IMESH

CONF42

# Ambient Mesh : The new sidecar-less and faster Istio for zero-trust

**Md Azmal**
Founding Engineer
IMESH
/md-azmal-570308160

# About Us

Formed- 2023

HQ- Dover, US
Office- Bangalore, India

Offerings-

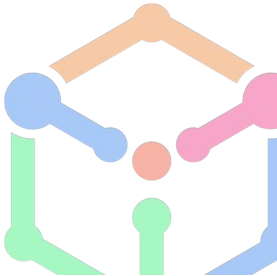
1. Enterprise Istio Support
2. IMESH Istio Dashboard

Mission-

Simplify and Secure the Network of Microservices in Cloud

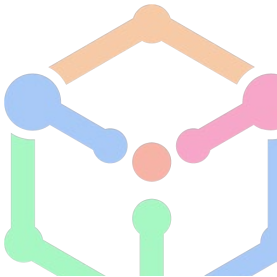
# Agenda


- Understanding Istio service mesh
- Limitations of sidecar based service mesh
- Understanding Istio Ambient mesh
- Security in the new Ambient mesh
- Benefits of using Istio Ambient mesh
- Demo
    - Enable Istio Ambient mesh with mTLS
    - Apply L4 and L7 authorization policies
    - Traffic management
    - Observability with open source tools




Meme source XD: https://www.mememaker.net/meme/buckle-up-were-gona-save-the-mesh


> A service mesh is a dedicated infrastructure layer that you can add to your applications. It allows you to transparently add capabilities like observability, traffic management, and security, without adding them to your own code



