CLOUDNATIVE SECURITYCON

NORTH AMERICA 2023

CLOUDNATIVE SECURITYCON
NORTH AMERICA 2023

Network Security at Scale

Bernard Van De Walle, Splunk
Mitch Connors, Aviatrix

Meet Your Speakers

CLOUDNATIVE SECURITYCON NORTH AMERICA 2023

Bernard Van De Walle
Principal software engineer, Splunk
K8s, Istio, Envoy operations at scale
Previously at Cruise

Mitch Connors
Principal software engineer, Aviatrix
Istio Member since 2018
Istio TOC

splunk®>cloud™

- **Splunk Cloud**: Cloud Native splunk
- Scale: ~ 35 K8s clusters
  - Distributed across all regions
  - AWS and GCP
- Cloud agnostic, K8s, Istio, Envoy

AGENDA

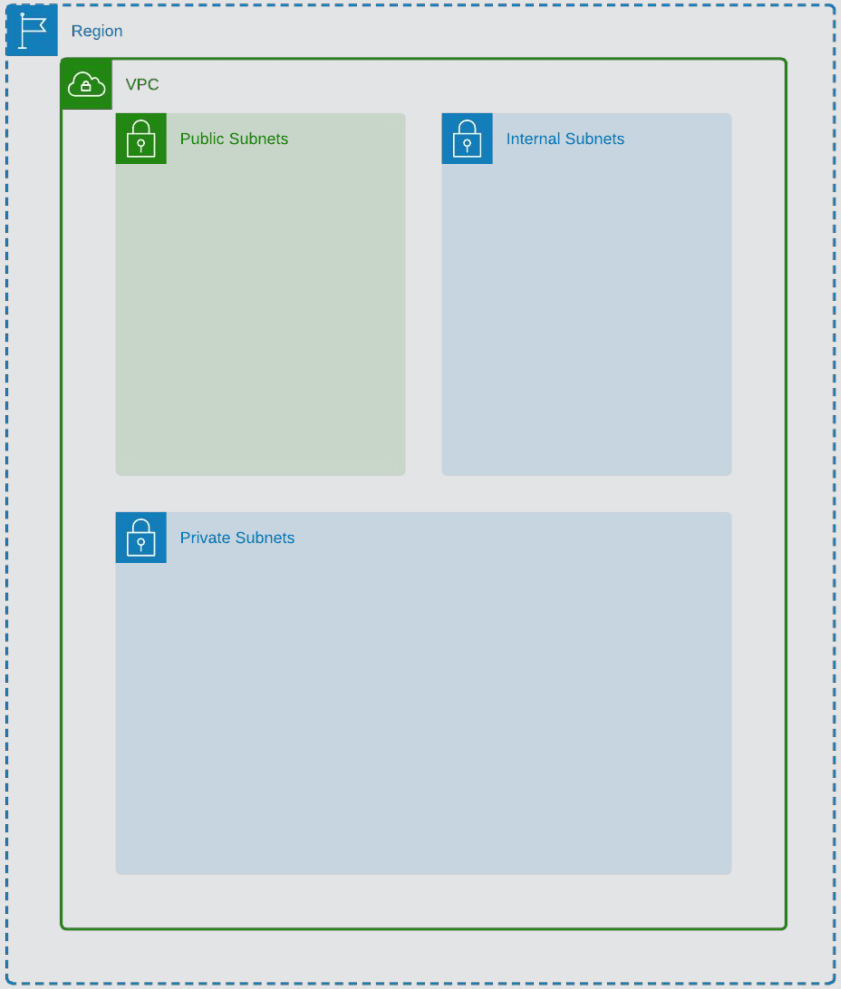
CLOUDNATIVE SECURITYCON
NORTH AMERICA 2023

- L3/L4: Cloud Provider
  - VPC
  - Network Load Balancer
- L3/L4: Kubernetes
  - Services
  - Network Policies
- L7: Istio/Envoy
  - AuthN/AuthZ

Cloud Provider L3/L4

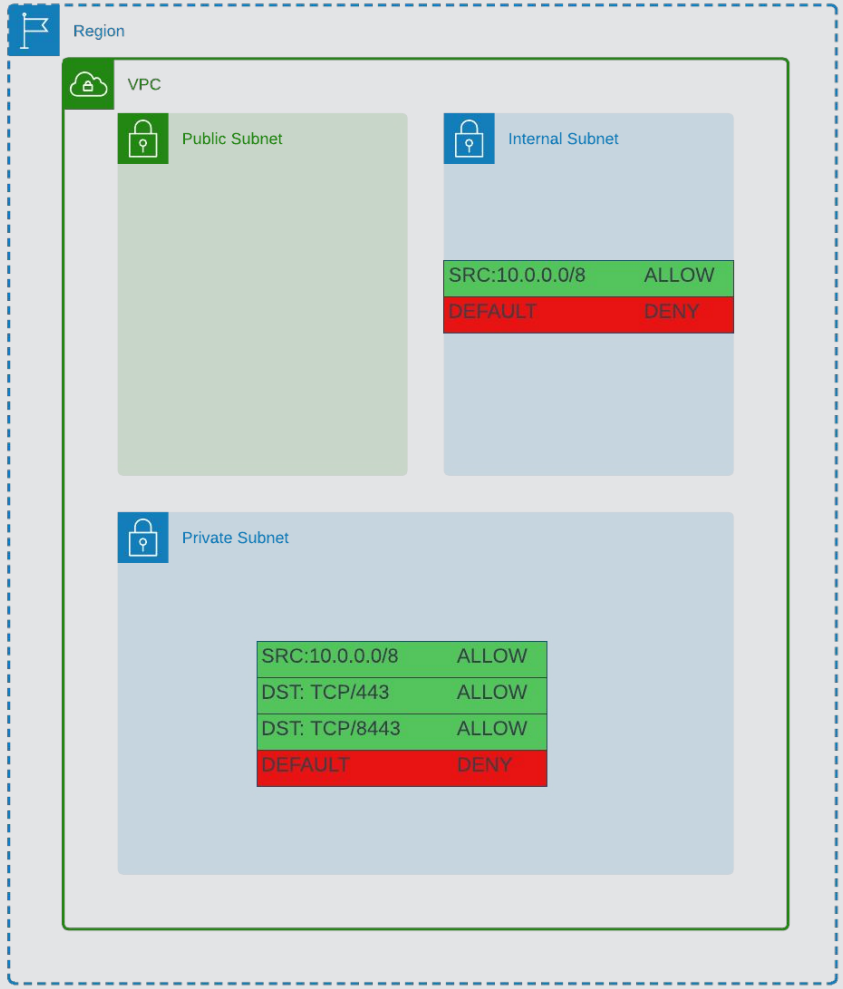
Standard VPC

- Isolated standard VPC
- 3 set of subnets:
  - **Private** for Kubernetes workloads
  - **Public** for Internet connectivity
  - **Internal** for Splunk connectivity



