Kubernetes Meetup

Istio at ME Bank

Who are we?

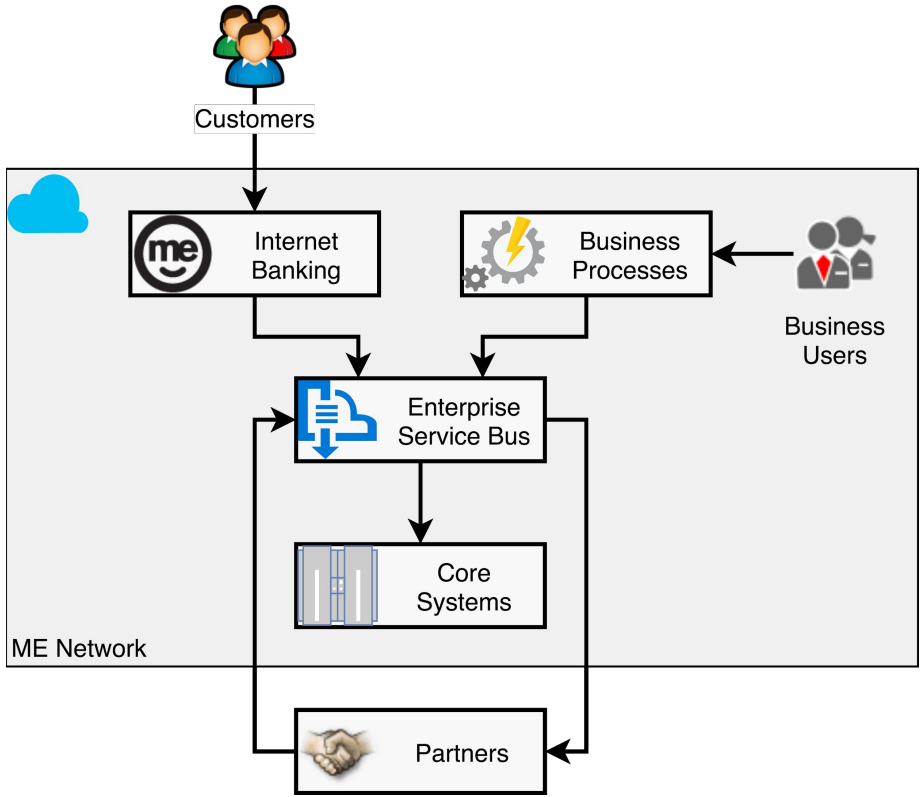
ME is owned by 26 of Australia's leading industry super funds, is branchless and has embarked the next transformation of it's digital services

DigIO is a melbourne based technology organisation who is partnering with ME Bank to establish their Kubernetes platform and microservice workloads

Topics for Today

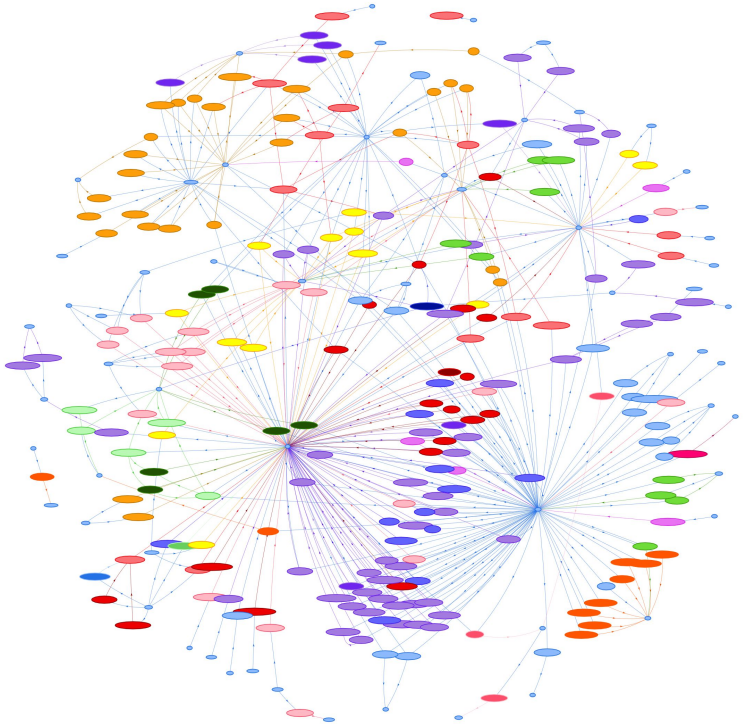
- Problem space
- Solution overview
- Why Istio?
- Topology
- Code and demo
- Takeaways