# Istio service mesh architecture

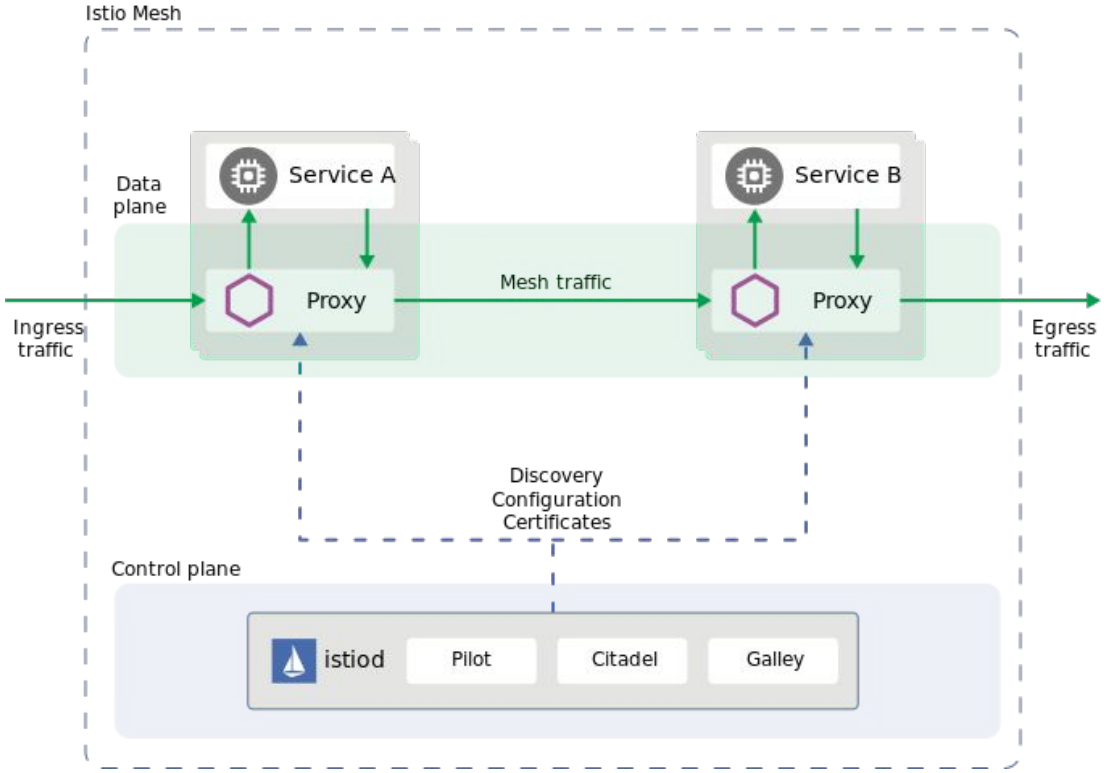



Image source: https://istio.io/latest/docs/ops/deployment/architecture/

# Limitations of sidecar

- Requires sidecar injection
- Sidecar modify pod specs and redirect traffic within the pod
- Sidecar updates require restarting the application
- Massive resource utilization, extra work to provision for worst case usage
- Traffic capture and HTTP processing is computationally expensive
- May result in breaking applications with non-conformant HTTP implementations
- Server first protocol may have impact on permissive mTLS

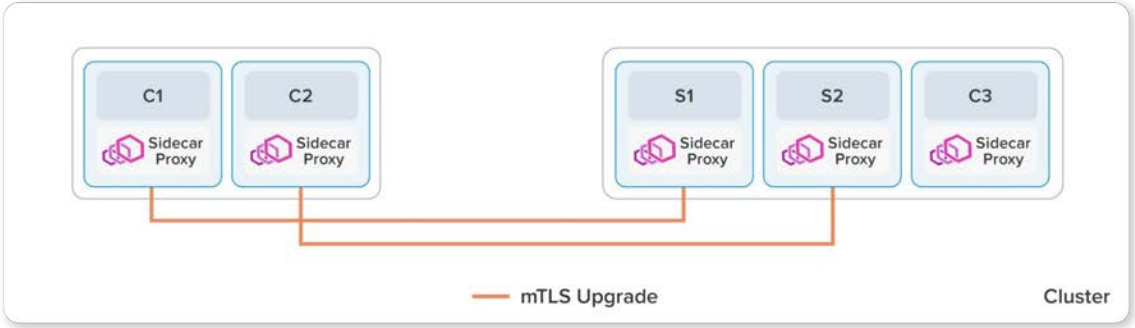
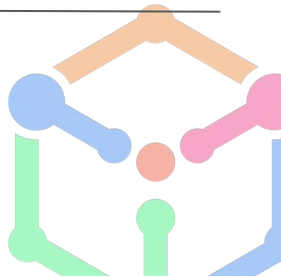


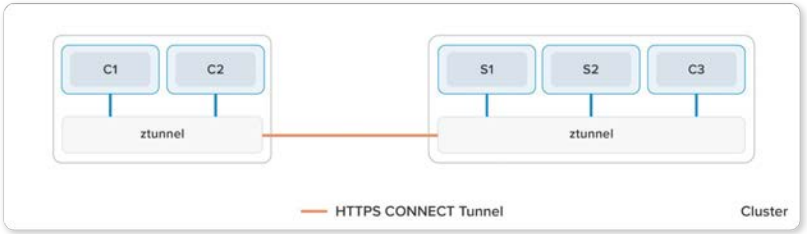

Image source: https://istio.io/latest/blog/2022/introducing-ambient-mesh/

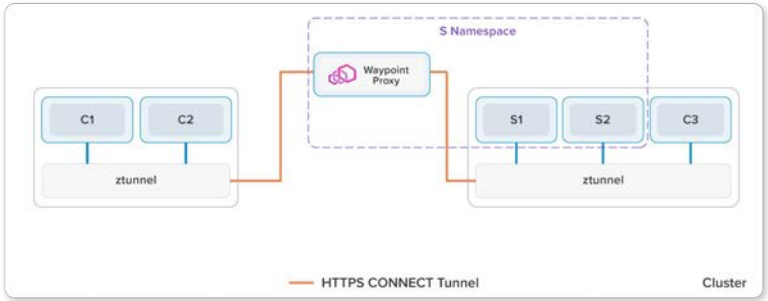
# Let’s break it down!


