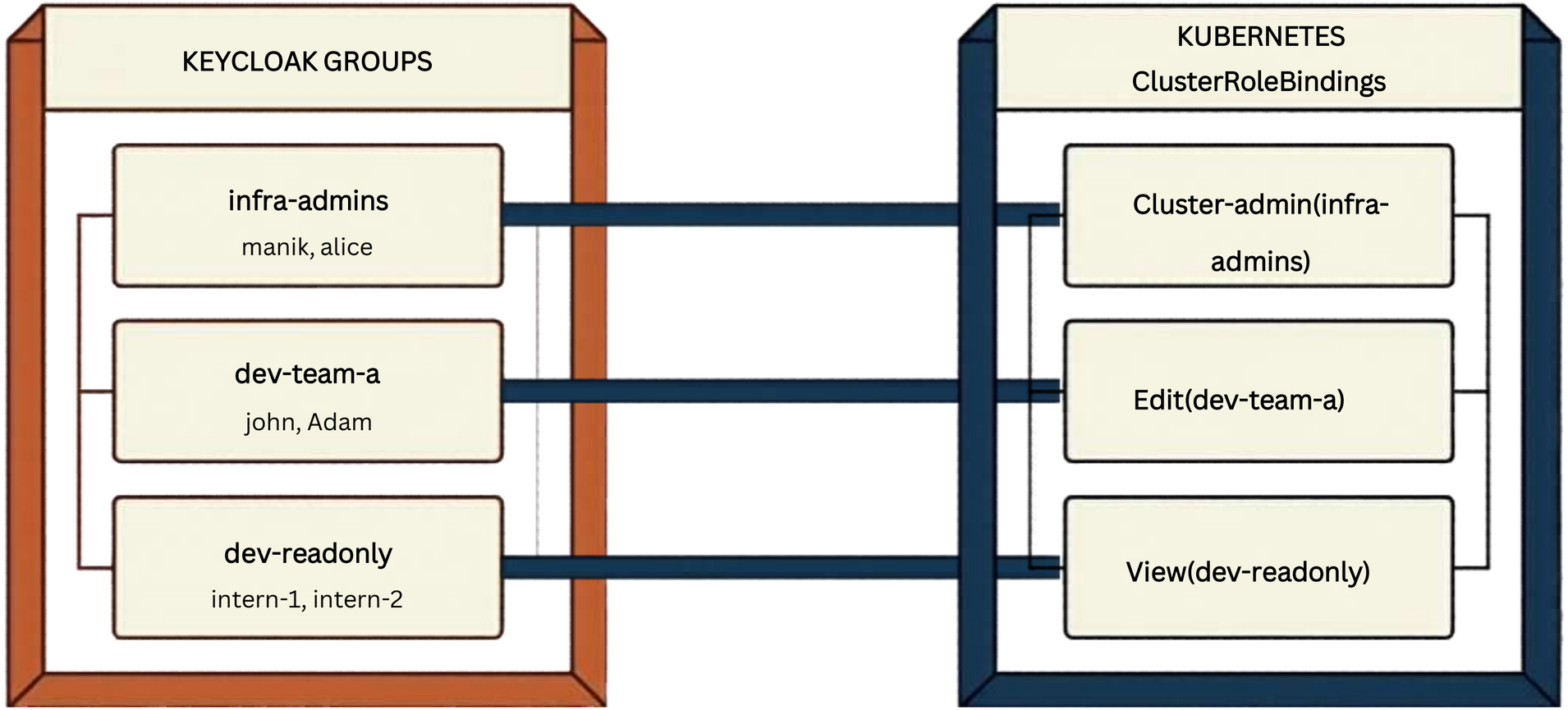


Authentication = Keycloak proves ***WHO*** you are **Authorization** = Kubernetes decides ***WHAT*** you can do .

X-Raying the JWT Token.