Problem space - Current



- Enterprise Service Bus is the common integration point between all major systems
- 250 operations, 150 services
- Majority are SOAP-based, ~40 batch file handlers

Problem space - Glimpse of reality



Problem space - The brief

Preserve API compatibility

Resilient to change

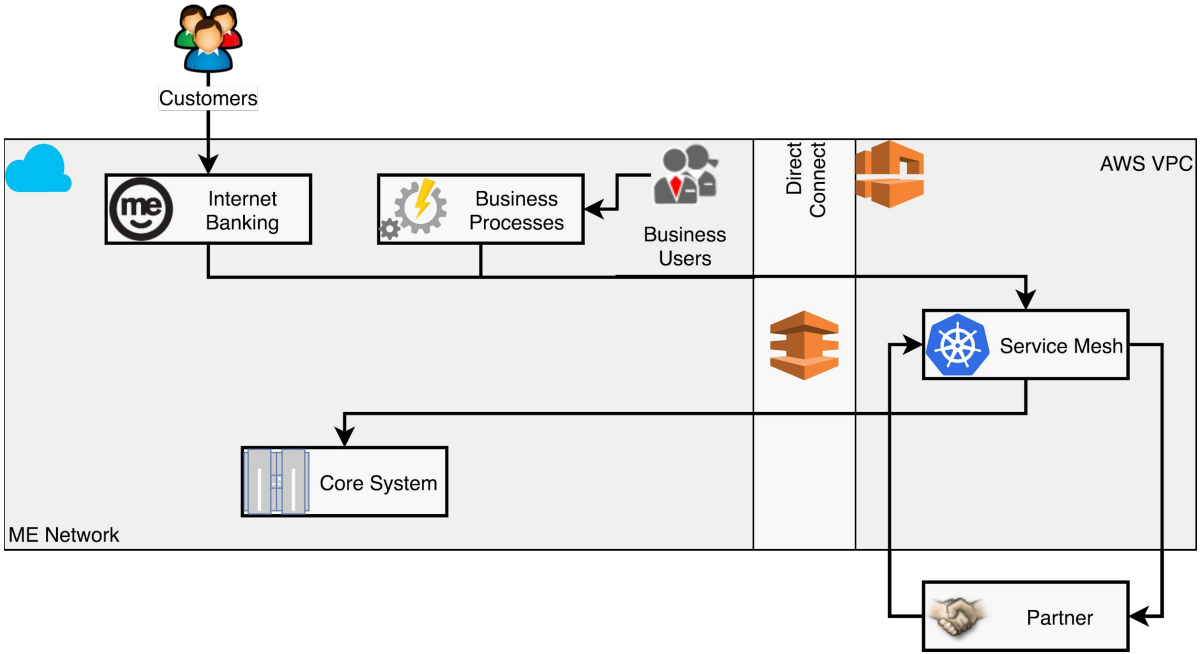
Reduce the risk of change

Optimise for development and ops

Reduce vendor lock-in

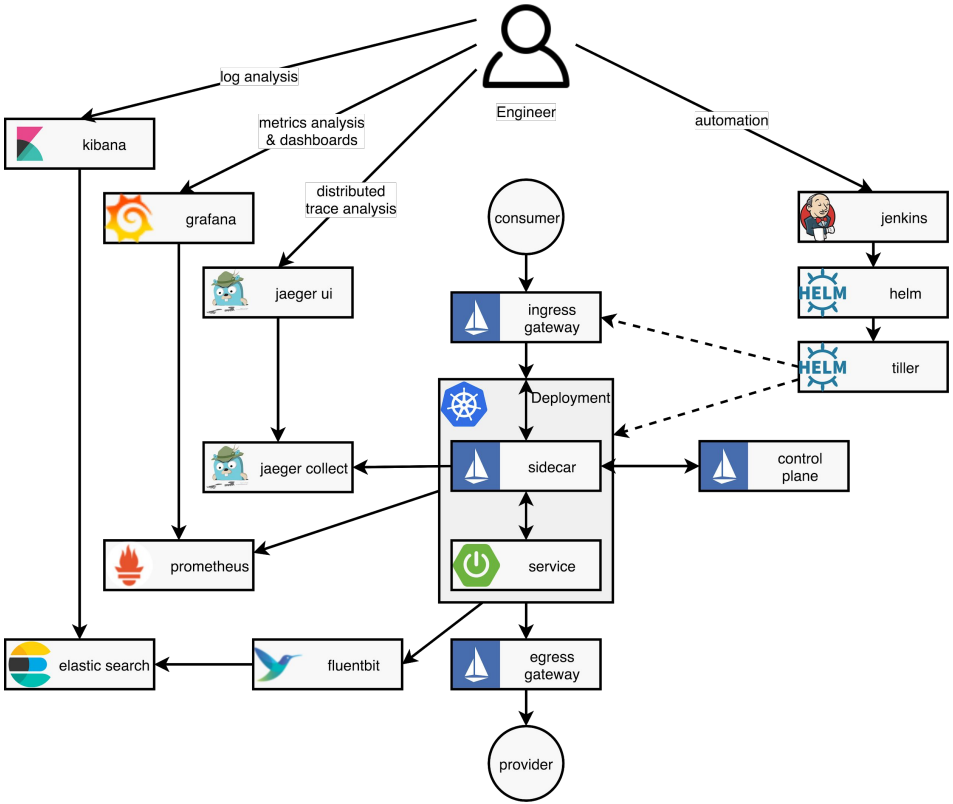
Attract and build a great team at ME

Solution Overview - To The Cloud!



- New AWS platform
- New Kubernetes platform

Solution Overview - A Peek Inside The Box



- New micro-service architecture
- New CI/CD solution
- New observability/ monitoring/alerting solution
- Same material workload requirements

Why Istio?

Sure, it was cool, interesting, and new. But ...

Remember, early this year:

- Still evolving, at 0.5 version, not considered production ready
- Limited skills and experience
- Limited community or case studies to learn from
- This is a large project for a bank, risks must be calculated

Why Istio? (continued)

It was a good fit for our requirements and objectives.

- **Secure** → Authentication, authorisation, fine-grained connectivity control.
- **Flexible** → Advanced routing topologies, canary deploys.
- **Available** → Circuit breakers, rate limiting, etc.
- **Observable** → Consistent logging, metrics, tracing.
- **Productive** → Manage complexity in platform. Minimise per-service effort.
- **Maintainable** → Healthy community, strong future with significant investment and support. Able to attract skilled people.
- **Resilient to change** → Support multiple languages/runtimes. Support a variety of integration protocols. Escape hatches if required. Apply new capabilities to existing services.

Why Istio? (continued)

Alternative? → Historically we would have used tech such as Netflix open source (e.g. Hystrix, Ribbon). Limits us to Java, hard to change direction. No single service able to justify new tooling.

Sidecar pattern provides an alternative → Separate connectivity concerns from app concerns, support multiple languages/runtimes. Java now, Node.js/Golang tomorrow? SOAP now, REST/gRPC tomorrow?

But 150 services → Managing 150 sidecar configurations efficiently is difficult. We need a way to manage the complexity.

Istio to the rescue ...

Risk mitigations → Monitor Istio progress, spike features early, selectively introduce Istio features.

