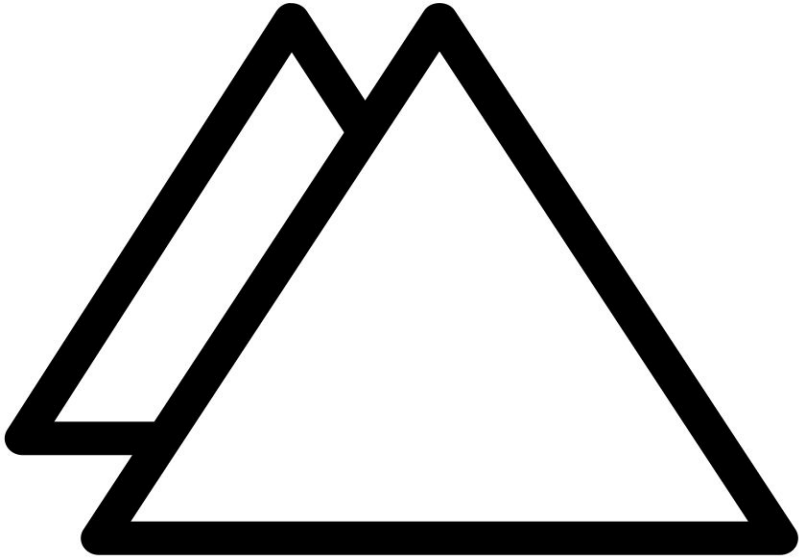
Service Mesh: Network Security?

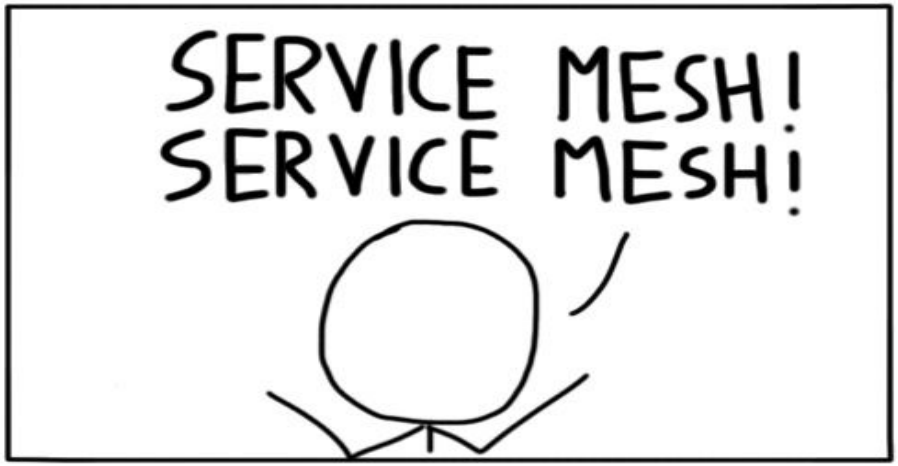
@piunnerup and @controlplaneio

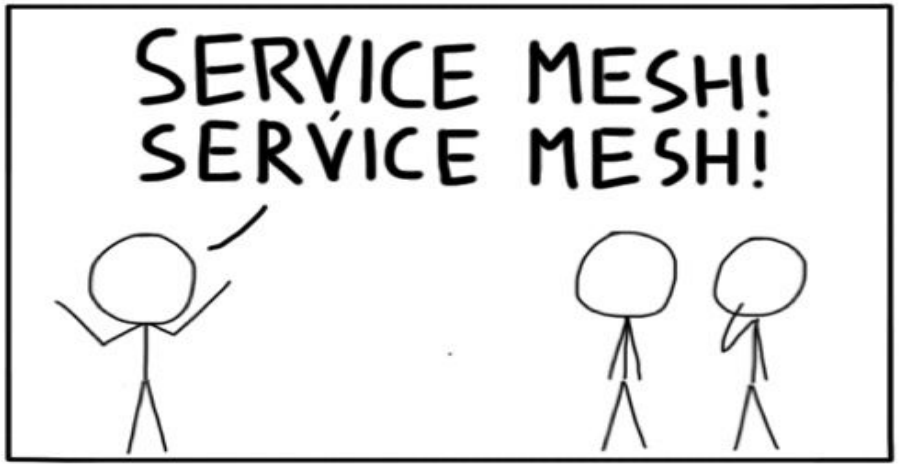
I’m:

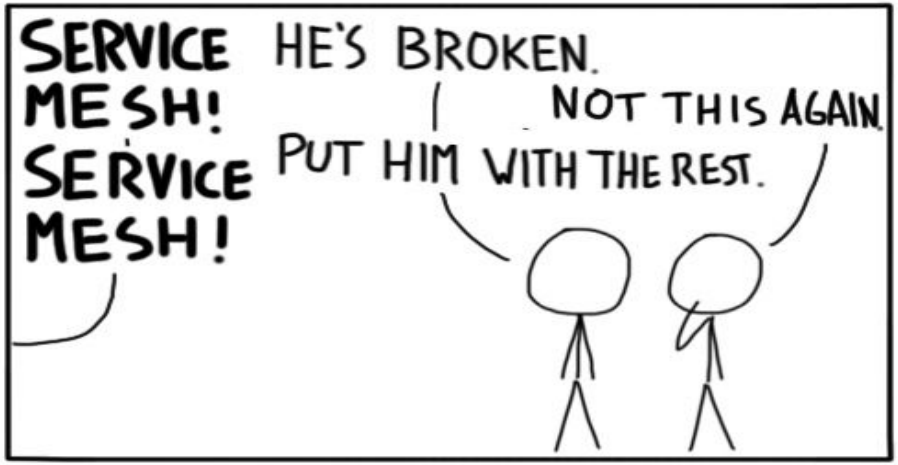
- Pi (yes, it’s my real name)
- ControlPlane
- Hardened Infrastructure

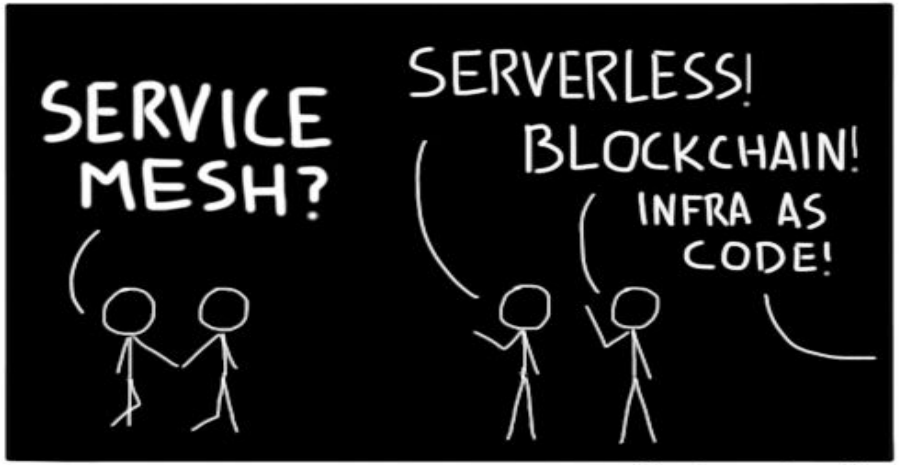


controlplane


SERVICE MESH!
SERVICE MESH!