| | Security | Observability | Traffic Management |
|---|---|---|---|
| L4 | mTLS tunneling, Simple authorization policies | TCP metrics, logging | TCP routing |
| L7 | Rich authorization policies, HTTP restrictions and policies | HTTP metrics, access logging, tracing | HTTP routing and load balancing, circuit breaking, rate limiting, retry, timeouts and more |



# The Ambient Mesh 🚀



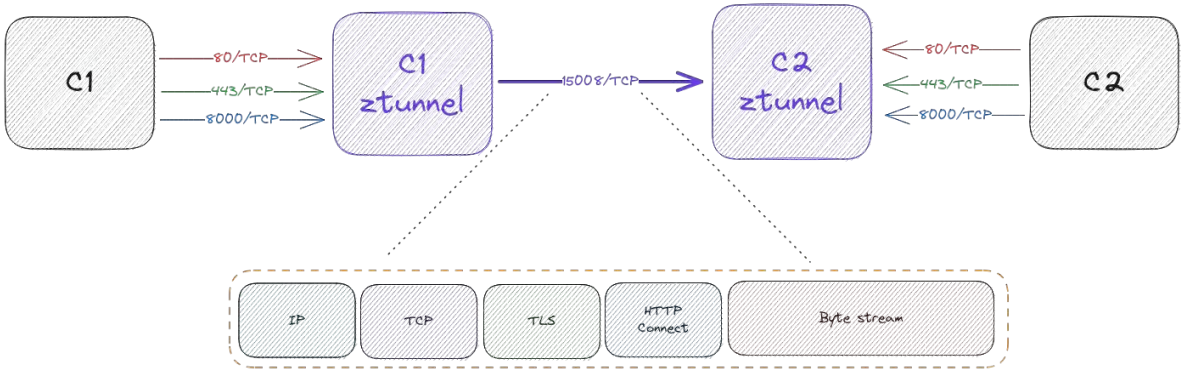




- ztunnel (zero-trust tunnel) securely connects and authenticates workloads in mesh
- Extremely lightweight as it does not do L7 processing
- Provides mTLS and L4 authorization without terminating or parsing HTTP

- Namespaces use waypoint proxy to implement istio capabilities and L7 processing
- ztunnel passes all L7 traffic through waypoint proxy
- Waypoint proxies are k8s pods that can be scaled as required

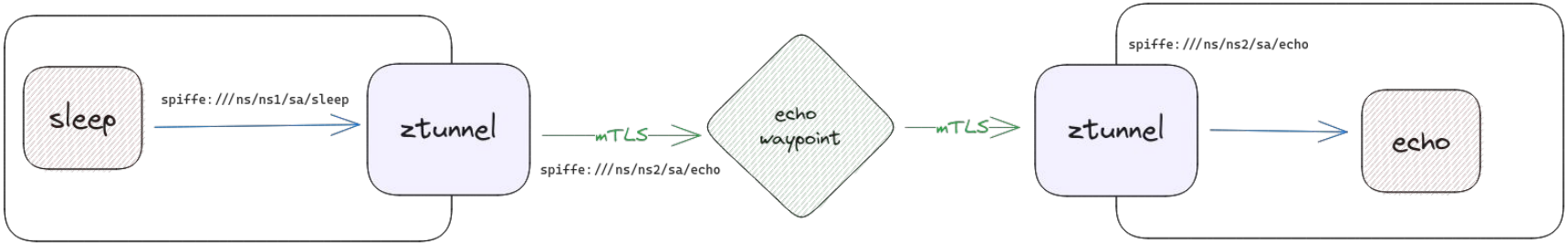
Image source: https://istio.io/latest/blog/2022/introducing-ambient-mesh/