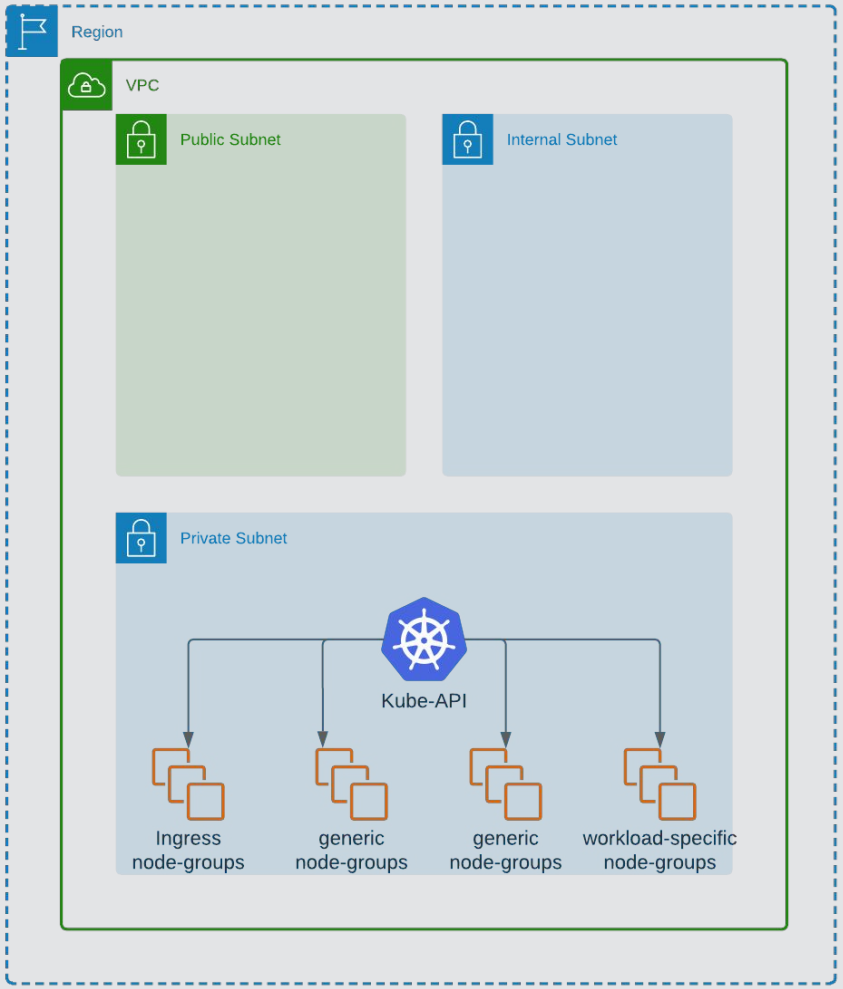
Network ACLs

- Stateless
- Applied by subnet
- Basic L3/L4 capabilities
- Provides a catch-all last resort set of rules
- Example:
  - Internal subnet SRC 10.x/8



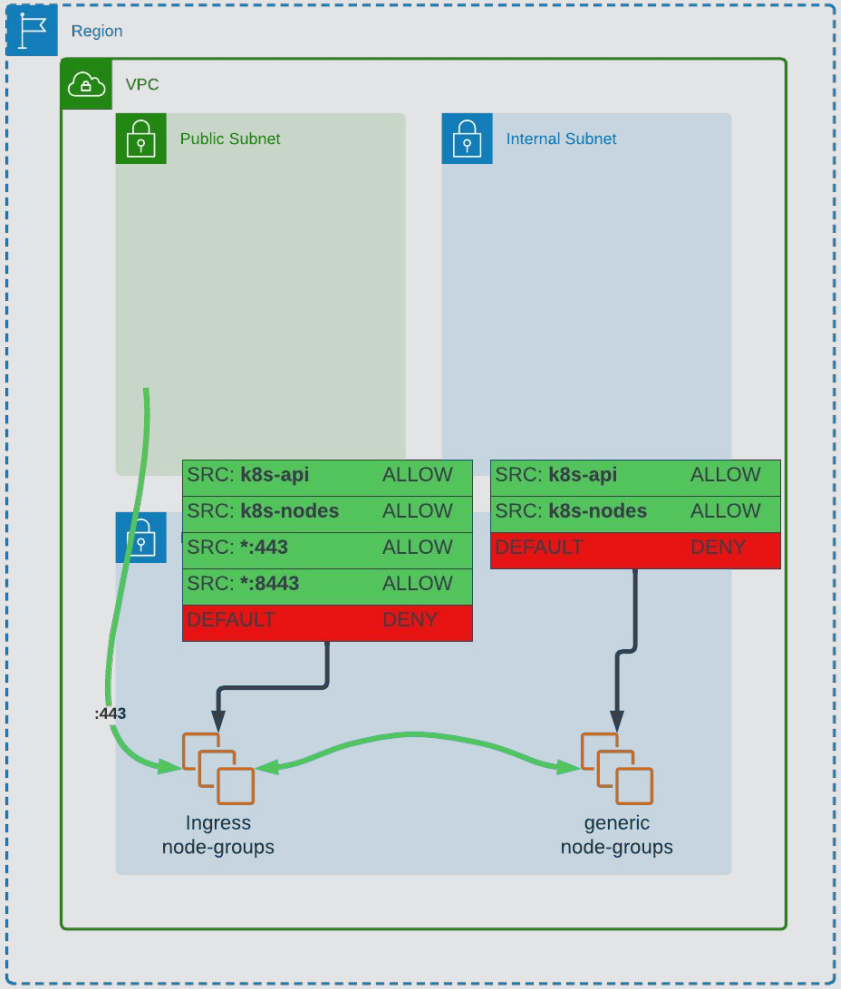
Kubernetes nodes

- Cluster nodes deployed on private IP space
- Multiple node-groups:
  - Ingress (Gateways)
  - Generic
  - Workload-specific



Security Groups

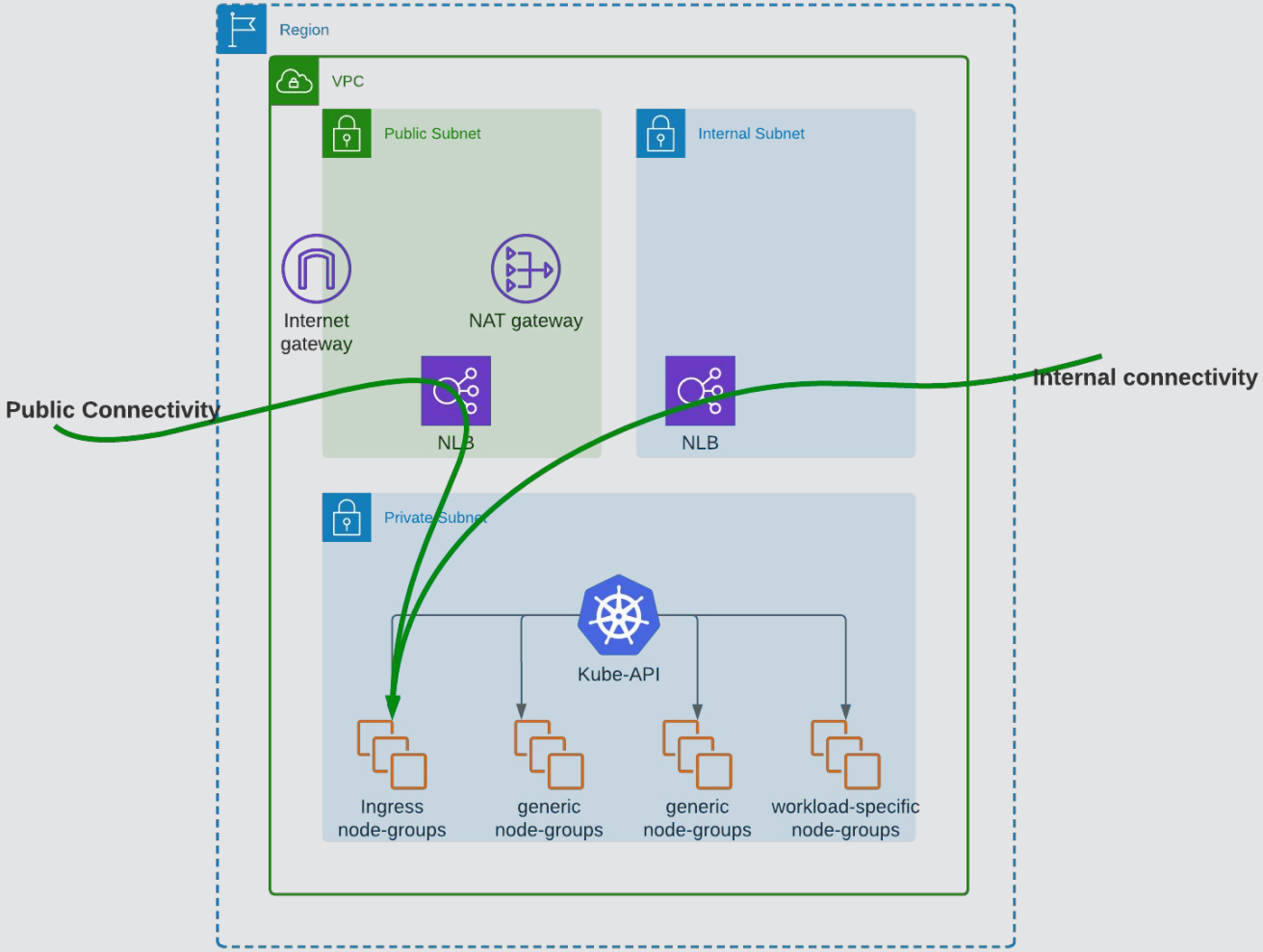
- Stateful
- Applied per instance
- Allow for fine-grained traffic across specific instances
- Example:
  - Ingress node-group -> Generic node-group