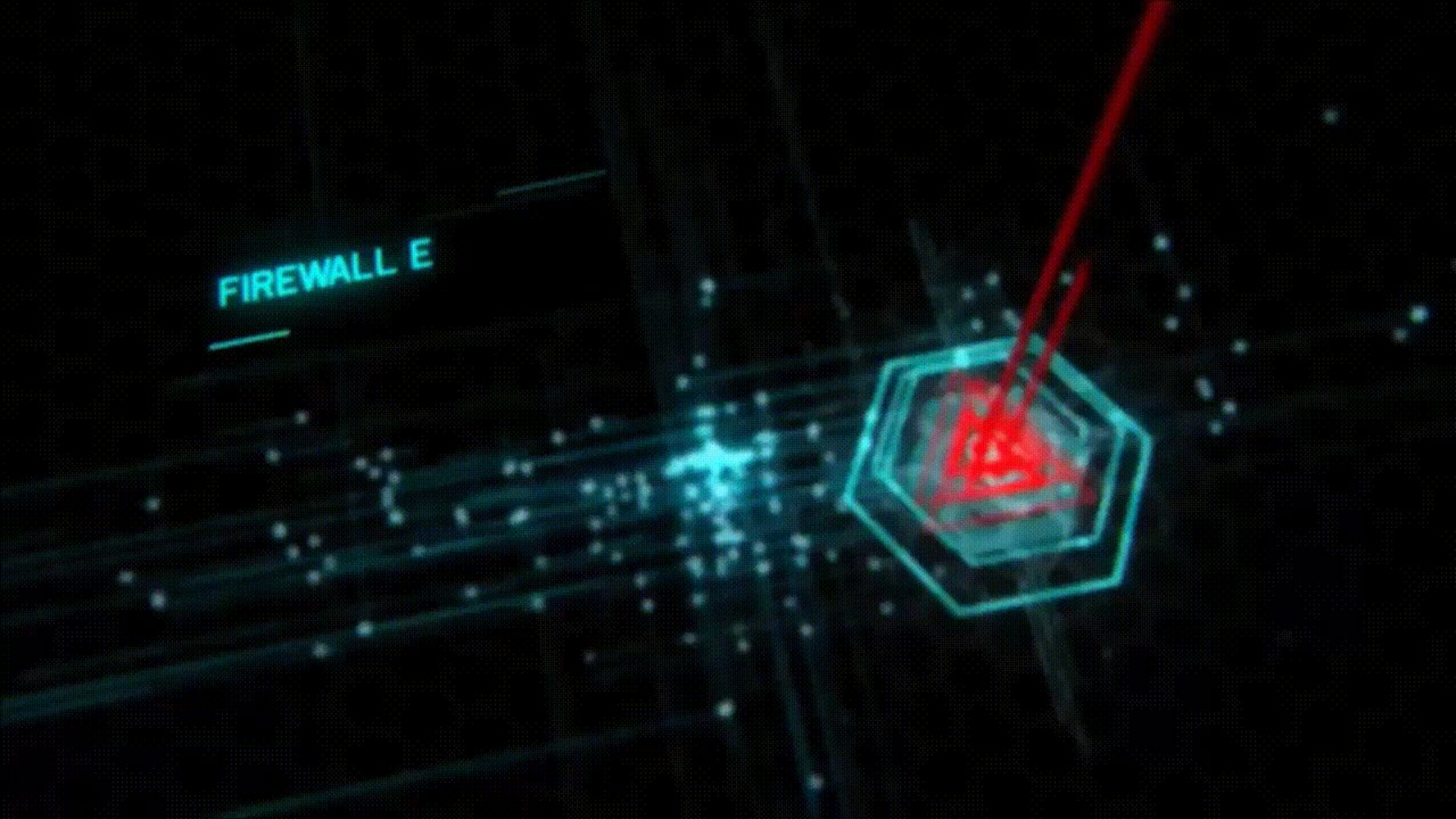

SERVICE MESH!
SERVICE MESH!


SERVICE MESH!
SERVICE MESH!
HE'S BROKEN.
PUT HIM WITH THE REST.
NOT THIS AGAIN.


SERVICE MESH?
SERVERLESS!
BLOCKCHAIN!
INFRA AS CODE!