# HTTPS CONNECT tunnel




- HBONE (HTTP-Based Overlay Network Environment)
- Cleaner encapsulation of traffic than TLS and interoperability with common load-balancer infrastructure
- Interoperability with sidecars does not limit security capabilities

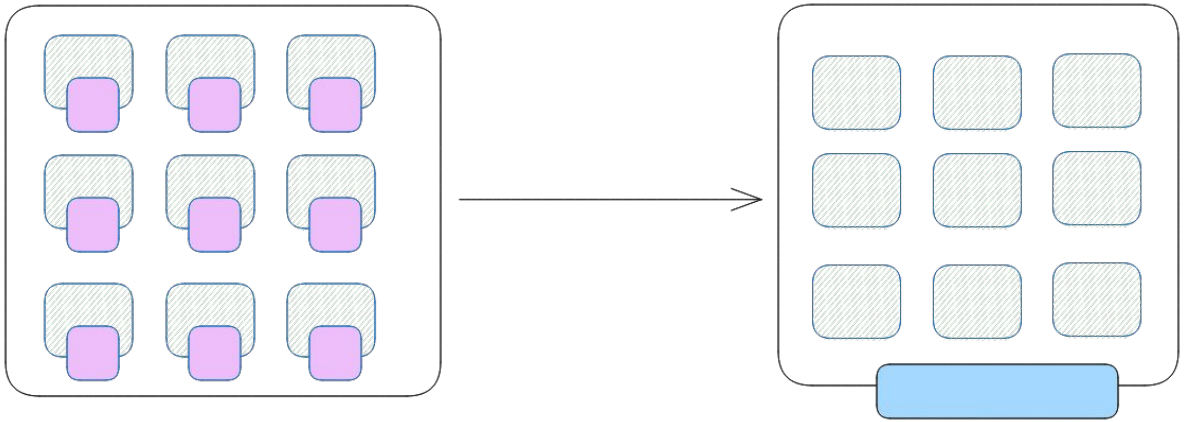
# Is ztunnel truly secure?




- CA (istiod) ensures certificates provided to ztunnel have associated workloads
- ztunnel assumes identity of workloads on the same node and acts as node agent
- Each SA has its own identity, CSR sent from ztunnel to istio control plane to get x.509 certificates
- Istio control plane acts as spiffe server to sign the certificates for workloads

# Benefits of ambient mesh

- Can be added without modifying existing workloads
- Zero downtime
- Minimal overhead cost
- Optimal resource utilization
- Interoperability with sidecar based istio and non-istio workloads

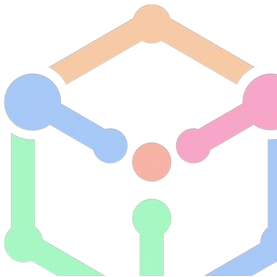



https://github.com/MD-AZMAL/ambient-mesh-deep-dive

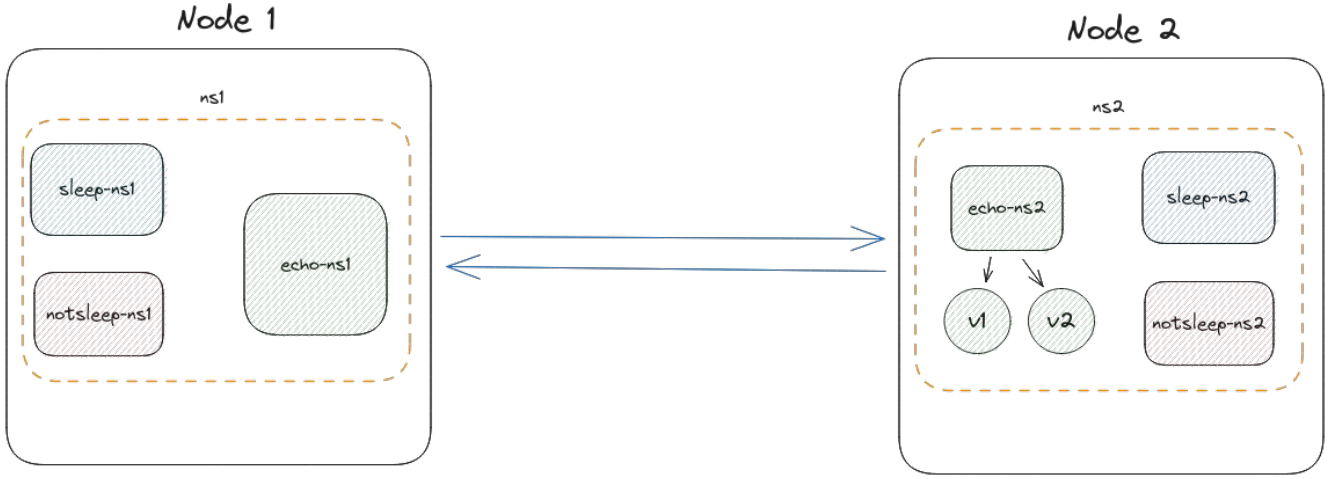
Meme source XD: https://makeameme.org/meme/demo-time-5bb733