Ingress connectivity

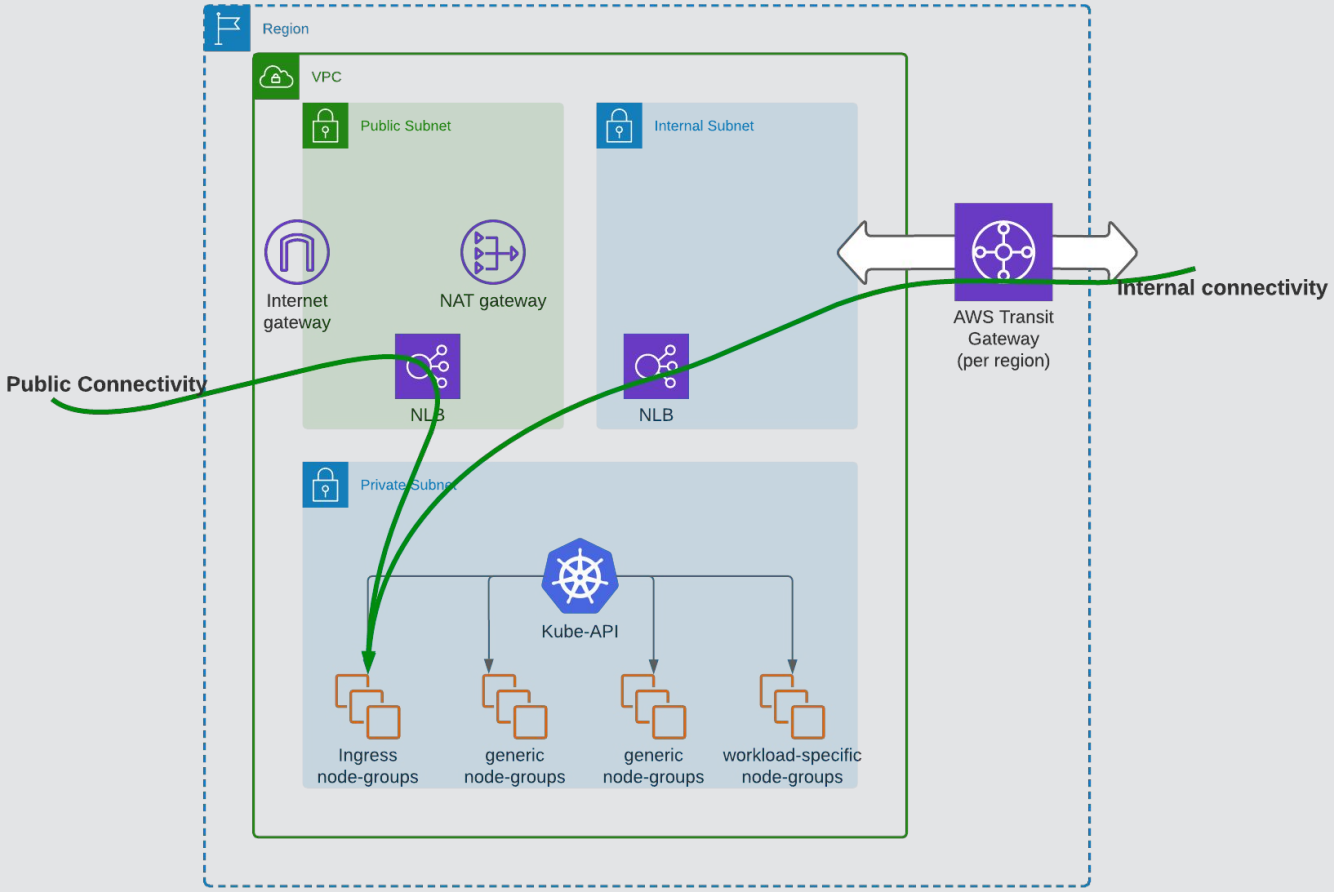
- Connectivity through NLBs
  - Public NLBs
  - Internal NLBs

- Connectivity through Ingress Gateways (Envoy/Istio)



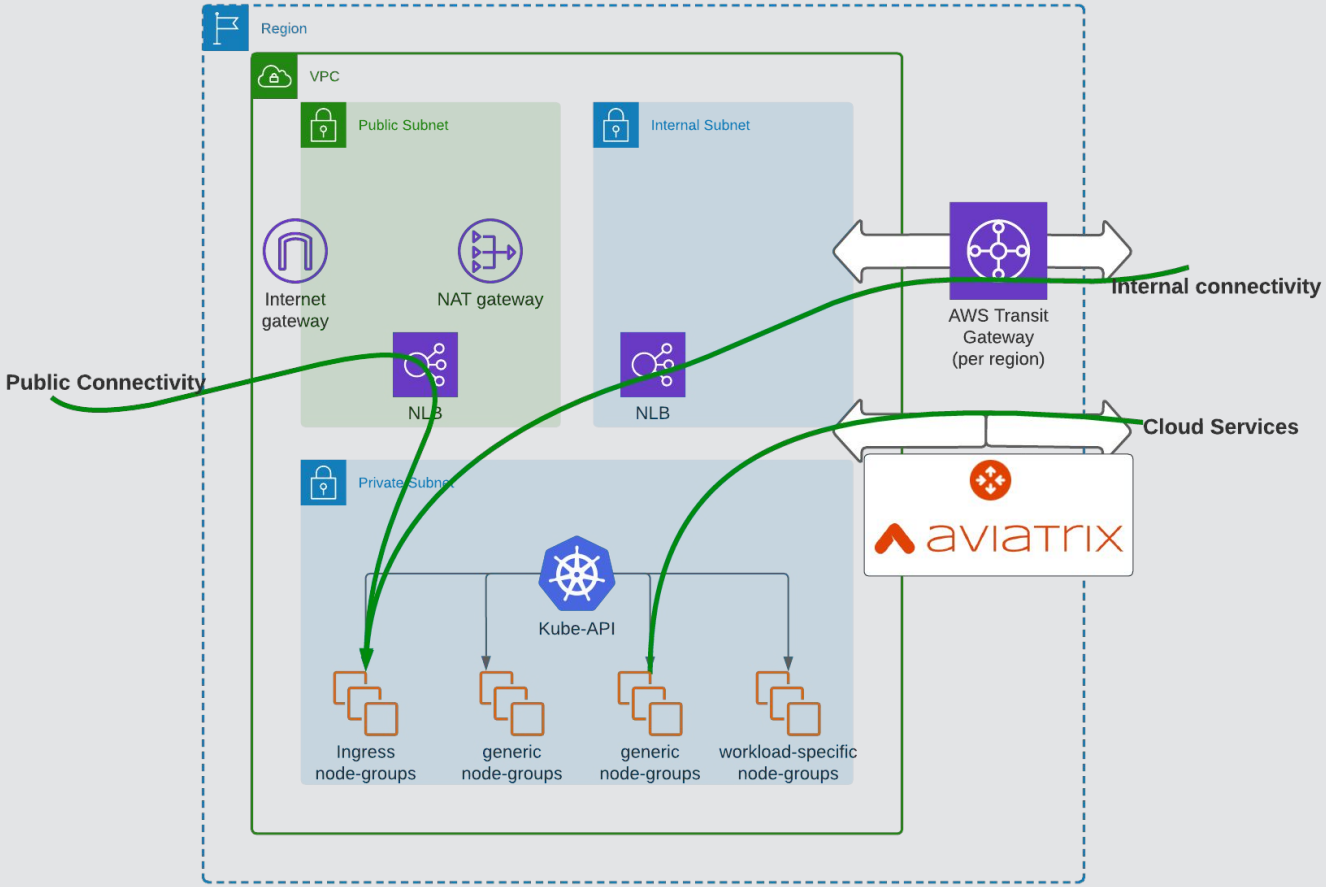
Internal connectivity (transit gateway)