@sebiwicb

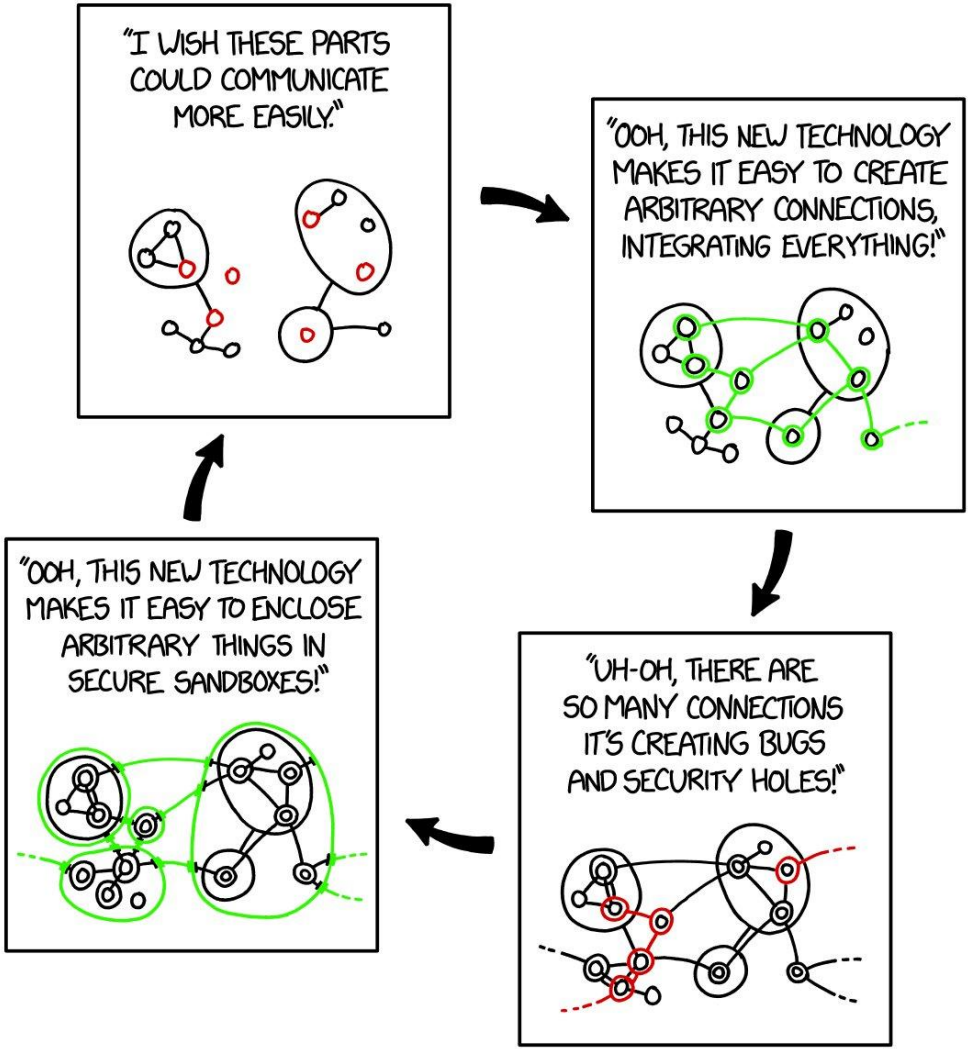

“Why everybody trying to break internet?”


FIREWALL E

Sandboxing Circle of Life

All I want is a secure system where it's easy to do anything I want. Is that so much to ask?

https://www.xkcd.com/2044/



controlplane

Istio: Fix Microservice Complexity with...Abstractions

controlplane

Istio is...Complex

controlplane

https://kubedex.com/istio-vs-linkerd-vs-linkerd2-vs-consul	Istio	Linkerd	Linkerd2	Consul Connect
Model	Sidecar	Node Agent	Sidecar	Sidecar
Platform	Kubernetes	Any	Kubernetes	Any
language	Go	JVM	Go / Rust	Go
Protocol	HTTP1.1 / HTTP2 / gRPC / TCP	HTTP1.1 / HTTP2 / gRPC	HTTP1.1 / HTTP2 / gRPC / TCP	TCP
Default Data Plane	Envoy (supports others)	Native	Native	Native (or Envoy)
Sidecar Injection	Yes	No	No	No
Encryption	Yes	Yes	Experimental	Yes
Traffic Control	label/content based routing, traffic shifting	Dynamic request routing, traffic shifting, per request routing	?	static upstream, prepared query, http api / dns with native integration
Resilience	timeouts, retries, connection pools, outlier detection	timeouts, retries, deadlines, circuit breaking	?	Pluggable
Prometheus Integration	Yes	Yes	Yes	No
Tracing Integration	Jaeger	Zipkin	None	Pluggable
Host to Host auth	Service Accounts	TLS Mutual Auth	No	Consul ACL
Agent Caching	Yes	No	No	Yes
Secure connection outside cluster	No	Yes	No	Yes
Complexity	High	High	Low	Low
Paid Support	No	Yes	Yes	Yes

What this talk is about

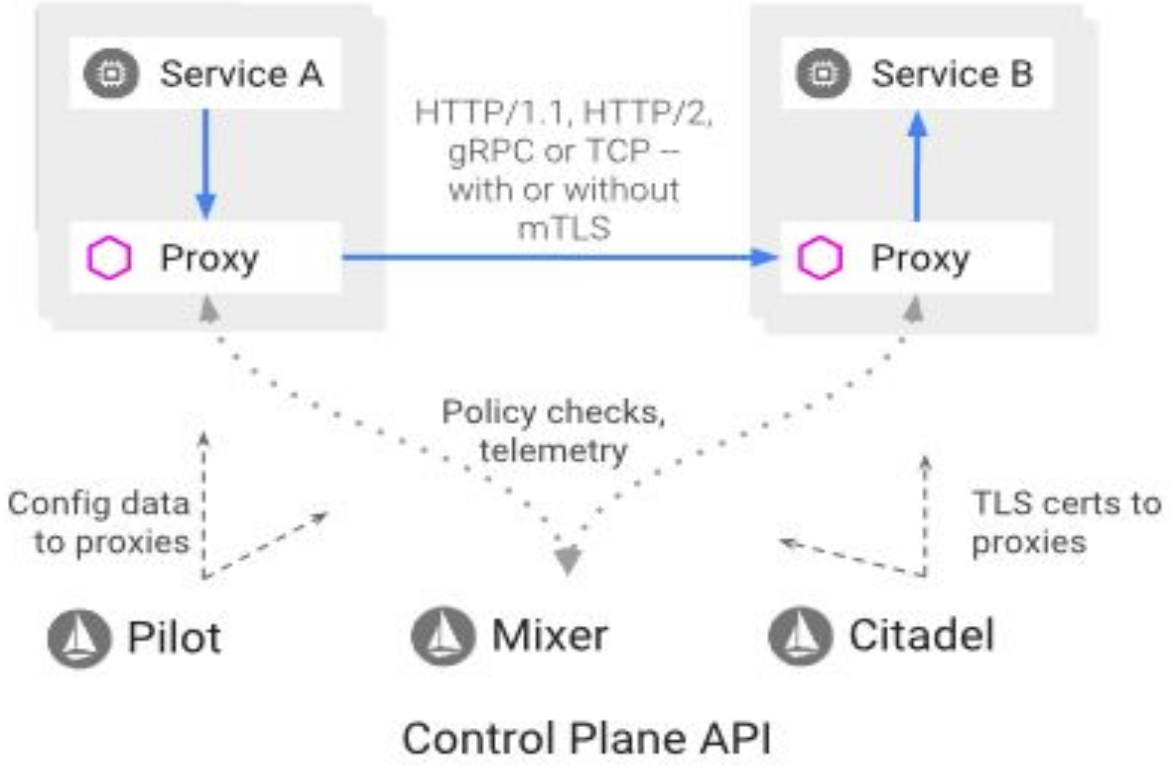
- Ye Olde Way
- The Istio control plane
- SPIFFE, TLS, and Istio network security
- Authentication and authorisation
- Traffic management, and policy

A long time ago in a galaxy far,
far, away...