# Cluster setup


- Azure CNI
- AKS cluster
- 2 x Standard B2ms (2 vCPUs, 8 GiB memory) nodes
- Istio version 1.18.3



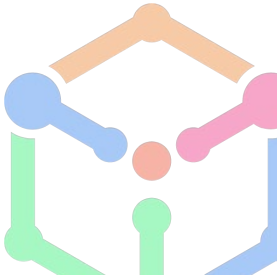
# Setup the workloads



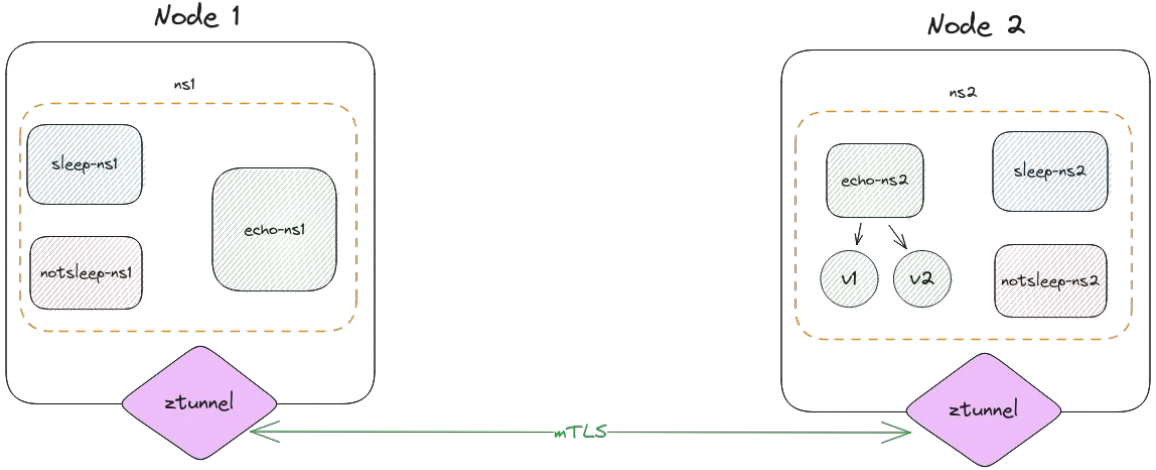

- `kubectl label nodes <your-node> name=node1`
- `kubectl label nodes <your-node> name=node2`
- `kubectl apply -f resources-ns1.yaml`
- `kubectl apply -f resources-ns1.yaml`
- `kubectl exec deploy/sleep-depl-ns1 -n ns1 -- curl -s http://echoserver-service-ns2.ns2.svc.cluster.local`