- Internal connectivity through AWS transit gateway
  - Internal SRC IP advertised only
  - Connected to Splunk firewall



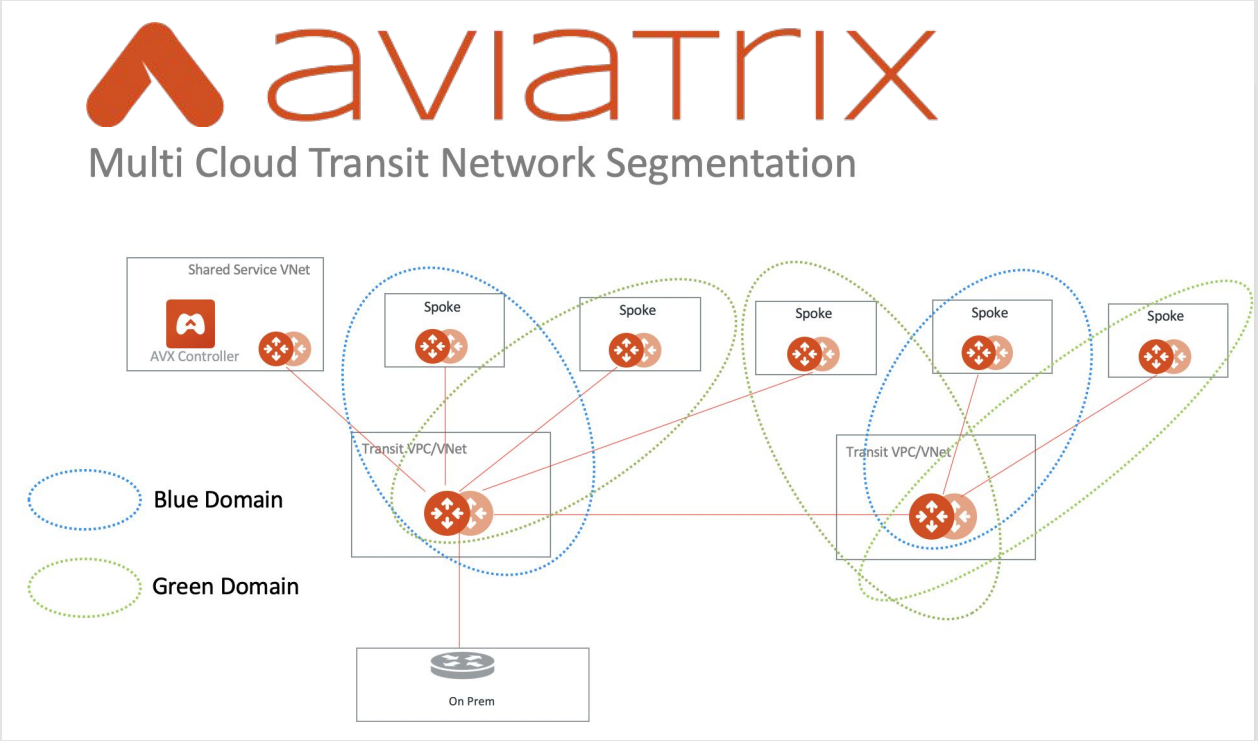
Cloud connectivity (Aviatrix)

- Cloud services connectivity through Aviatrix
  - Flat network
  - Supports overlapping IPs



Aviatrix Network Domains

- By Default All connected domains are routable in flat network across clouds, regions, on-prem
- Network Domains limit routes
- Prod can talk to Shared, Dev can talk to Shared, Dev cannot talk to Prod

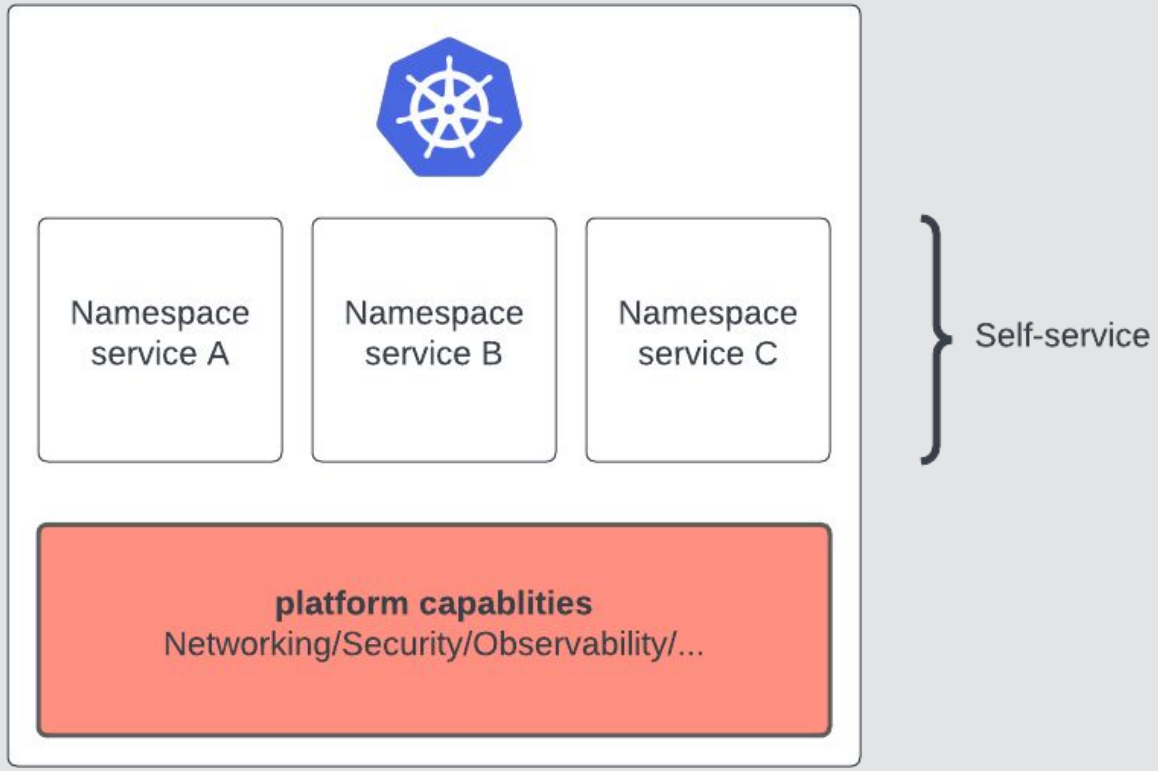


Kubernetes

L3/L4

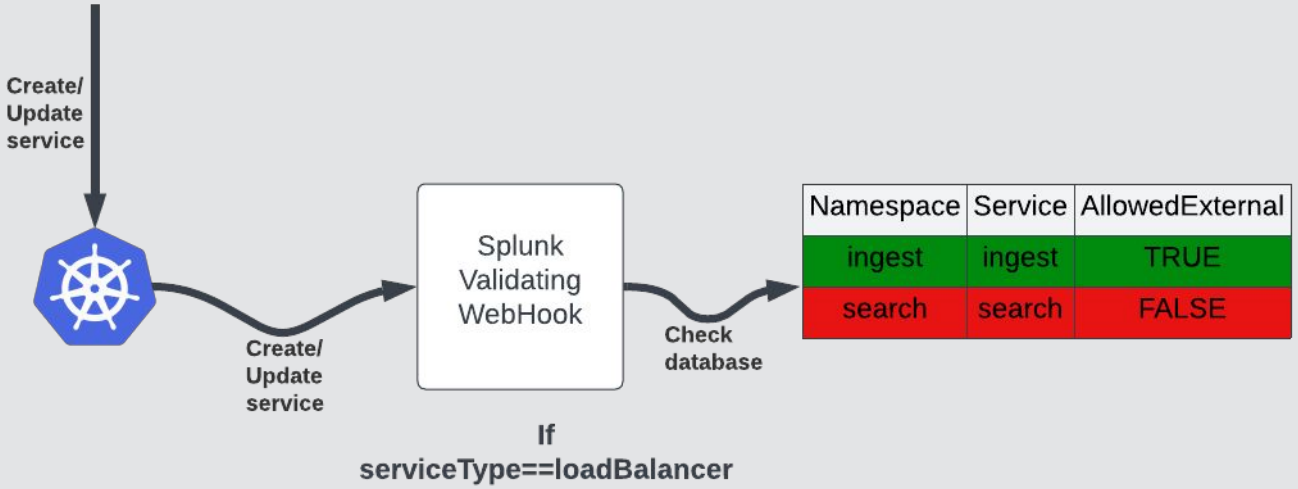
Kubernetes deployment

- Self-Service platform
- External connectivity **ONLY** through NLBs
- Pod to pod connectivity and Network policies through Calico