Topology - Deployment

Pilot → Routing, connectivity config

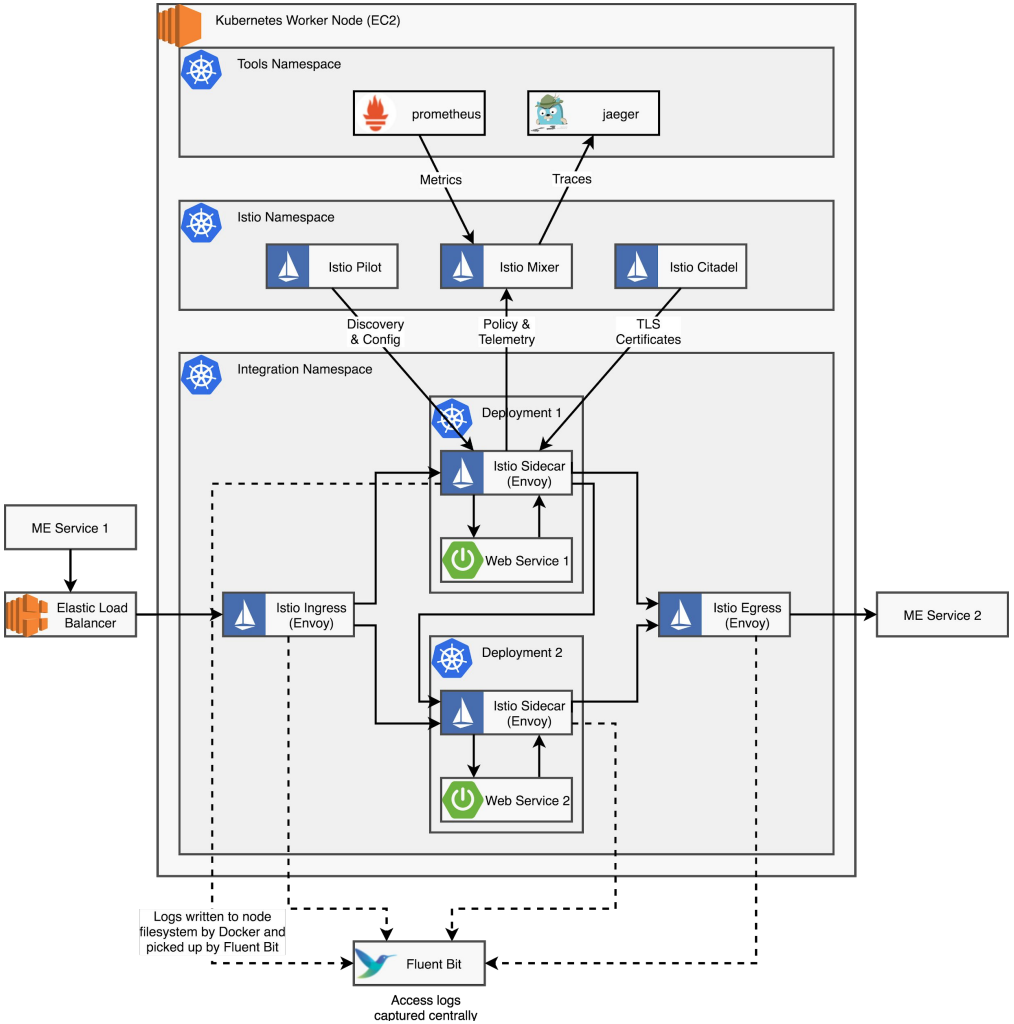
Mixer → Telemetry, auth/policy lookups

Citadel → Certificate management

Ingress → Inbound connectivity

Egress → Outbound connectivity

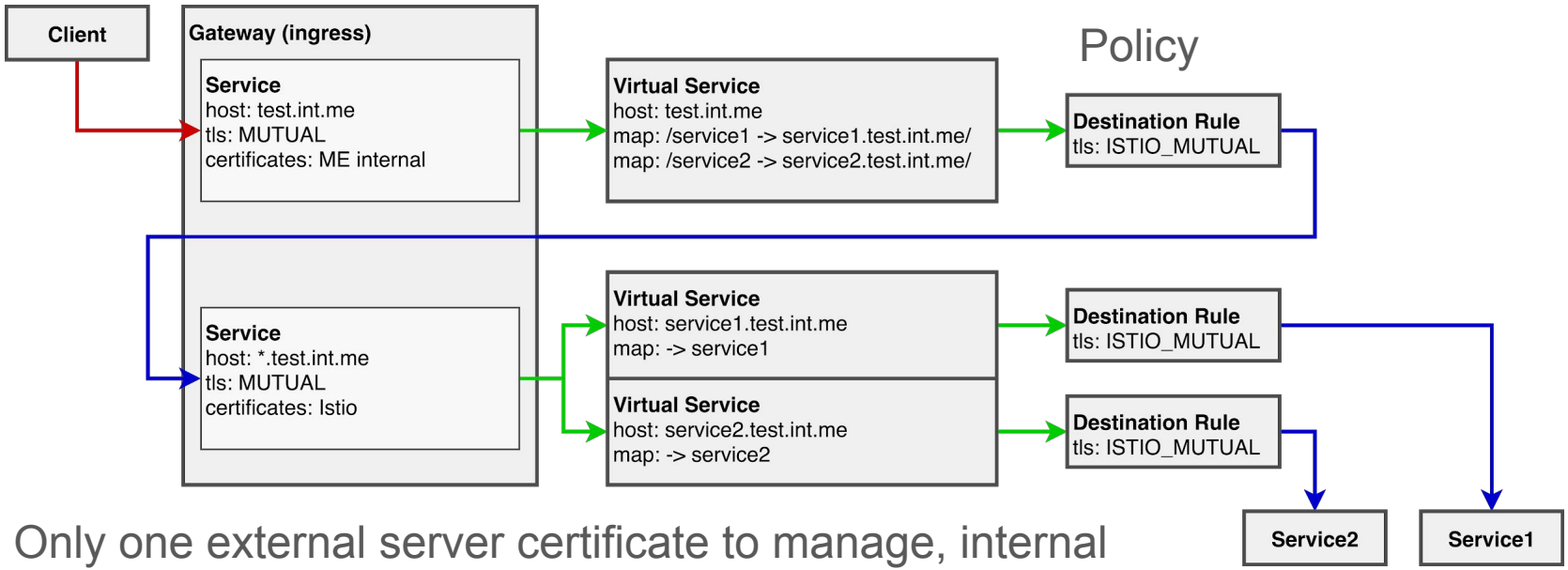
Sidecar → Per-service routing and policy enforcement