# Install Istio with ambient mesh

- `istioctl install --set profile=ambient --set components.ingressGateways[0].enabled=true --set components.ingressGateways[0].name=istio-ingressgateway --skip-confirmation`
- `kubectl get crd gateways.gateway.networking.k8s.io &> /dev/null || \ { kubectl kustomize "github.com/kubernetes-sigs/gateway-api/config/crd/experimental?ref=v0.8.0-rc1" | kubectl apply -f -; }`
- `kubectl apply -f samples/addons/prometheus.yaml`
- `kubectl apply -f samples/addons/grafana.yaml`
- `kubectl apply -f samples/addons/kiali.yaml`

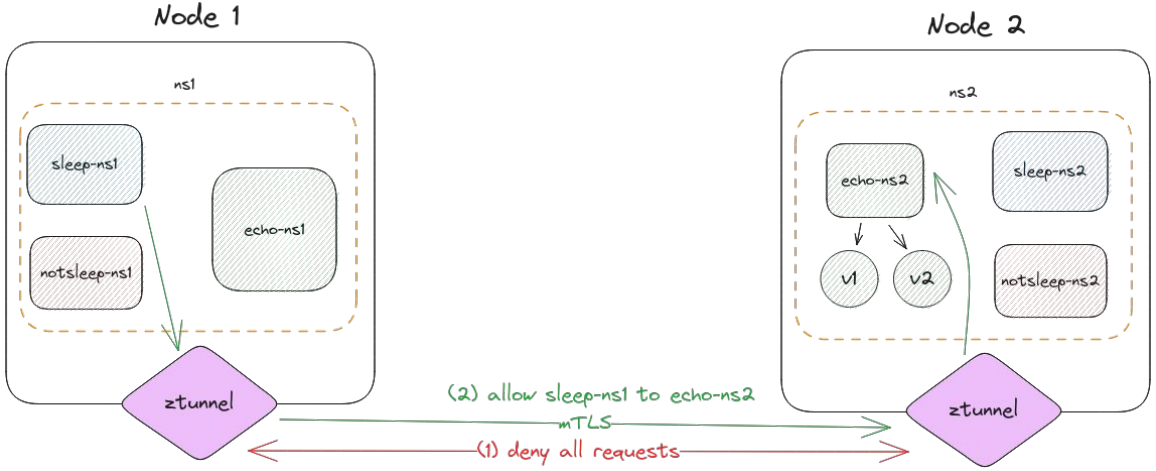


# Enable Ambient mesh



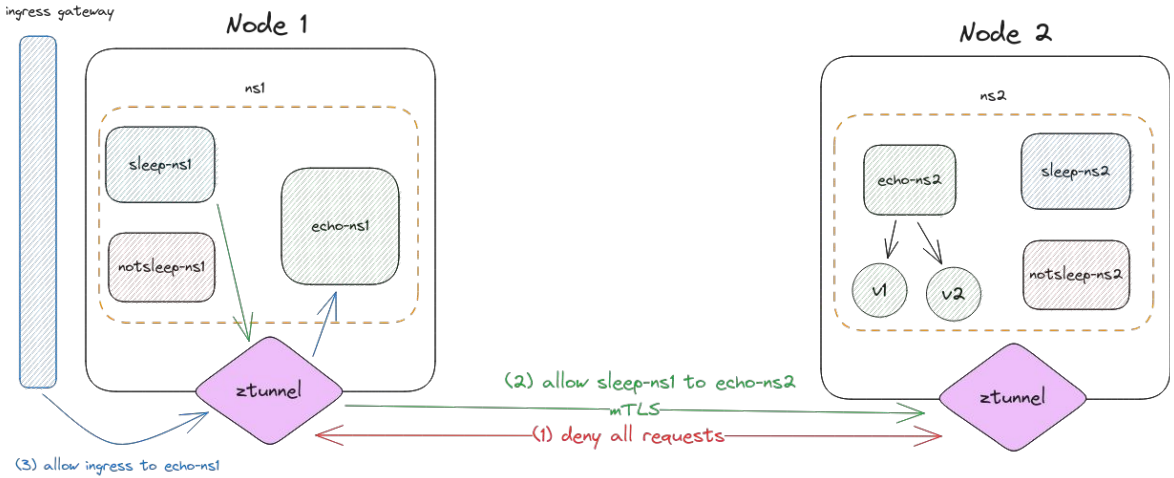

