Jackie Gan

Professor Carl Williams

CIS 4360 Microservices Architecture

MSA case study one pager

Snapchat’s Service Mesh from Monolith to Multi-cloud microservices

Snap's backend was initially built on Google App Engine, but as the application grew, scaling became difficult due to its monolithic structure. To address this, Snap engineers proposed a microservices architecture, which uses independent backend services to limit outages during peak periods. They chose Envoy as their proxy, which allows communication with TLS and manages service dependencies, consumers, and traffic routing. Snap also implemented a switchboard to manage smaller pieces of the Envoy API, as well as provisioning, deployment, and k8 clusters. They minimized service exposure to the internet by using a shared, internal, regional network for their microservices. A main API gateway filters authentications and routes requests to the appropriate Service Mesh. As of today, Snap handles 10 million queries per second with 300+ productions using 7 service meshes on AWS and Google cloud.

In this topic, I would like show how snapchat transition from a monolithic application to a scalable microservices service mesh. I hope I can show why they made this approach, whether it’s cutting cost, scalabilities, or owning their own applications. I would like to show my fellow classmates how fortune 500 tech companies like snapchat decided to scale their application from a microservice perspective. I personally think this is a great way to show us that how a social media company would scale their applications compared to a traditional tech company like Google, Amazon, and Oracle that has all the servers and resources in their own hands. Why Snapchat design this application through multi-cloud instead a singular cloud infrastructure. What is the point of this resiliency. And many more decisions.

I would like to present on April 6th.