Ye Olde Way

- Netflix distributed system tooling (Eureka, Hystrix, Zuul)
- Redis rate limiting
- Internal PKI
- DNS or route-level service releases

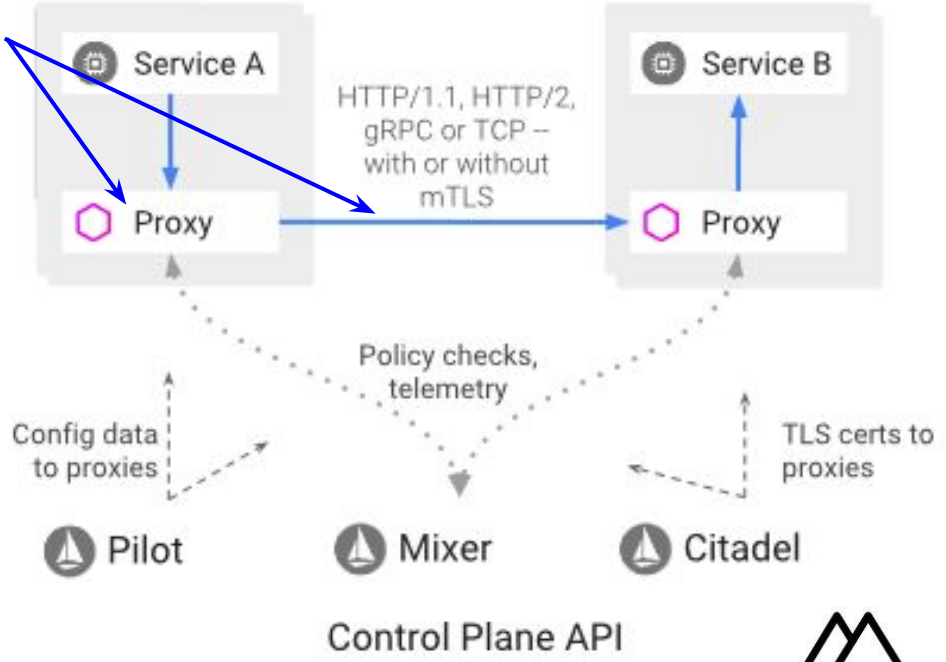
A New Hope - Service Mesh



controlplane

A New Hope - Service Mesh

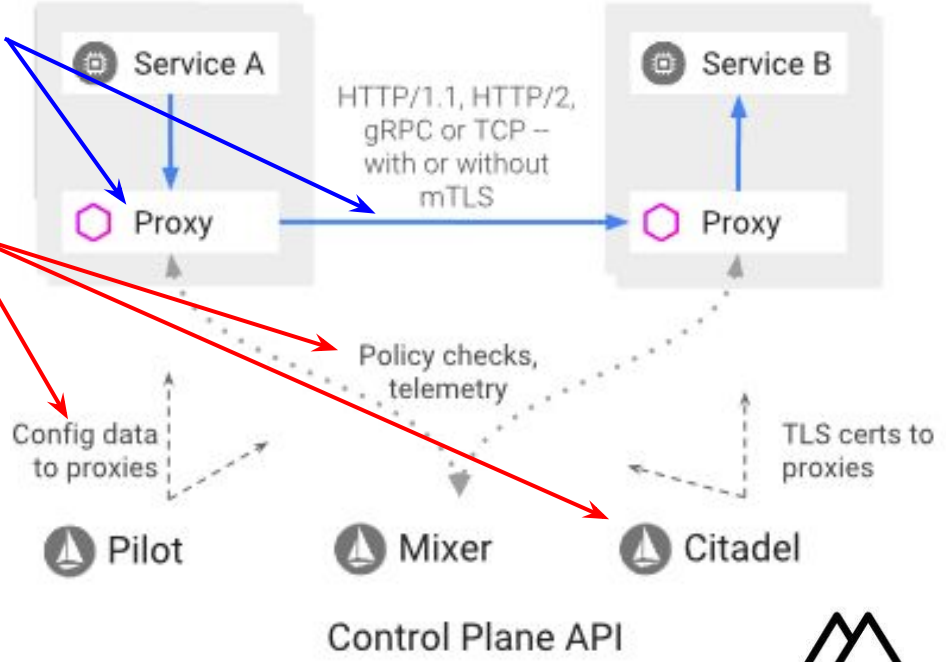
- Resilient, secure **Data Plane**



controlplane

A New Hope - Service Mesh

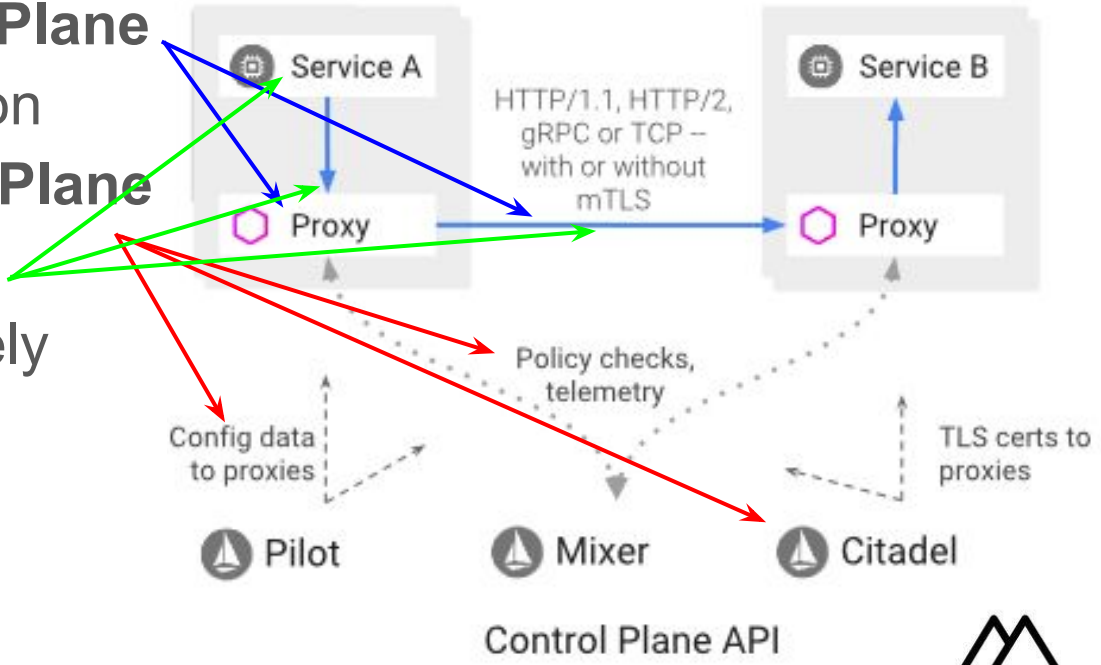
- Resilient, secure **Data Plane**
- Data Plane configuration automated by **Control Plane**



