



API LIFECYCLE & GOVERNANCE

Self-Serve APIs. Centralized Governance.

A platform pattern for enterprise teams — exposing services through **Envoy Gateway** and cataloging them in **Backstage**.

POWERED BY



Envoy Gateway



Backstage

IMESH Engineering Team

Five Stages, One Lifecycle

Onboard · Design · Govern

Every API moves through the same five stages – from first registration to live operation.

01



Discovery & Ownership

Every API registered, owned, and findable in one catalog.

02



Publication & Exposure

Services exposed safely through standardized gateway routes.

03



Security & Governance

AuthN/AuthZ, rate limits, and mTLS enforced as policy.

04



Consumption & Self-Service

Teams find and integrate APIs without filing tickets.

05







Observability & Operations

Traffic, health, and usage visible across every environment.

What Powers Each Stage – and Who Owns It

Backstage and Envoy Gateway do the heavy lifting. Ownership stays explicit at every stage.

LIFECYCLE STAGE	ENABLING TECHNOLOGY	OWNER
01 Discovery & Ownership	 Backstage Catalog	Development
02 Publication & Exposure	 Envoy Gateway + HTTPRoute	Development
03 Security & Governance	 Envoy SecurityPolicy	Platform
04 Consumption & Self-Service	 Backstage IDP	Development
05 Observability & Operations	Envoy Metrics + Backstage	Platform + Dev

Three Teams, One Platform

A producer, a consumer, and the platform that connects them safely.

A **Alex**
Payments Team · 5 devs

PRODUCES APIS

- Creates payment & billing endpoints
- Exposes APIs for Team B
- Applies security & traffic policies

/payments /invoices /health

B **Bob**
Order Management · 3 devs

CONSUMES APIS

- Discovers APIs created by Team A
- Consumes APIs for testing
- Integrates without filing tickets

payment API invoice lookup

P **Pooja**
Platform Engineering

GOVERNS THE PLATFORM

- Visualizes every API, owner & policy
- Governs APIs and team actions
- Applies policies to control usage

Backstage Envoy GitOps / RBAC

The Golden Path for Onboarding

★ The paved road, defined once

Five reusable building blocks every team inherits — standardized, secure, and self-service by default.



Backstage Templates

Pre-built scaffolding for registration, discovery & onboarding.



API Exposure Workflow

A standardized path for safely exposing services to consumers.



Gateway & HTTPRoute

Form-based templates that configure routing resources.



Security Policies

AuthN/AuthZ, rate limiting & mTLS as reusable policy objects.



Ownership Model

Clear team-to-API mapping with RBAC and admin visibility.

API Creation, the GitOps Way

Alex fills a Backstage form – the platform's templates do the rest, all the way to the cluster.

<> **FastAPI Service**

PORT 8000

GET /payments

GET /invoices

GET /health

SPEC /openapi.json

The service Alex registers and exposes.

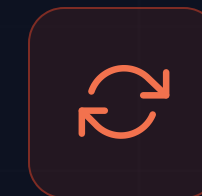
GITOPS PIPELINE



Backstage
Template &
scaffolding



GitHub
Config
repository




Argo CD
Monitors &
syncs




Kubernetes
Envoy config
applied


One Catalog, Total Discoverability


Backstage becomes the centralized API catalog – Bob discovers and consumes without ever pinging Team A.


 **Backstage · API Catalog**


API	OWNER	STATUS
payment-service /payments	Team A	prod
billing-api /invoices	Team A	prod
transaction-api /txn/validate	Team A	staging

 Inline OpenAPI / Swagger spec rendered per API

 **Search**
Full-text by name, team, or tag.

 **Ownership**
Team mapping with contact info.

 **Environments**
Dev, staging & production view.

 **RBAC**
See only what policy permits.

Ownership Stays with Dev. Governance Stays with Platform.

<> Application Team

- ✓ Owns & creates APIs
- ✓ Consumes APIs across teams
- ✓ Self-service onboarding
- ✓ Controls API visibility

VS

🛡 Platform Team

- ✓ 360° visibility of every API
- ✓ Implements RBAC & Envoy policy
- ✓ Deprecates or deletes APIs
- ✓ Full audit trail via Git history

Granular Access

Standardization

Developer Self-Service



Thank You

Self-serve API exposure with centralized governance — built for enterprise teams.

✉ [\[email protected\]](#)



Envoy Gateway



Backstage