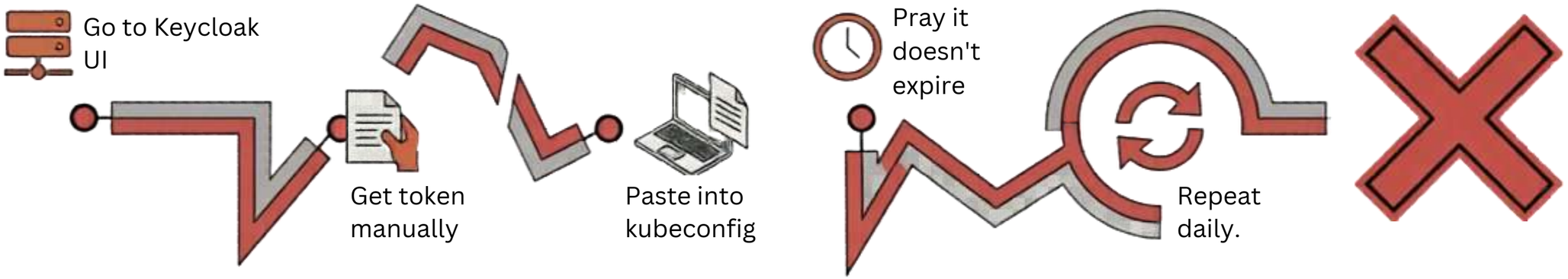
Groups map directly to Kubernetes RBAC.



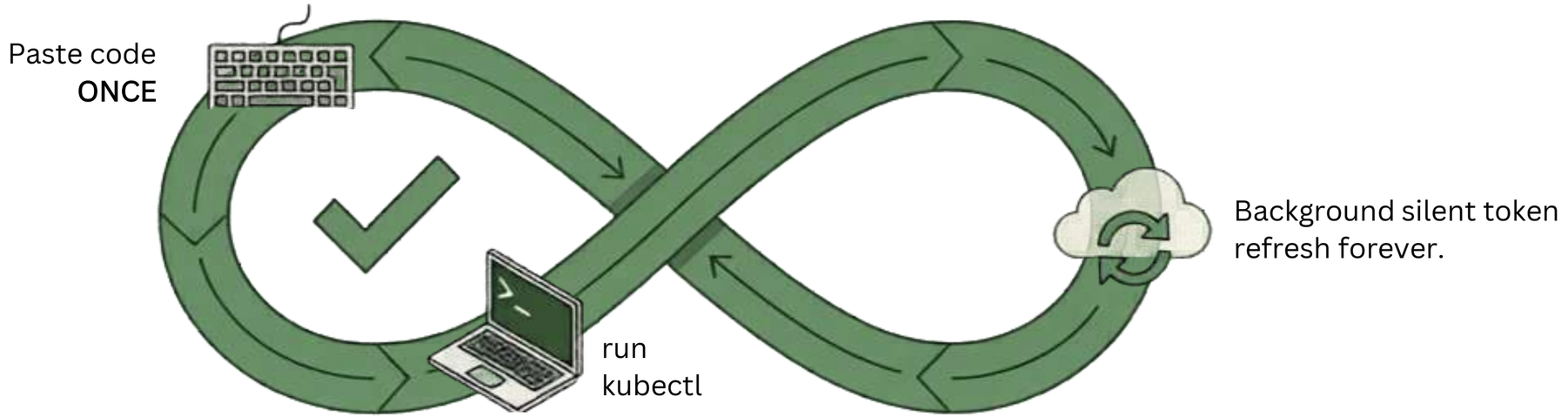
Add someone to a Keycloak group -> they get that RBAC level. No kubctl apply. Automatic.

The missing engine: kubelLogin

Without kubelLogin



With kubelLogin



kubelLogin is a kubectl exec credential plugin. It lives in your kubeconfig and manages refresh automatically.

Onboarding and off boarding done right.