controlplane

A New Hope - Service Mesh

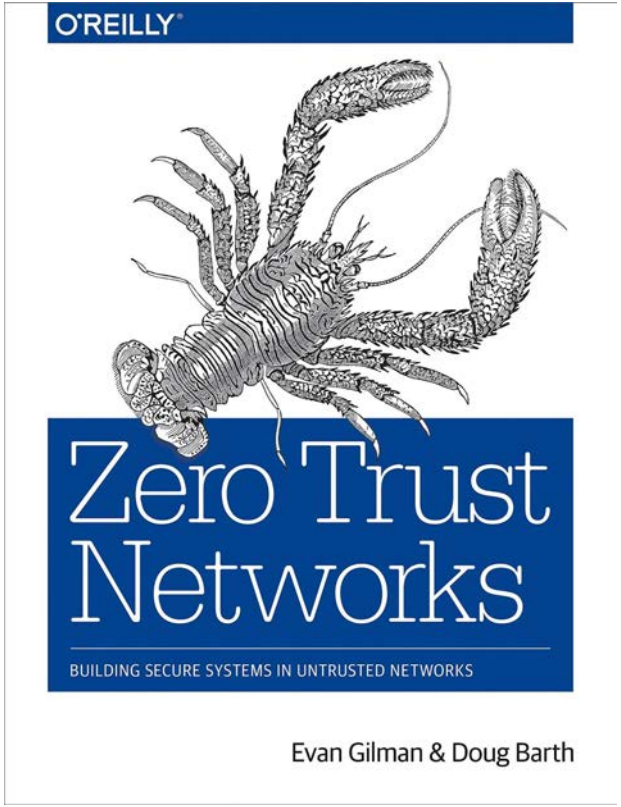
- Resilient, secure **Data Plane**
- Data Plane configuration automated by **Control Plane**
- **Application/Services** communicate exclusively through the mesh



controlplane

A New Hope - Service Mesh

- Resilient, secure **Data Plane**
- Data Plane configuration automated by **Control Plane**
- **Application/Services** communicate exclusively through the mesh
- **Zero Trust Networking** secures the mesh



controlplane

Service Mesh: Tables Stakes

- Resiliency features (retries, timeouts, deadlines, etc)
- Cascading failure prevention (circuit breaking)
- Robust load balancing algorithms
- Control over request routing (useful for things like CI/CD release patterns)
- The ability to introduce and manage TLS termination between communication endpoints
- Rich sets of metrics to provide instrumentation at the service-to-service layer

https://thenewstack.io/which-service-mesh-should-i-use/

Istio

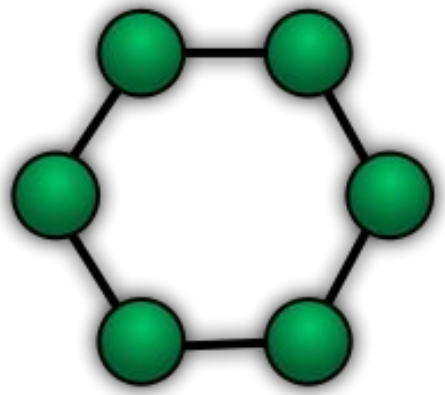
- Automatic mutual TLS between services
- Service-level RBAC
- External identity provider integration
- Policy and quota enforcement, dynamic per-request routing
- Deployment strategies such as red/black, canary, dark/mirrored
- Distributed tracing
- Network policy between apps/services, and on ingress/egress
- Zero-ish code changes

Istio

- **Automatic mutual TLS** between services
- **Service-level RBAC**
- **External identity provider** integration
- **Policy and quota enforcement**, dynamic **per-request routing**
- Deployment strategies such as red/black, canary, dark/mirrored
- Distributed tracing
- **Network policy** between apps/services, and on ingress/egress
- **Zero-ish code changes**

What is a Service Mesh?

Some Network Topologies

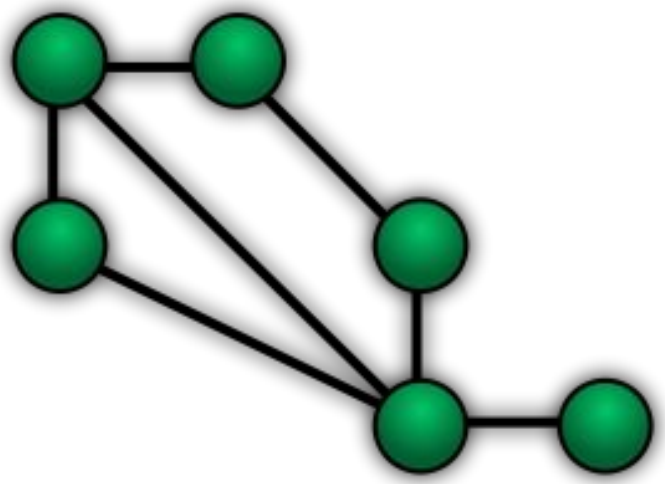


Ring

Star

controlplane

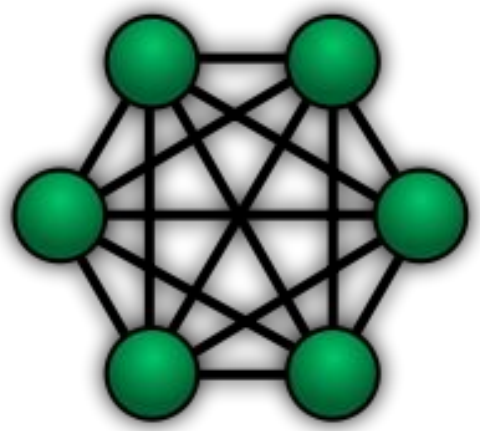
What is a Service Mesh?