- `kubectl label namespace ns1 istio.io/dataplane-mode=ambient`
- `kubectl label namespace ns2 istio.io/dataplane-mode=ambient`
- `kubectl exec deploy/sleep-depl-ns1 -n ns1 -- curl -s http://echoserver-service-ns2.ns2.svc.cluster.local`
- `kubectl logs -f <your-ztunnel-pod> -n istio-system`

# Apply L4 authorization policies



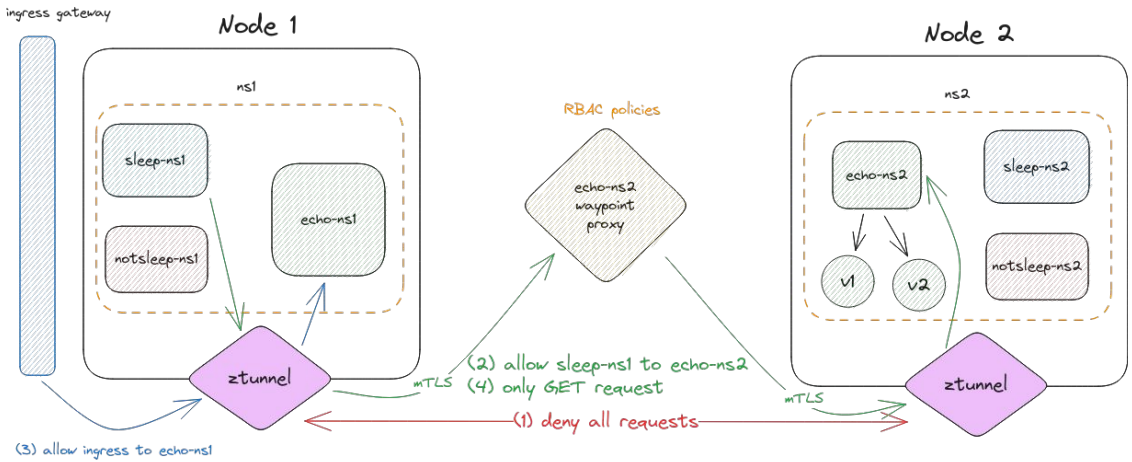

- `kubectl apply -f deny-all-l4.yaml`
- `kubectl exec deploy/sleep-depl-ns1 -n ns1 -- curl -s http://echoserver-service-ns2.ns2.svc.cluster.local`
- `kubectl apply -f allow-comms-echo-ns2.yaml`
- `kubectl exec deploy/sleep-depl-ns2 -n ns2 -- curl -s http://echoserver-service-ns1.ns1.svc.cluster.local`