ValidatingWebhook

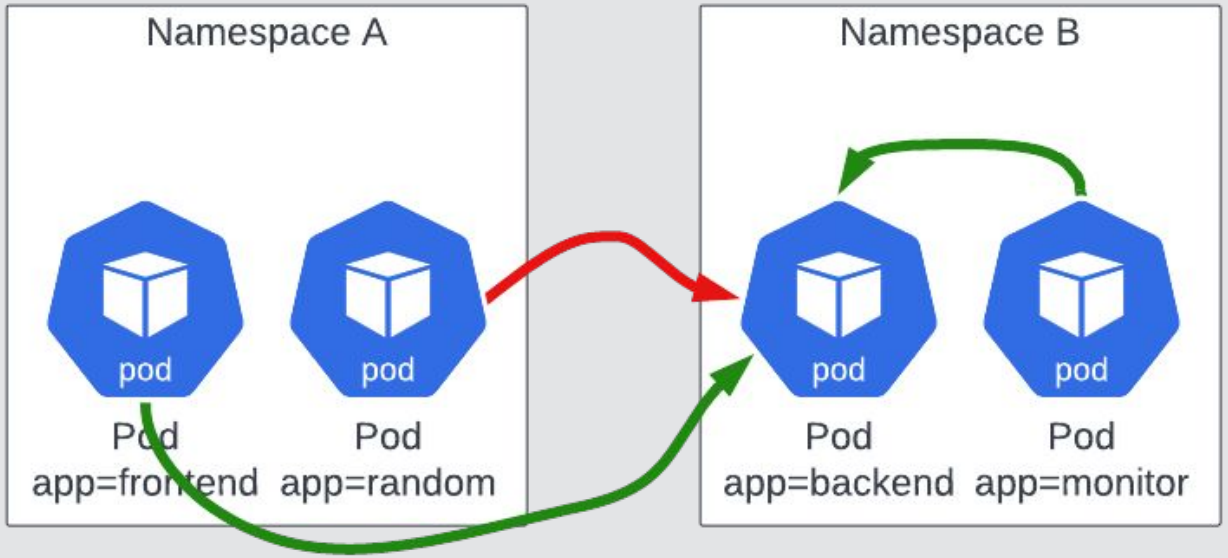
- Splunk validating webhook
- Denies service type load-balancer
- Plenty of open implementations
  - OPA
  - K-Rail(https://github.com/cruise-automation/k-rail)



NetworkPolicies on K8S

- K8S-centric,L3/L4, stateful
- Implemented by your CNI plugin (Cilium,Calico,...)

```yaml
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
spec:
  podSelector:
    matchLabels:
      app: backend
  ingress:
    - from:
        - podSelector:
            matchLabels:
              app: monitor
    - from:
        - namespaceSelector:
            matchLabels:
              name: frontend
          podSelector:
            matchLabels:
              app: frontend
```



Istio

L7

Istio Service Mesh

- Application Layer Networking
  - All POST requests from svc A route to this subset of svc B
  - What is the success latency from svc A to svc B?
  - svc A may send GET requests to svc B only at path /foo/*/bar
- How do we do it?

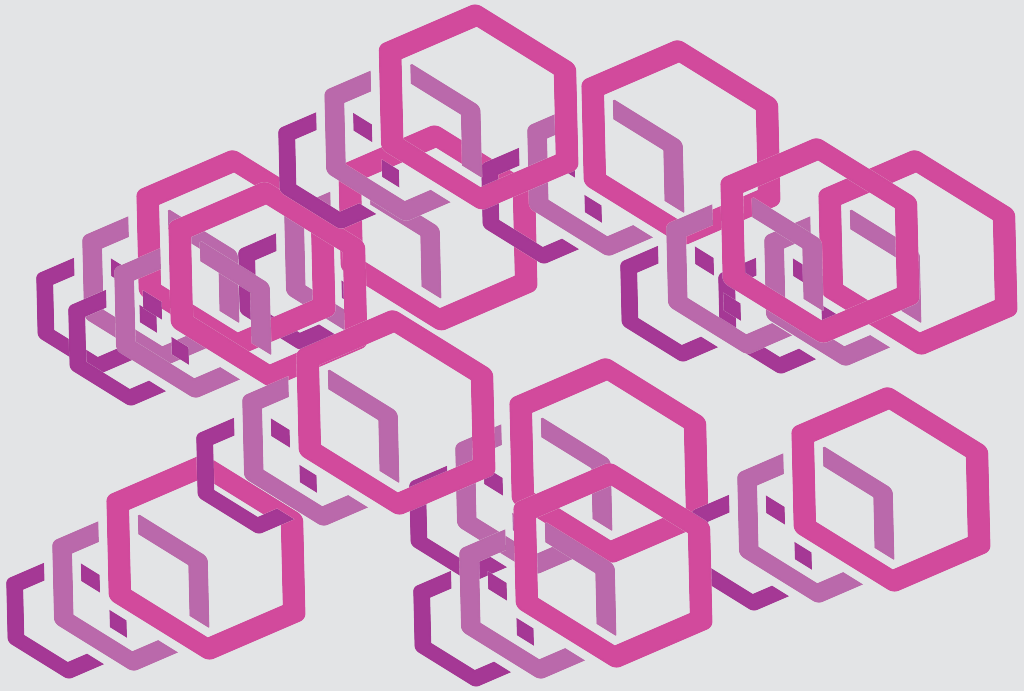
Traffic Routing

Observability

Security

Istio Service Mesh

=



In-transit encryption

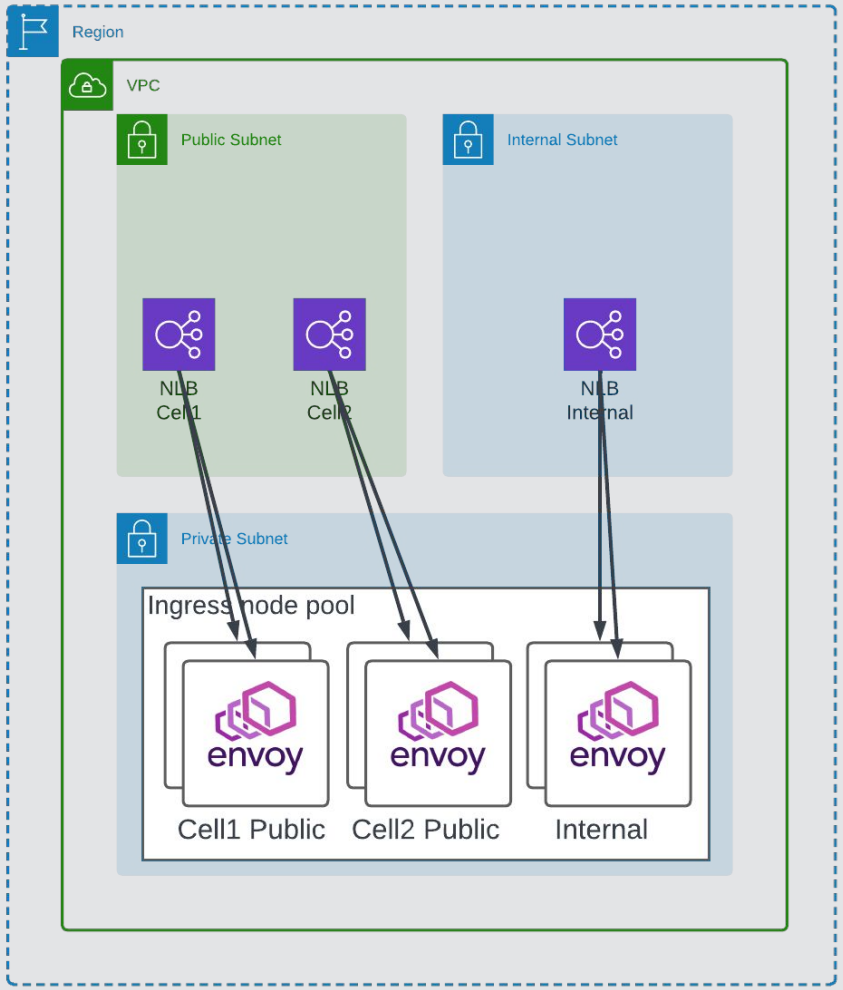
- Run the mesh in **permissive** mode
- Monitor the **passthrough cluster** and alert teams not using mTLS
- Alternative to **strict** mode