Mesh

controlplane

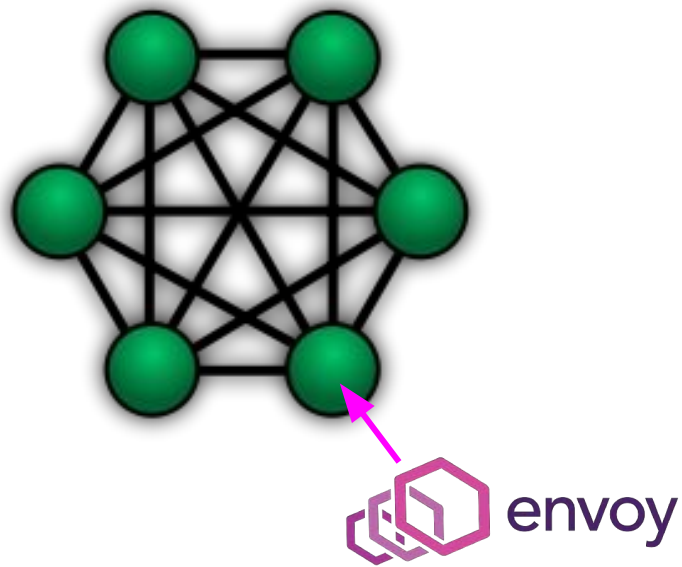
What is a Service Mesh?



Fully Connected

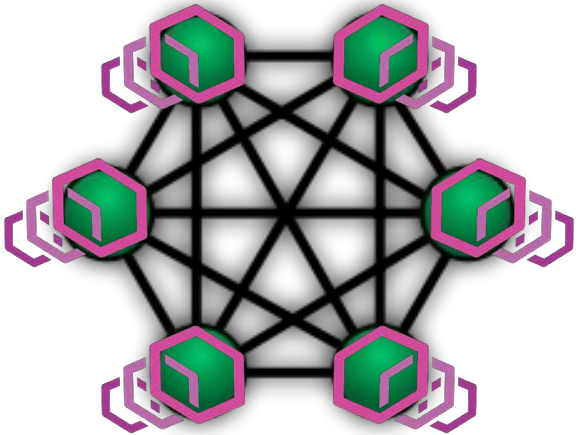
controlplane

What is a Service Mesh?



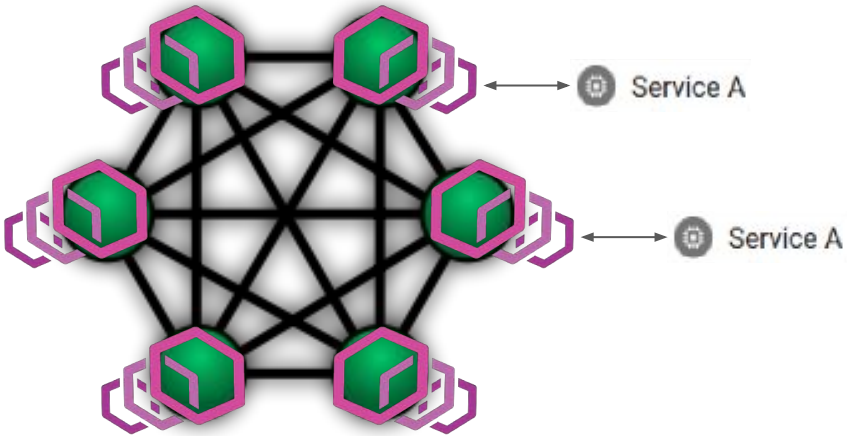
controlplane

What is a Service Mesh?



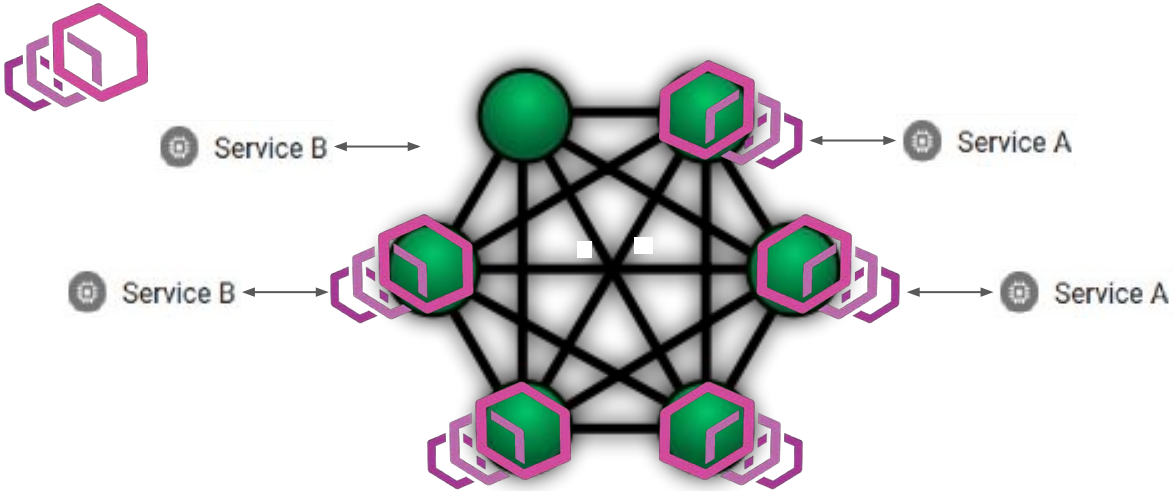
controlplane

What is a Service Mesh?



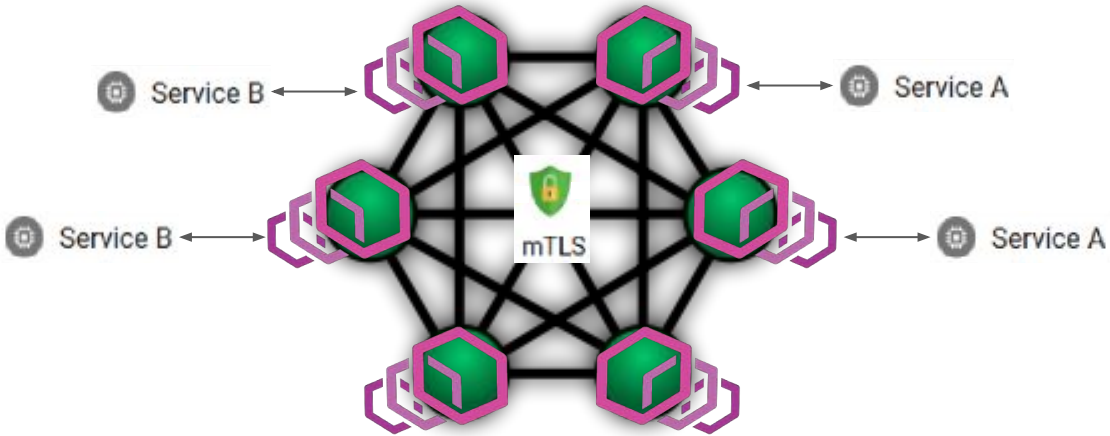
controlplane

What is a Service Mesh?



