Service Mesh Networking with Containers and Microservices Applications

Brian Redmond, Microsoft
Brian Redmond

- Cloud Architect @ Microsoft (18 years)
- Azure Global Black Belt Team
- Live in Pittsburgh, PA, USA
- Avid marathon runner and outdoors enthusiast
- World traveler

@chzbrgr71
We replaced our monolith with micro services so that every outage could be more like a murder mystery.
Microservices

The answer to all problems.
The answer to all problems.
The answer to all problems.
Container networking adds challenges

“Understanding kubernetes networking: pods” (Mark Betz)
https://medium.com/google-cloud/understanding-kubernetes-networking-pods-7117dd28727
What’s missing?
service discovery & intelligent routing
Many ways to solve these problems...
Terminology

**microservice**

*noun | mic·ro·ser·vice | mike-ro-sər-vəs*

“architectural style that structures an application as a collection of loosely coupled services”

*(wikipedia)*
Terminology

**service mesh**

*noun | ser·vice mesh | sər-vəs mesh*

“a dedicated infrastructure layer for making service-to-service communication safe, fast, and reliable”

*(William Morgan, Buoyant)*
Service Mesh Architecture

Linkerd instances form a service mesh, allowing application code to communicate reliably.

What benefits does a Service Mesh offer?
Observability

Istio Workload Dashboard

Namespace: default
Workload: details-v1
Inbound Workload Namespace: All
Inbound Workload: All
Destination Service: All

WORKLOAD: details-v1.default

Incoming Request Volume: 0.15 ops
Incoming Success Rate (non-4xx responses): 100%
Request Duration

TCP Server Traffic: N/A
TCP Client Traffic: N/A

INBOUND WORKLOADS

Incoming Requests by Source And Response Code
Incoming Request Duration by Source
Incoming Request Size By Source
Incoming Success Rate (non-4xx responses) By Source
Response Size By Source

source: https://istio.io
Advanced routing and traffic shaping
Security and Encryption (TLS)

Source: https://linkerd.io
Resiliency. eg - circuit breaker pattern, retries

source: https://martinfowler.com
Other benefits

- Traffic mirroring
- Distributed tracing
- Fault injection
- Centralized policy enforcement
- Supports polyglot (language agnostic)
- Multicloud-platform (Kubernetes, Mesos, Nomad, etc.)
- Abstraction layer
Which service mesh should you use?

Everything is a service mesh 😊

**Actual Service Mesh**
- linkerd
- istio
- cilium

**Service Proxy**
- envoy
- nginx
- HAPrxy

**API Gateway**
- kong
- traefik
- tyk

**Others**
- consul connect
- nelson
Demo time

THE TECHNOLOGY DEMO

THE SOFTWARE ISN'T 100% COMPLETE.

IF IT HAD A USER INTERFACE YOU WOULD SEE SOMETHING HERE... HERE...AND SOMETIMES HERE.

AND THEN YOU'D BE SAYING, "I GOTTA GET ME SOME OF THAT."

ANY QUESTIONS?

@chzbrgr71
Summary

- “A dedicated infrastructure layer for making service-to-service communication safe, fast, and reliable”
- Simplifies microservices deployments on kubernetes
- “Stand on the shoulders of giants”
- Still an emerging space