Metrics

```
sum(federate:istio_requests_total:sum_rate2m{
   reporter="source",
   k8s_cluster="$cluster",
   destination_service_name="PassthroughCluster"})
by (destination_service)
```

Gateway provisioning

- Ingress only through NLB/gateways
  - For all types of traffic (HTTP/TCP)
- NLB/Gateways per
  - Ingress source (public/internal)
  - Workload types/Blast radius
- Gateway provisioned in **istio-gateway** namespace



Ingress setup

- Gateway CRD
  - Istio-gateway namespace
- VirtualService CRD
  - Workload namespace
- Certificate through Let’sEncrypt
- Gated through ValidationWebhook

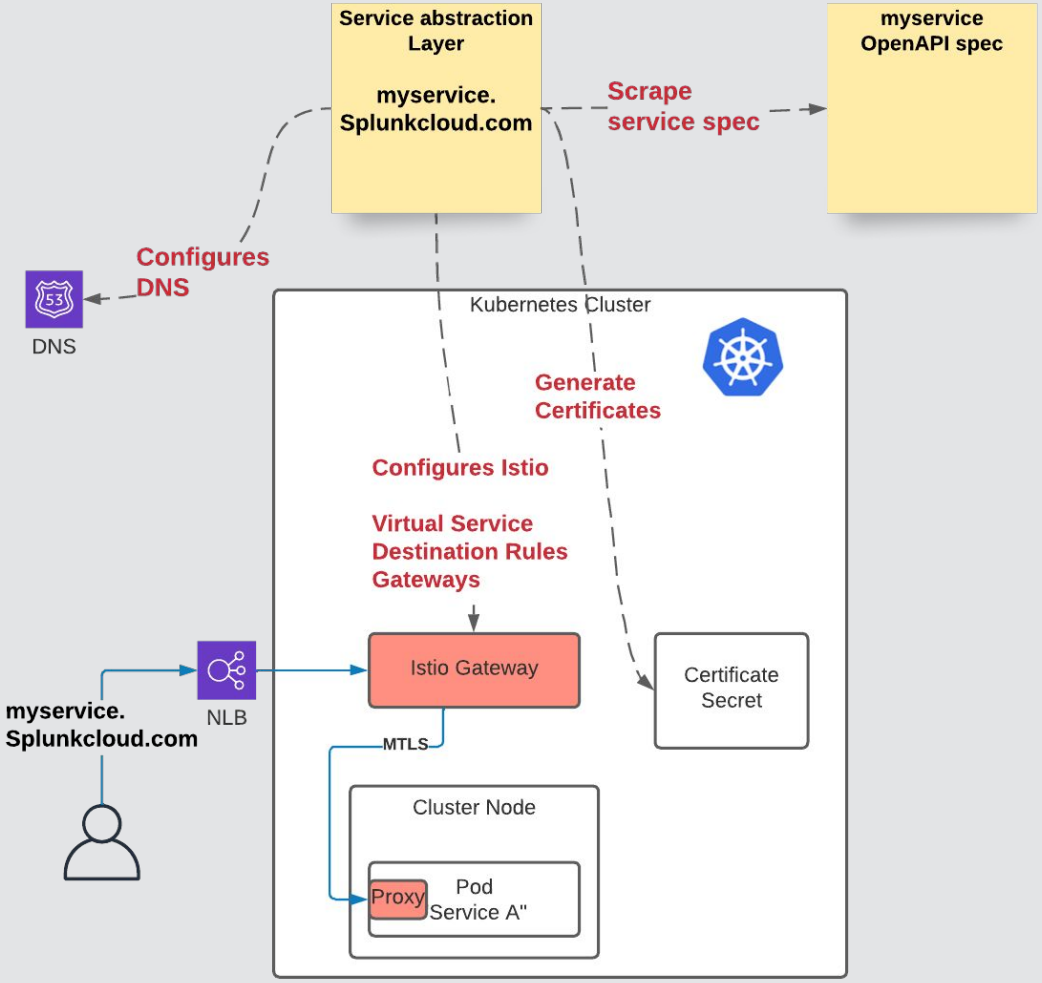

```yaml
apiVersion: networking.istio.io/v1beta1
kind: Gateway
  namespace: istio-gateway
spec:
  selector:
    istio: ingressgateway
  servers:
  - hosts:
    - myservice.splunkcloud.com
    port:
      number: 443
      protocol: HTTPS
    tls:
      credentialName: myservice-cert
```

```yaml
apiVersion: networking.istio.io/v1beta1
kind: VirtualService
metadata:
  namespace: myservice
spec:
  gateways:
  - istio-gateway/myservice
  hosts:
  - myservice.splunkcloud.com
  http:
  - route:
    - destination:
        host: myservice.myservice.svc.cluster.local
        port:
          number: 8443
```

```yaml
apiVersion: cert-manager.io/v1
kind: Certificate
metadata:
  name: acs-cert
  namespace: istio-gateway
spec:
  dnsNames:
  - myservice.splunkcloud.com
  issuerRef:
    kind: ClusterIssuer
    name: letsencrypt
  secretName: myservice-cert
```