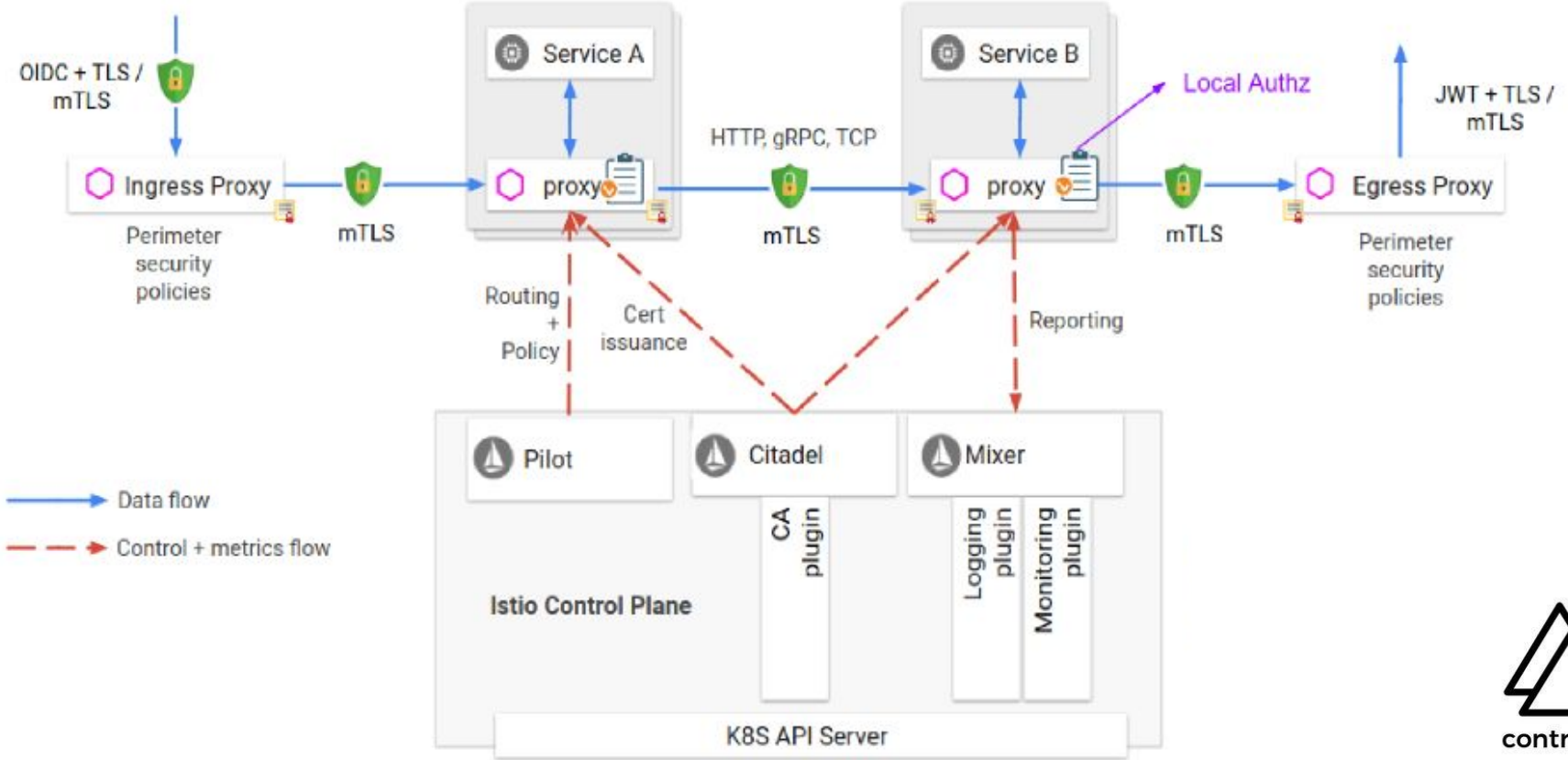
controlplane

What is a Service Mesh?