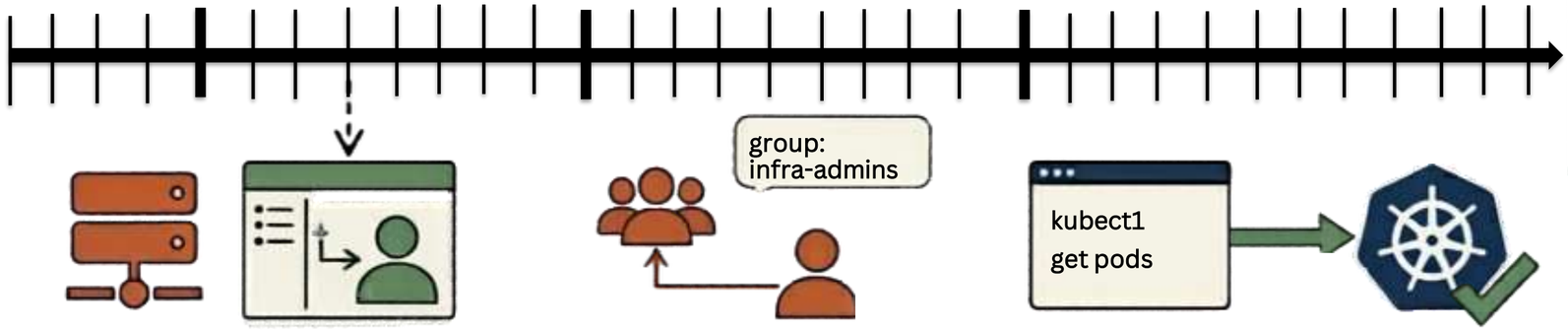
Monday 9:00 AM

- Onboarding

9:00 AM
Admin adds user to Keycloak.

9:01 AM
Added to “infra-admins” group.

9:02 AM
kubectl runs.
Access granted.



Zero kubeconfig files shared.

Total time: 2 minutes.

Friday 5:00 PM

- Offboarding

5:00:00 PM
Admin disables user in Keycloak.

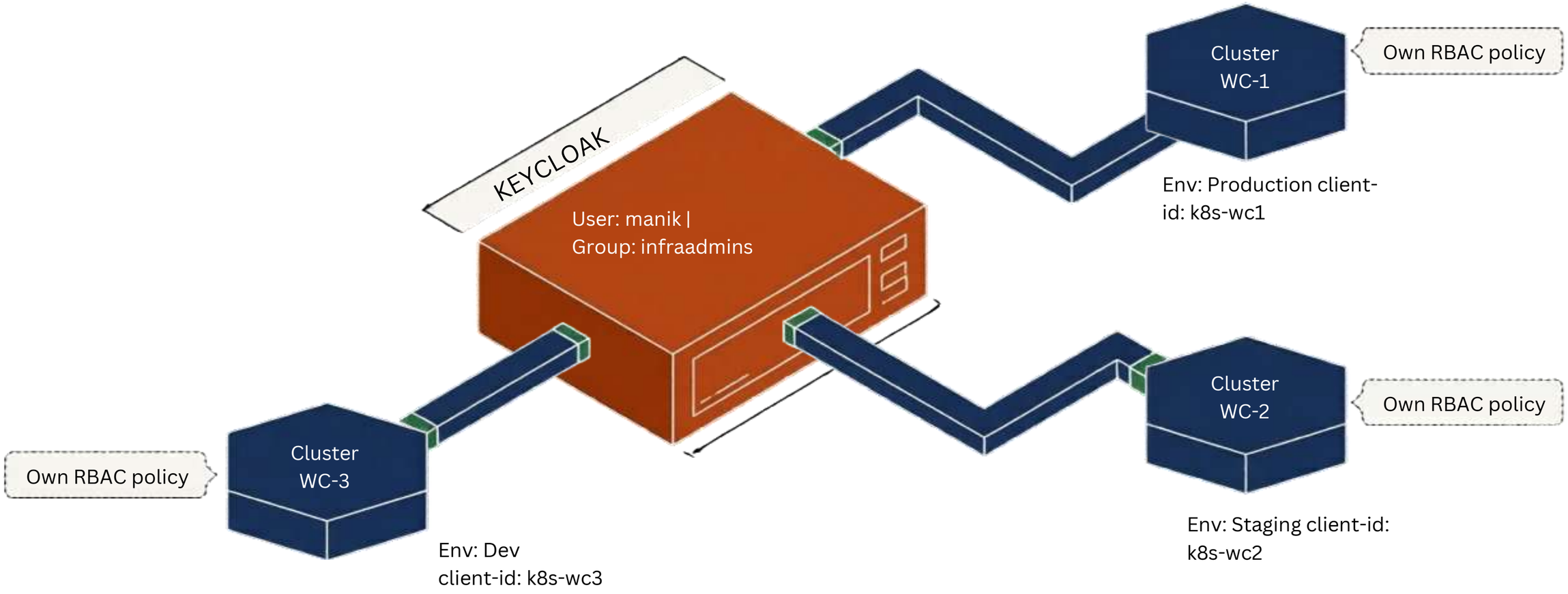
5:00:01 PM
Next token refresh fails.