Topology - Istio Ingress

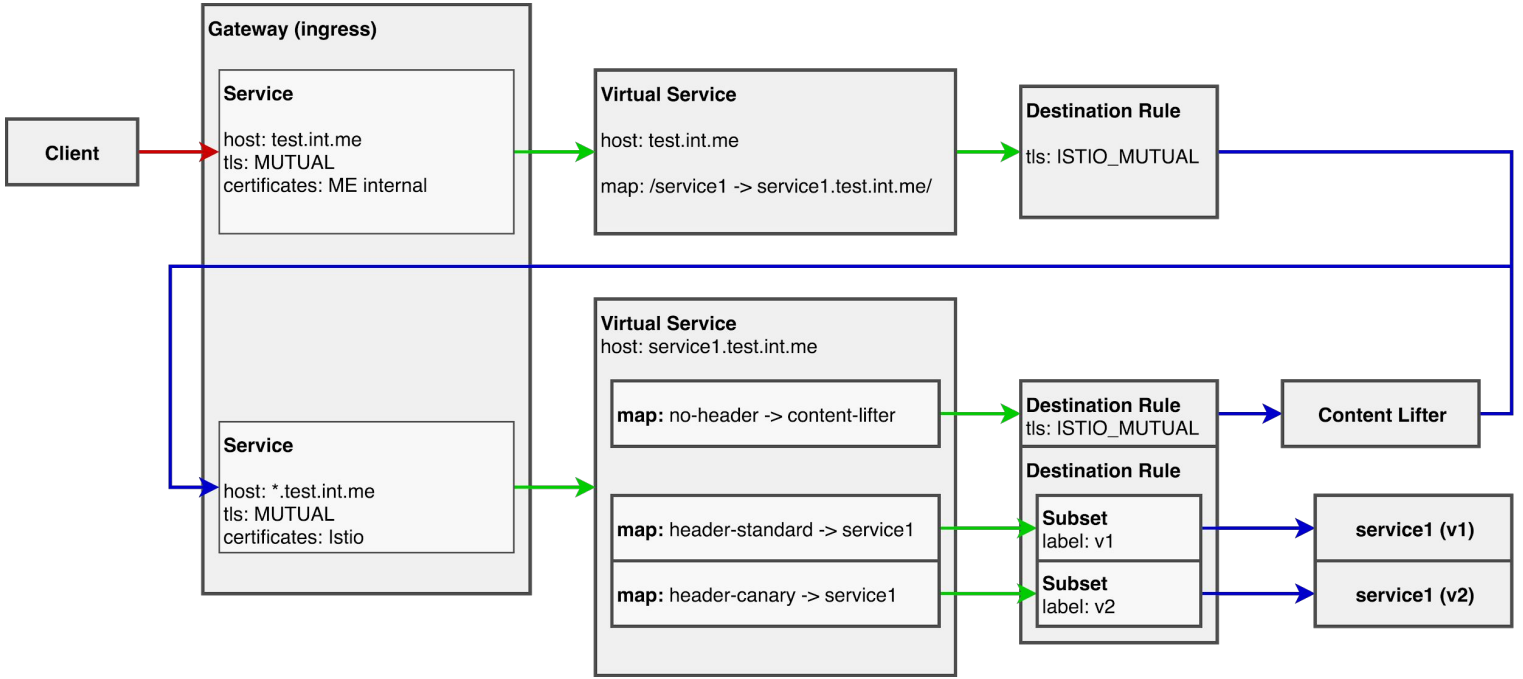
Resources

- **Gateway** → Mesh entry point
- **Virtual Service** → Routing rules
- **Destination Rule** → Connection Policy



Only one external server certificate to manage, internal certificates fully managed by Istio.