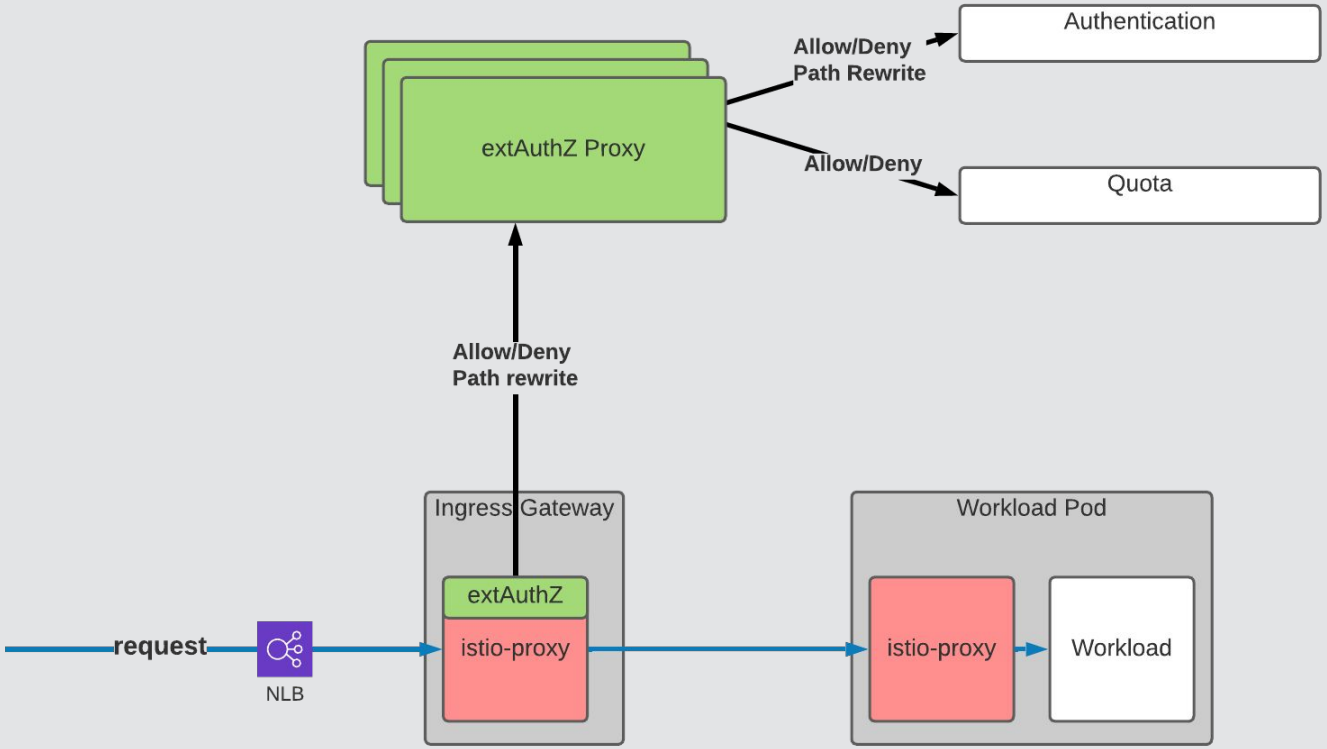
Service abstraction layer

- “Golden path” abstraction layer for 80% of the use cases
- A single abstraction layer for:
  - **VirtualServices**, **DestinationRules**, **Gateways** and **ServiceEntry** CRD
  - Certificate management
  - DNS management
- OpenAPI spec per service
- Abstraction Layer controller scrapes those openAPI specs



Layer7 Authentication

- Gateways authenticate requests through envoy ExtAuthZ
- extAuthZ-proxy allows plugins by adding them inline
- Bogus requests are blocked on the gateway
- More info: [External Authorization — envoy 1.26.0-dev-7cc893 documentation]



IP/HTTP AllowList with Istio

- AllowLists on L4-L7
- Mix and match IP and HTTP concepts

```yaml
apiVersion: security.istio.io/v1beta1
kind: AuthorizationPolicy
spec:
  action: ALLOW
  rules:
  - from:
    - source:
        ipBlocks:
        - 1.2.3.4
    to:
    - operation:
        methods:
        - GET
        paths:
        - /myservice/api/v1/*
  selector:
    matchLabels:
      istio: ingressgateway-default
```

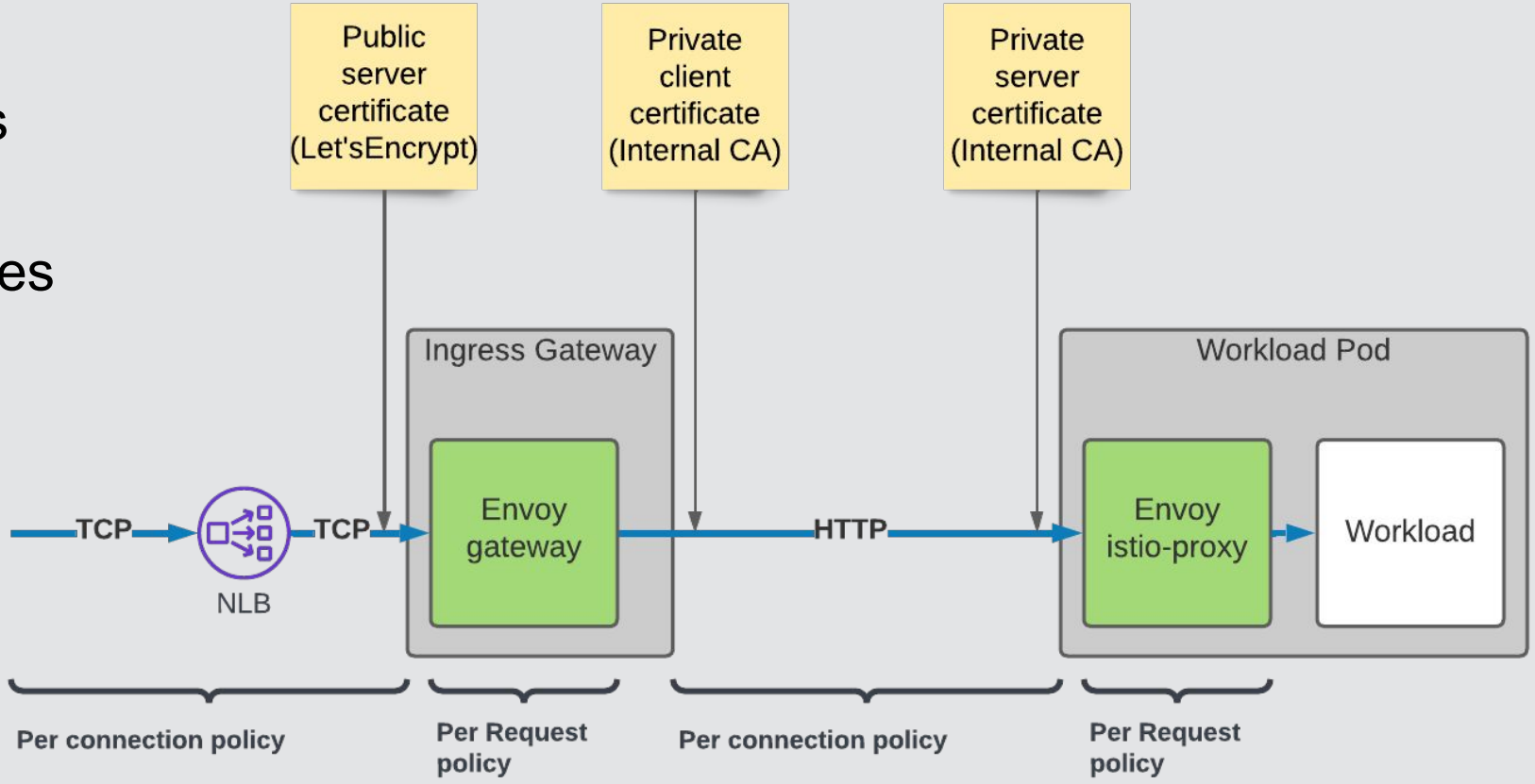
JWT Auth with Istio

- JWT validation in Istio

```yaml
apiVersion: security.istio.io/v1beta1
kind: RequestAuthentication
metadata:
  namespace: myservice
spec:
  jwtRules:
  - issuer: vault.splunkcloud
    jwks: '{ "keys": [ {"kty": "RSA", "e": "AQAB","use":"sig", "alg": "RS256", "n": "..."} }
  selector:
    matchLabels:
      app.kubernetes.io/name: myservice
```