5:00:05 PM
kubectl stops working.

No secrets to rotate.

Total time: 5 seconds.

One Identity Plane, Infinite Clusters.



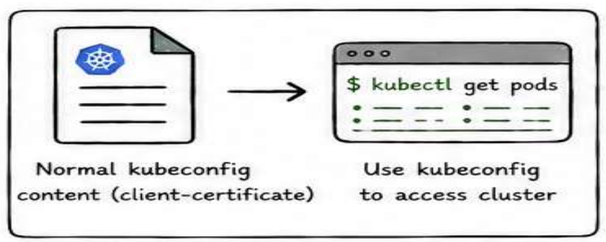
One user. One password. Kubelogin caches per-cluster tokens and refreshes them independently.

Demonstration

DEMO FLOW

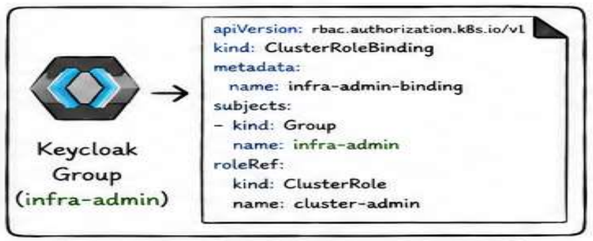
What we'll see in the demo

1 Traditional Way



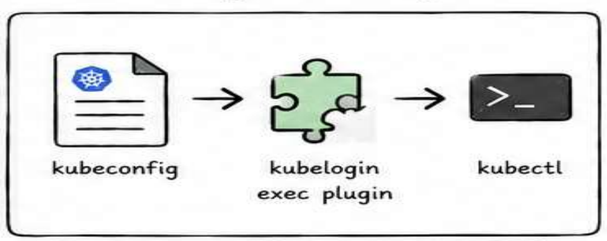
☆ Client certificate based authentication

2 Mapping of Keycloak Group with Cluster Role



☆ Keycloak group mapped to cluster-admin permissions

3 How kubeconfig creds use to configure kubelogin

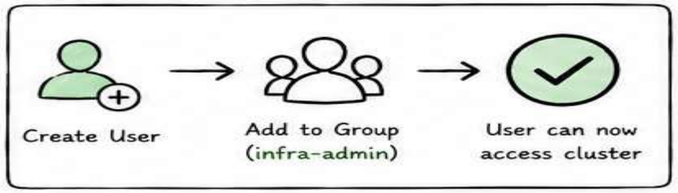


☆ kubelogin plugin in kubeconfig enables OIDC authentication

4 Accessibility of Keycloak UI

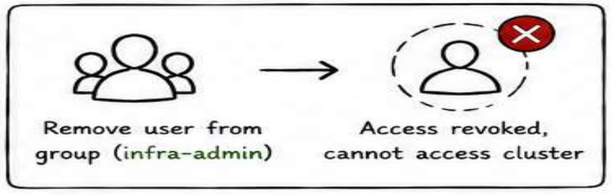
☆ Manage users, groups and roles centrally

5 User Creation and Assigning in Group



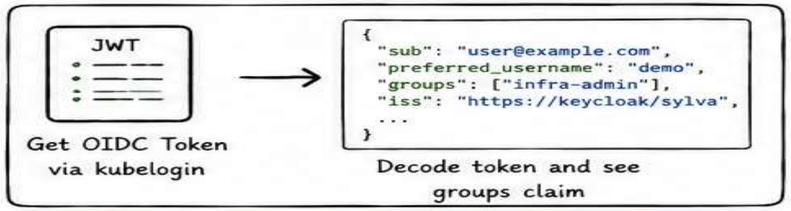
☆ Access granted based on group membership