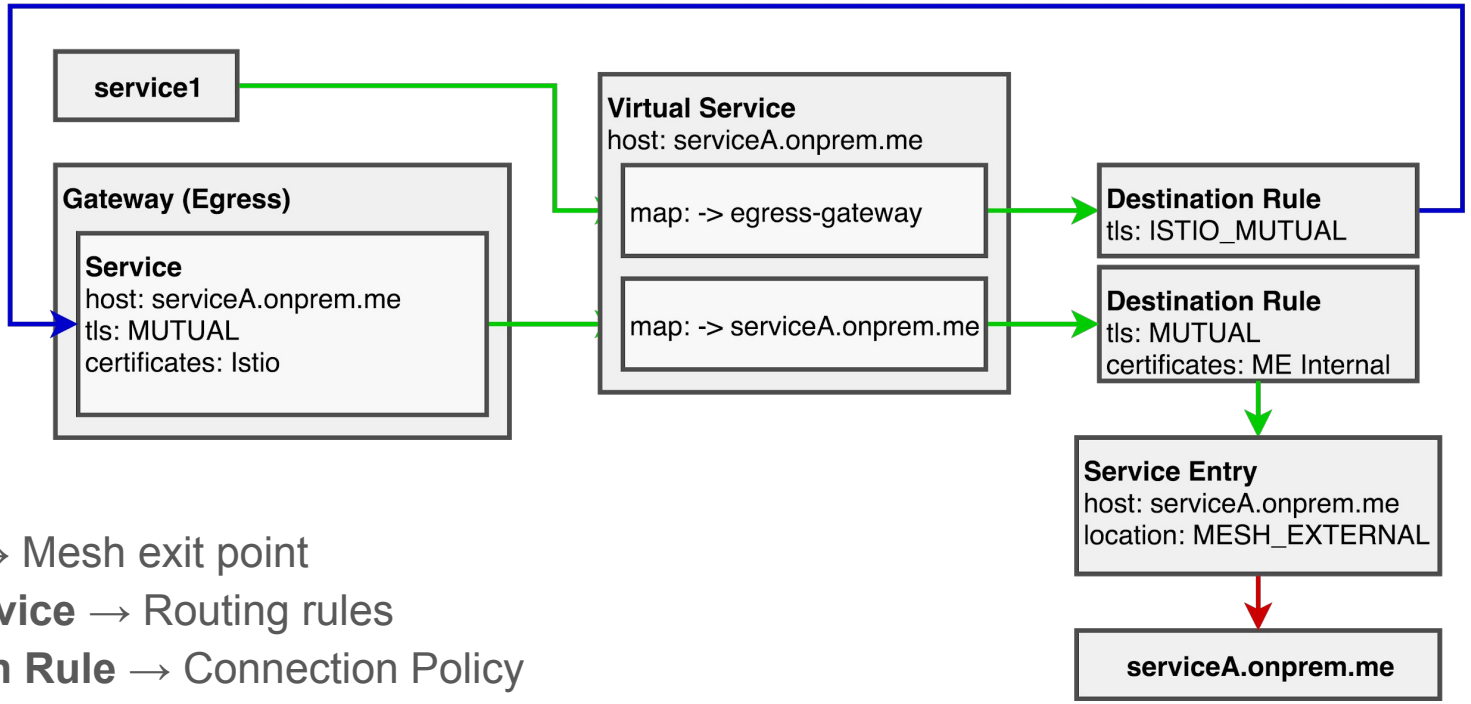
Topology - Canary



Istio supports canary deployments, but can’t route on content body.

Content Lifter “lifts” SOAP fields into HTTP headers.

Topology - Istio Egress



Resources

- **Gateway** → Mesh exit point
- **Virtual Service** → Routing rules
- **Destination Rule** → Connection Policy
- **Service Entry** → Manually registered host

Code and Demo ...

Takeaways

- Istio is ready. It has strong industry backing, and a healthy community.
- Just like Kubernetes, the complexity of Istio is justified for particular problem sizes. For a handful of endpoints it is difficult to justify. For hundreds of endpoints it is very compelling. Crossover point? Will reduce over time.
- Provides a solution that’s more resilient to change.
- Want to know more? We’d love to chat.

Thanks for your time!

Presentation & Samples: https://github.com/mantel-digio/istio-demo

Contact us:

- David Lochrie - david.lochrie@mebank.com.au
- Brett Henderson - brett.henderson@digio.com.au
- Ben Ebsworth - ben.ebsworth@digio.com.au