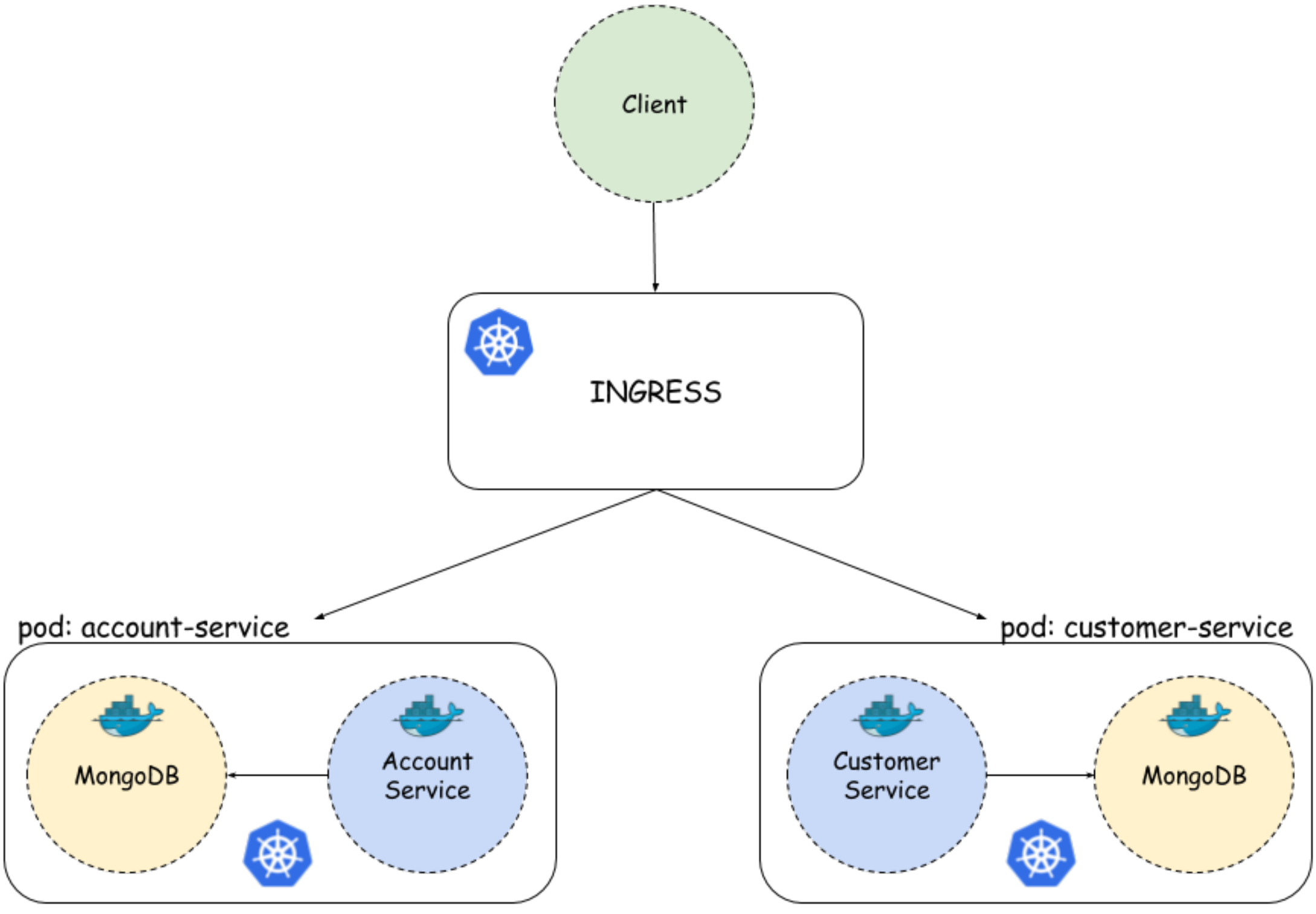