6 User Removal from Group



☆ Access is revoked immediately after group removal

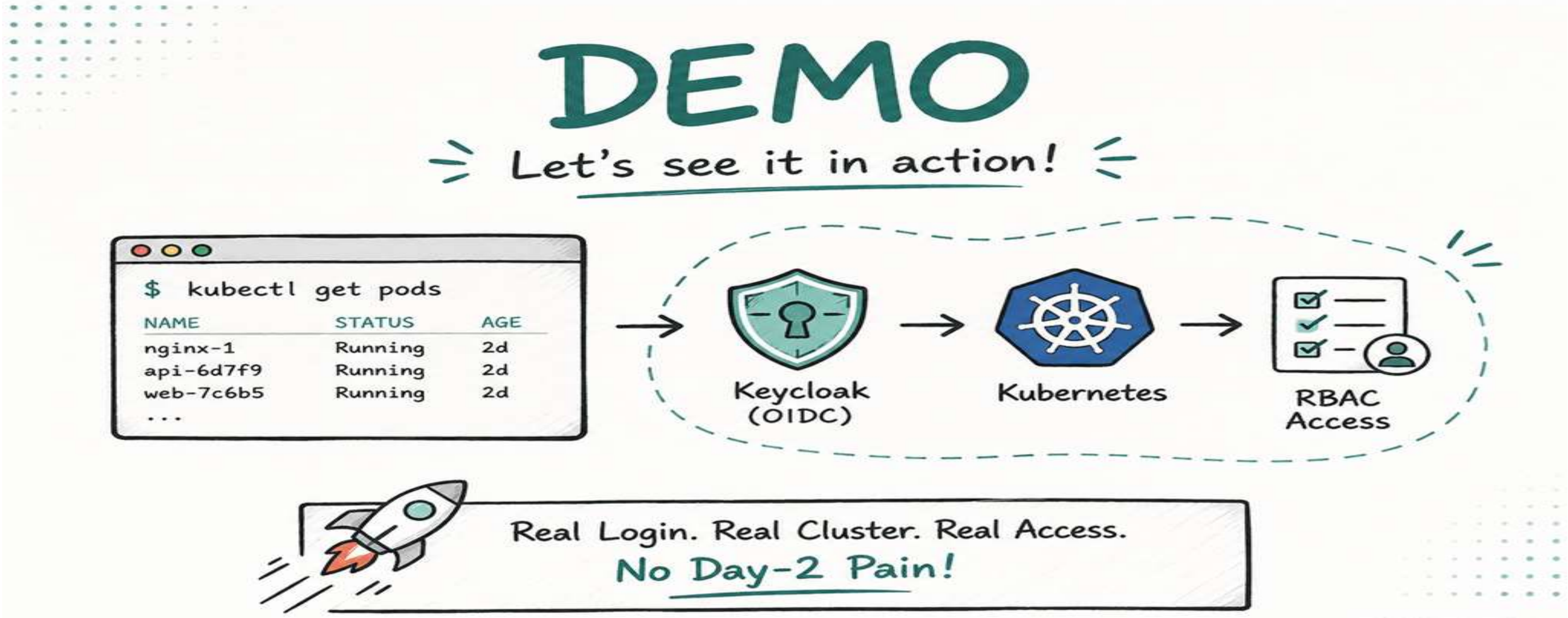
7 Decode Token



☆ Groups claim in token drives Kubernetes authorization

See how Kubernetes access is fully managed via Keycloak groups, OIDC & kubelogin.
No certificates. No manual user management in Kubernetes.

Demonstration



Diagnostic Cheat Sheet: 5 Common Traps.

You see	Cause	10-second fix
Forbidden	groups: null in token	Mapper needs multivalued: true
`x509: unknown authority`	Private CA not passed	`--certificate-authority=ca.pem`
Fix applied, still fails	Cached old token	Delete `~/.kube/cache/oidc-login/`
User logged in, wrong permissions	Wrong Keycloak realm	Check realm = sylva not `master`
Stops working after a week	Refresh token expired	Just re-login (by design)

The Golden Path Achieved.

Kubernetes finally knows your people.

Thank You

Let's Continue the Conversation

- Manik Bindlish
- Technical Lead | Orange Business Services

Scan to connect on LinkedIn

THE LINUX FOUNDATION

OPEN SOURCE SUMMIT INDIA