Life of an ingress request

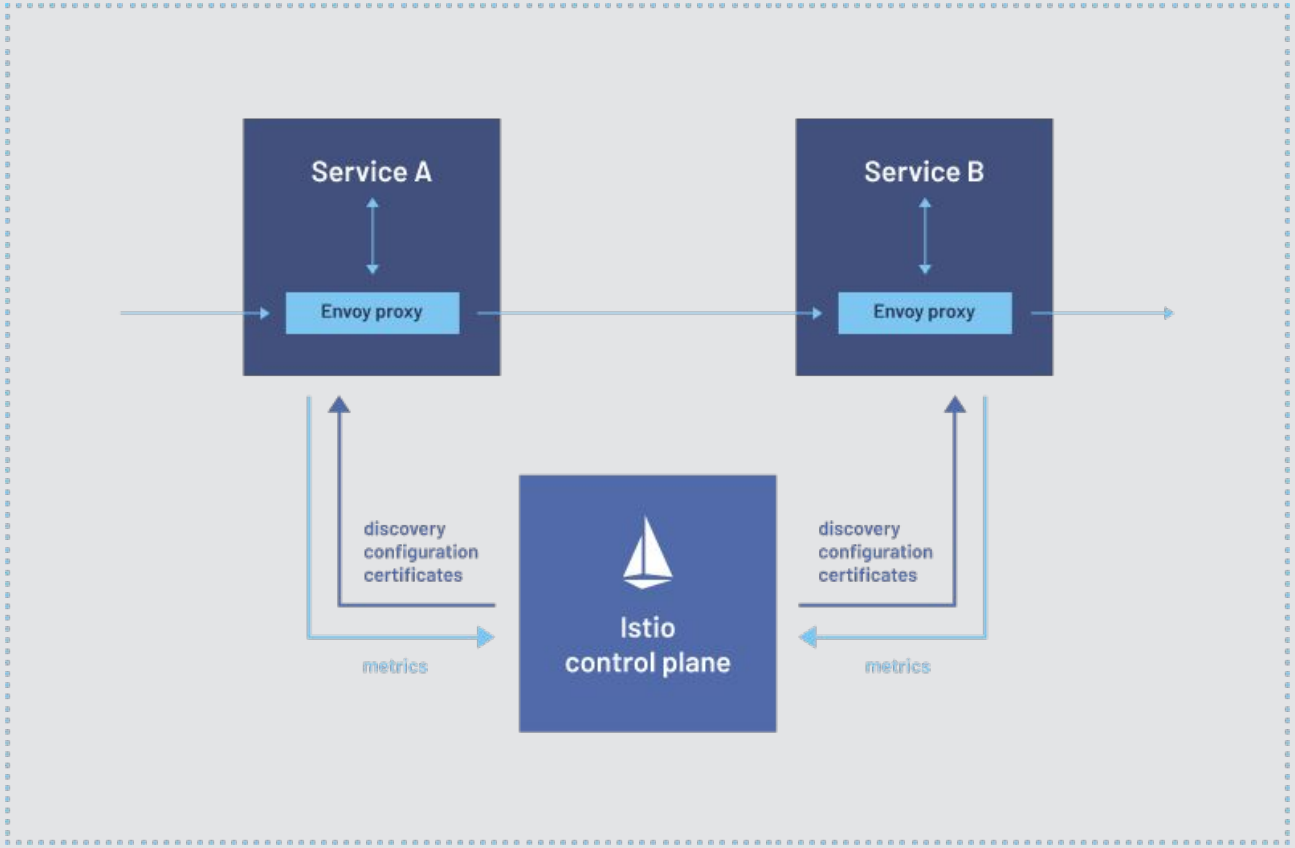
- VPC/K8s apply policies **per connection**
- Istio/Envoy apply policies **per request**



Pain Points

Running One Proxy Per Instance

- Every instance of every application gets a sidecar instance
- Pros: Envoy can control all traffic
- Cons: So. Many. Envoys.
  - Vertical Scaling becomes extremely expensive.



Managing the Magic

- Pods ≠ Deployment Spec
  - Injection modifies Pods, not Deployments
- Pods ≠ Deployment + Injection
  - Injection only occurs at Pod creation time
- Which version of Envoy?
  - ¯_(ツ)_/¯

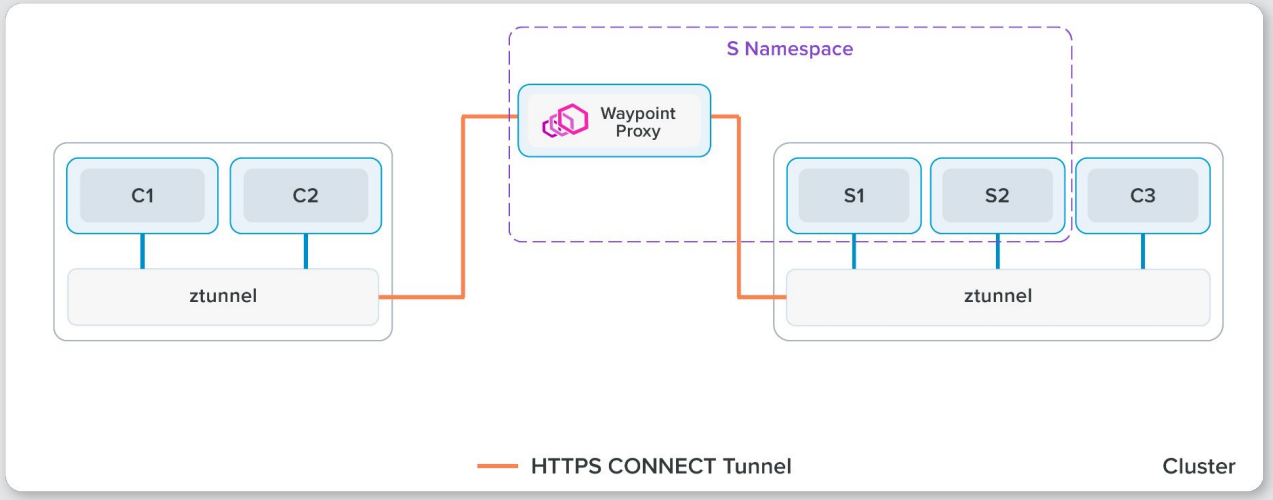
Sidecars can be automatically added to applicable Kubernetes pods using a mutating webhook admission controller provided by Istio.

When you set the `istio-injection=enabled` label on a namespace and the injection webhook is enabled, any new pods that are created in that namespace will automatically have a sidecar added to them.

-istio.io

Ambient Reduces User Pain

- One L4 Proxy per Node
- One scalable L7 Proxy per Service Account + Gateway
- All proxies are managed through Deployments/Daemonsets
- Pods = Deployment Spec
- For more info: istio.io/latest/blog/2022/introducing-ambient-mesh/



Conclusion

Defense at Every Layer

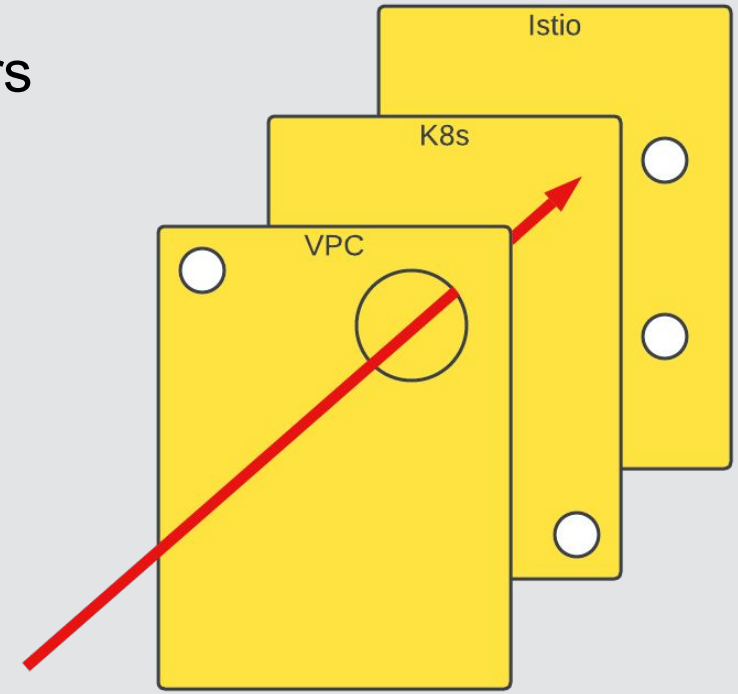
Strong Identity

	Identity	Policy	Observability
VPC Network ACLs	IP/Ports/VM	Network ACLs SecurityGroups	VPC Flow Logs
Aviatrix	Network Domains	Network Domains	Copilot Flow Logs
K8S	IP/Ports/Pods	NetworkPolicy	Node Telemetry
Istio	ServiceAccount Request Headers	AuthN/AuthZ AuthorizationPolicies RequestPolicies	Istio Telemetry

- **Self-service platforms are hard**
  - Safeguards to avoid users shooting themselves in the foot
  - Provide a Golden Path to avoid configuration errors

- **Defense in depth**: Add redundant security at all layers

- **Observability is key**
  - Help debug
  - Detect misconfiguration



Thank You!

Please scan the QR Code above to leave feedback on this session