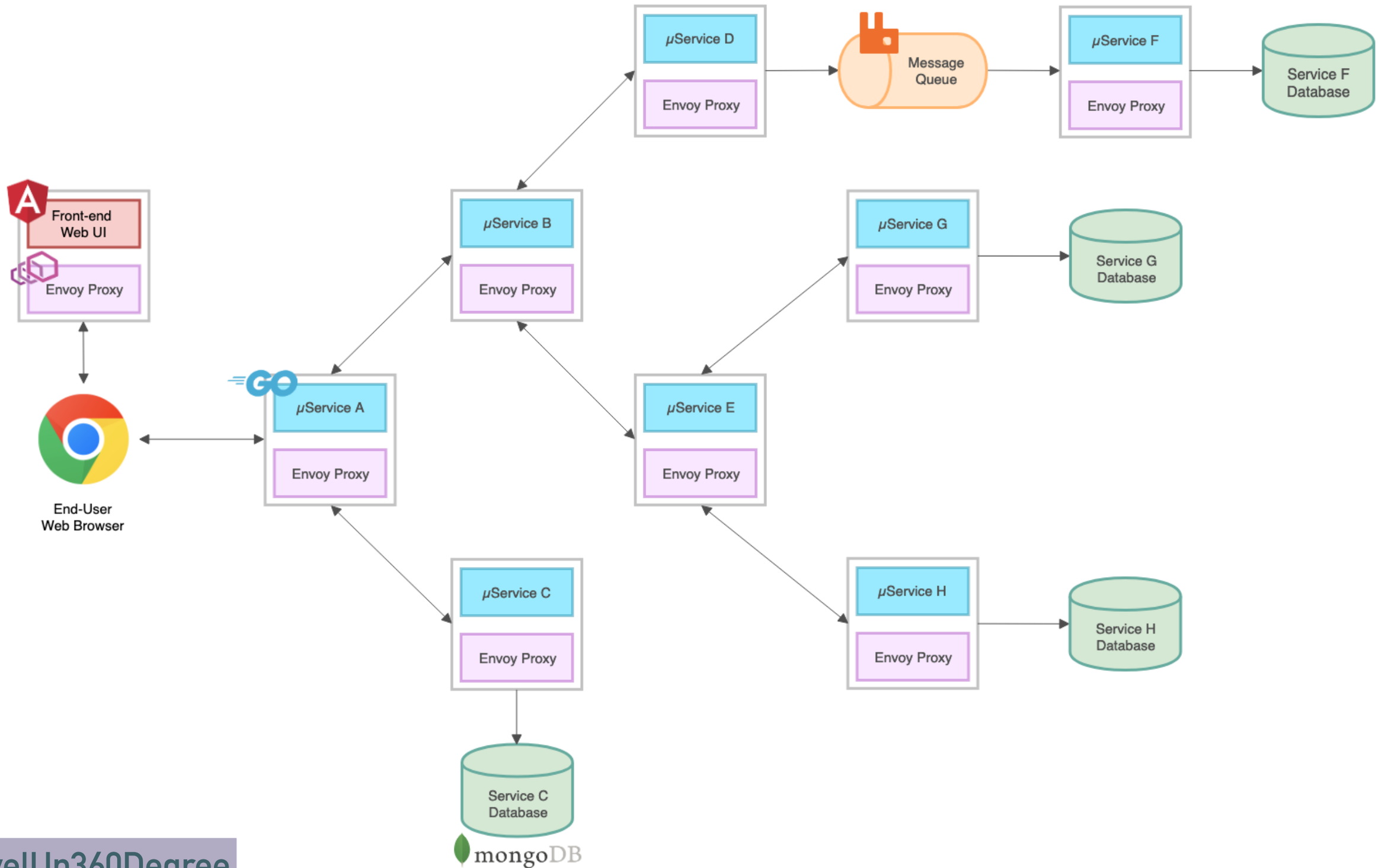
kubernetes