# Allow traffic through ingress gateway



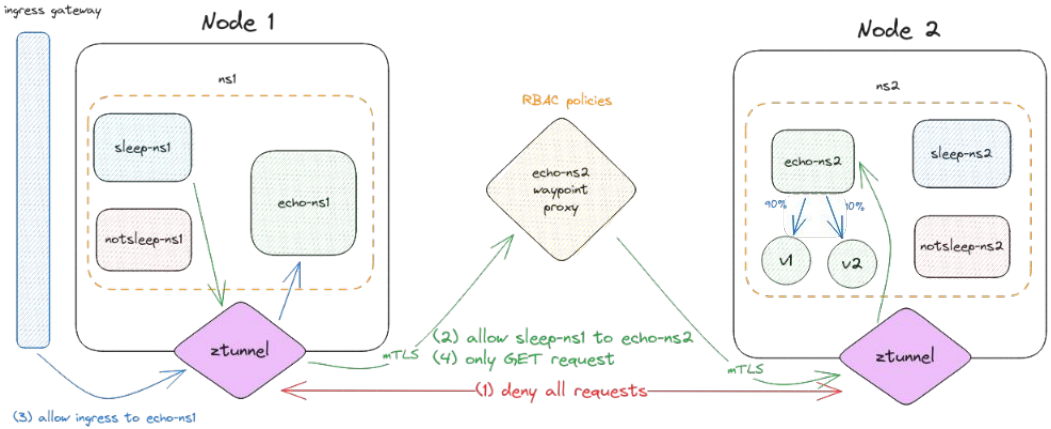

- `kubectl apply -f istio-gateway.yaml`
- `kubectl get svc -n istio-system`
- `curl -v http://<your-gateway-external-ip>`
- `kubectl apply -f allow-comms-echo-ns1.yaml`

# Apply L7 authorization policy



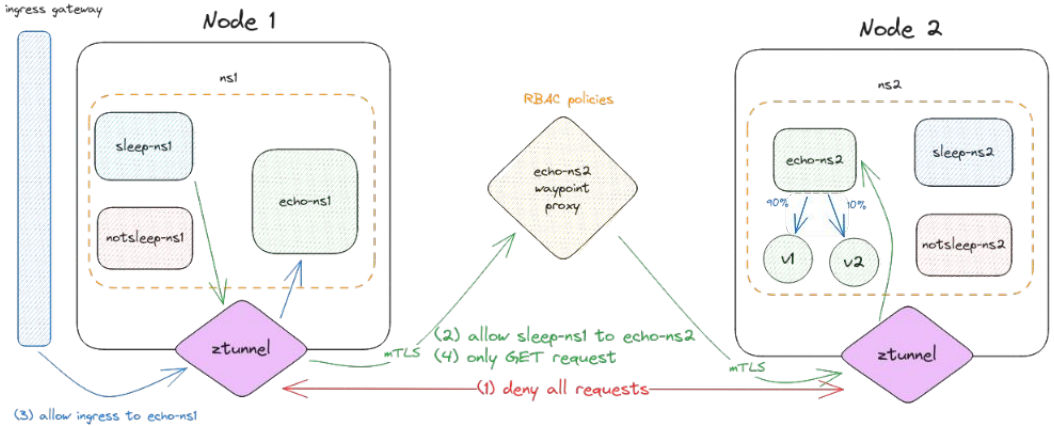

- `istioctl x waypoint generate -n ns2 -s echo-service-account-ns2`
- `kubectl apply -f waypoint-proxy.yaml`
- `kubectl apply -f allow-get-to-echo-ns2.yaml`
- `kubectl exec deploy/sleep-depl-ns1 -n ns1 -- curl -s http://echoserver-service-ns2.ns2.svc.cluster.local`
- `kubectl exec deploy/sleep-depl-ns1 -n ns1 -- curl -s http://echoserver-service-ns2.ns2.svc.cluster.local -X POST`

# Traffic management: Canary release




- `kubectl apply -f canary.yaml`
- `kubectl exec deploy/sleep-depl-ns1 -n ns1 -- sh -c 'for i in $(seq 1 100); do curl -s http://echoserver-service-ns2.ns2.svc.cluster.local; done| grep -c echoserver-depl-ns2-v1'`
- `kubectl exec deploy/sleep-depl-ns1 -n ns1 -- sh -c 'for i in $(seq 1 100); do curl -s http://echoserver-service-ns2.ns2.svc.cluster.local; done| grep -c echoserver-depl-ns2-v2'`

# Observability and Debugging




kiali

Grafana

- istioctl dashboard grafana
- istioctl dashboard prometheus
- istioctl dashboard kiali
- istioctl ps
- istioctl pc all <your-pod>.<namespace>


Meme source XD: https://www.idlememe.com/good-job-meme-52/

IMESH

CONF42

# Thank You

**Md Azmal**
Founding Engineer
IMESH
/md-azmal-570308160