Kubernetes: Micro Services

- On **Kubernetes** we can deploy a variety of Applications.
- Those services could be **Stand Alone Services** that nothing to do with Other Services or **Micro-Services**, some small services that make an Application.
- Micro-Service Architecture is very Popular in these days and widely being used in Top IT Firms.
- Micro-Service Facilitate Developer to Split Application multiple chunks and individual processing capacity.


Client
INGRESS
pod: account-service
MongoDB
Account Service
pod: customer-service
Customer Service
MongoDB

KUBERNETES : MicroServices



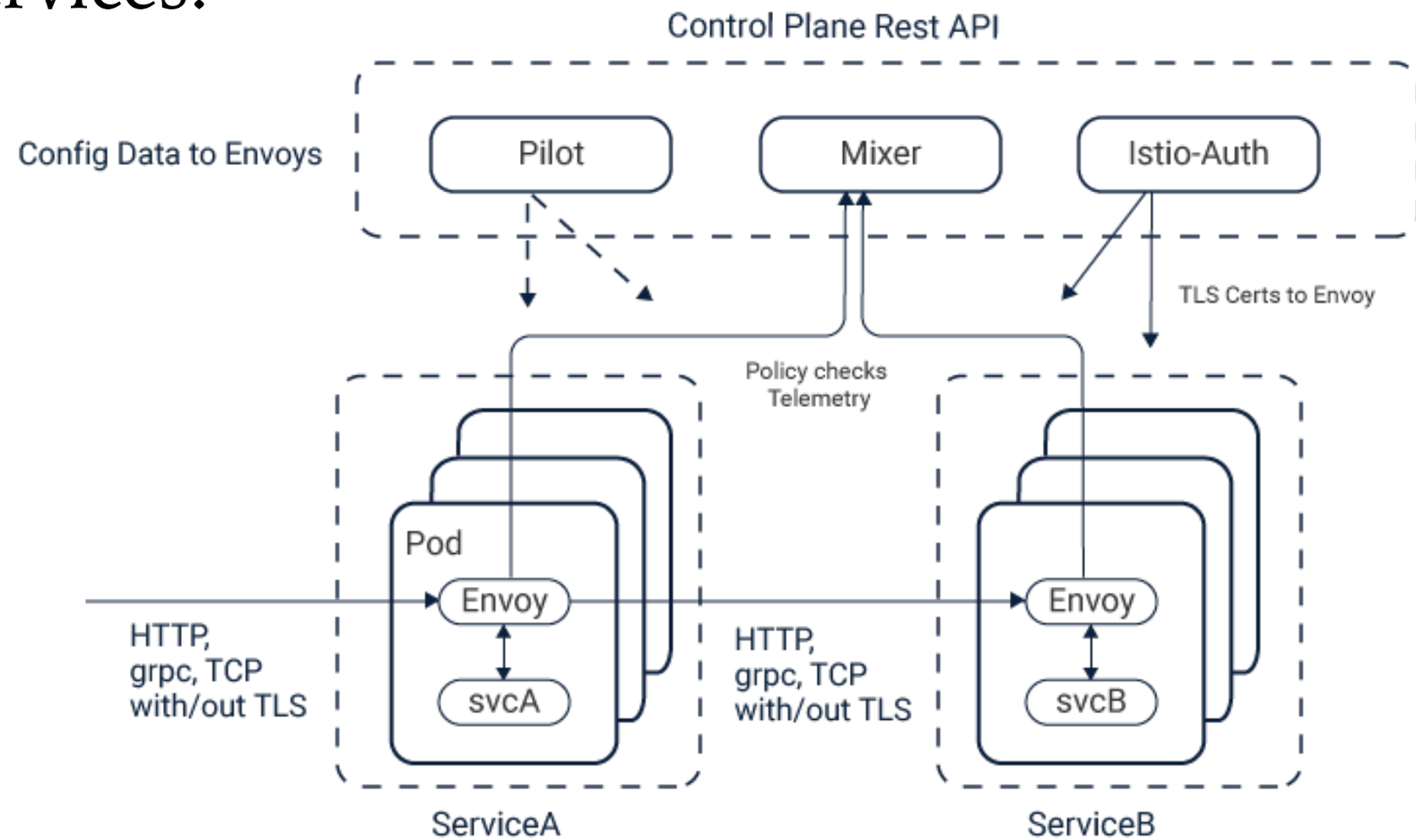
- Challenges with Microservices:
- No Encryption
- No Load Balancing
- No Failover / Auto Retries
- Routing Decisions
- Load Metrics/ Logs
- Access Control to Services

- Challenges with Microservices:
- No Encryption
- No Load Balancing
- No Failover / Auto Retries
- Routing Decisions
- Load Metrics/ Logs
- Access Control to Services

- **Istio** service mesh provides several capabilities for traffic monitoring, access control, discovery, security, resiliency, and other useful things to a bundle of services.
- **Istio** deployed for Micro Services without any change in code of **Micro Service**.
- To make this possible, Istio deploys an **Istio proxy** (called an **Istio sidecar**) next to each service.
- All of the traffic meant for assistance is **directed to the proxy**, which uses policies to decide **how, when, or if that traffic should be deployed to the service**.

How Istio Works With Containers and Kubernetes

- **Istio** service mesh, as suggested, uses a sidecar container implementation of the features and functions required mainly for microservices.



Will see you in Next Lecture…

See you in next lecture …