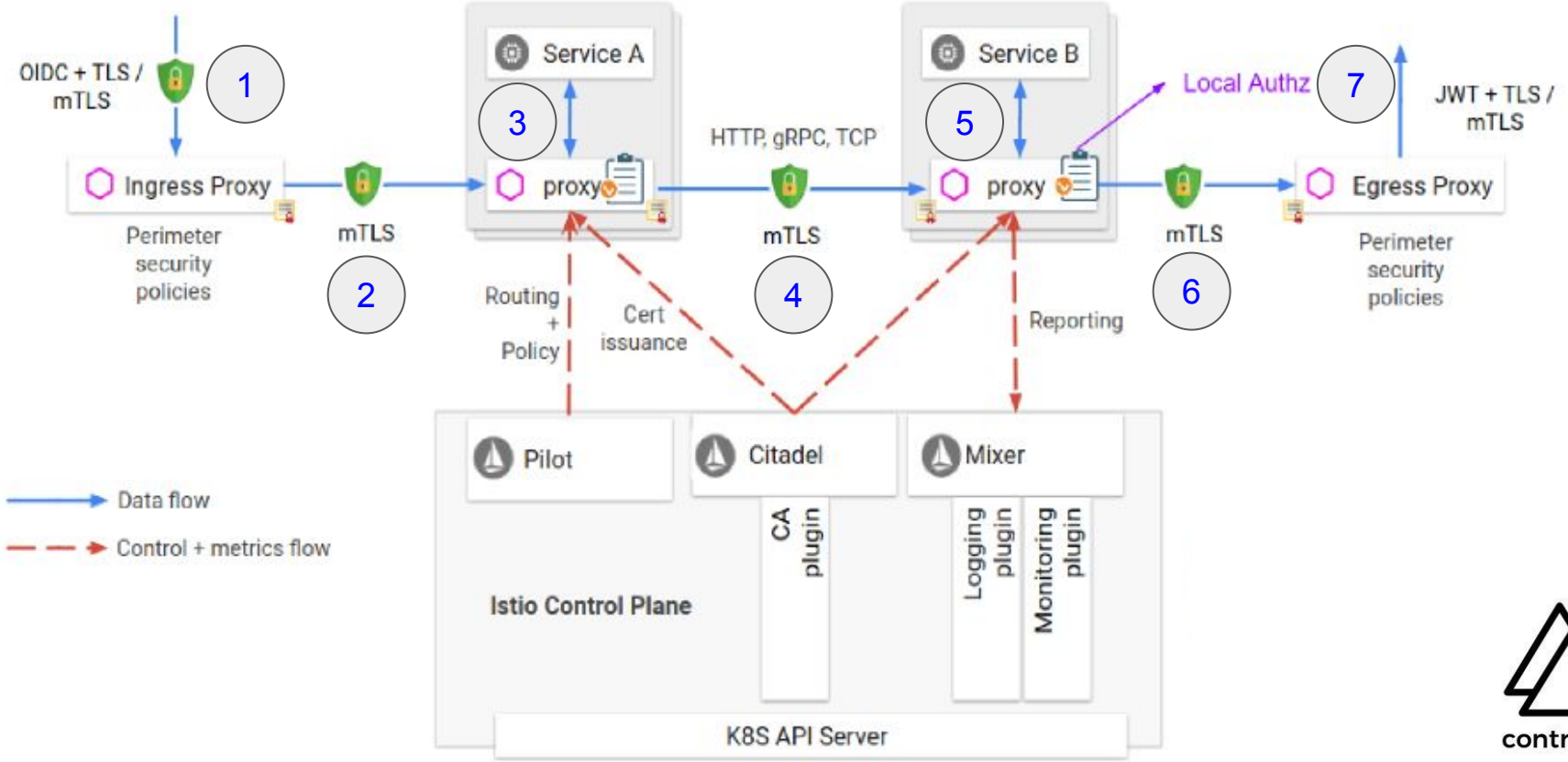
controlplane

Istio Architecture



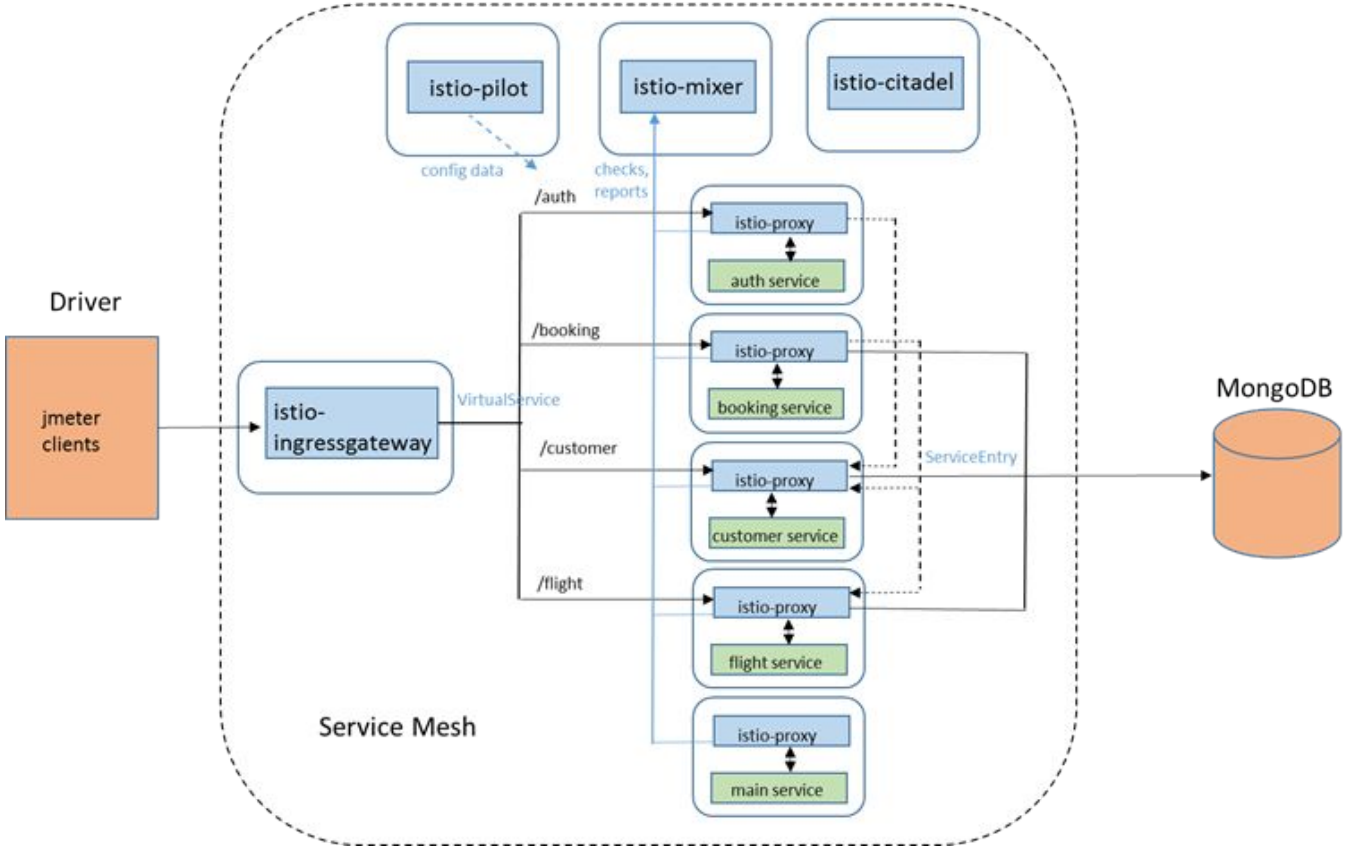
controlplane

Istio Architecture



controlplane

Istio Architecture



controlplane

Istio Security


RISE OF THE
HACKERS
Source: Carna Botnet

Problem: Strong Service Security at Scale

- **Concerns**
  - Insiders
  - Hijacked services
  - Microservice attack surface
  - Workload mobility
  - Brittle fine-grained models
  - Securing resources not just endpoints
  - Audit & Compliance
- **Wants**
  - Workload mobility
  - Remote admin & development
  - Shared & 3rd party services
  - User & Service identity
  - Lower costs

https://www.infoq.com/presentations/istio-service-mesh

What Istio Installs

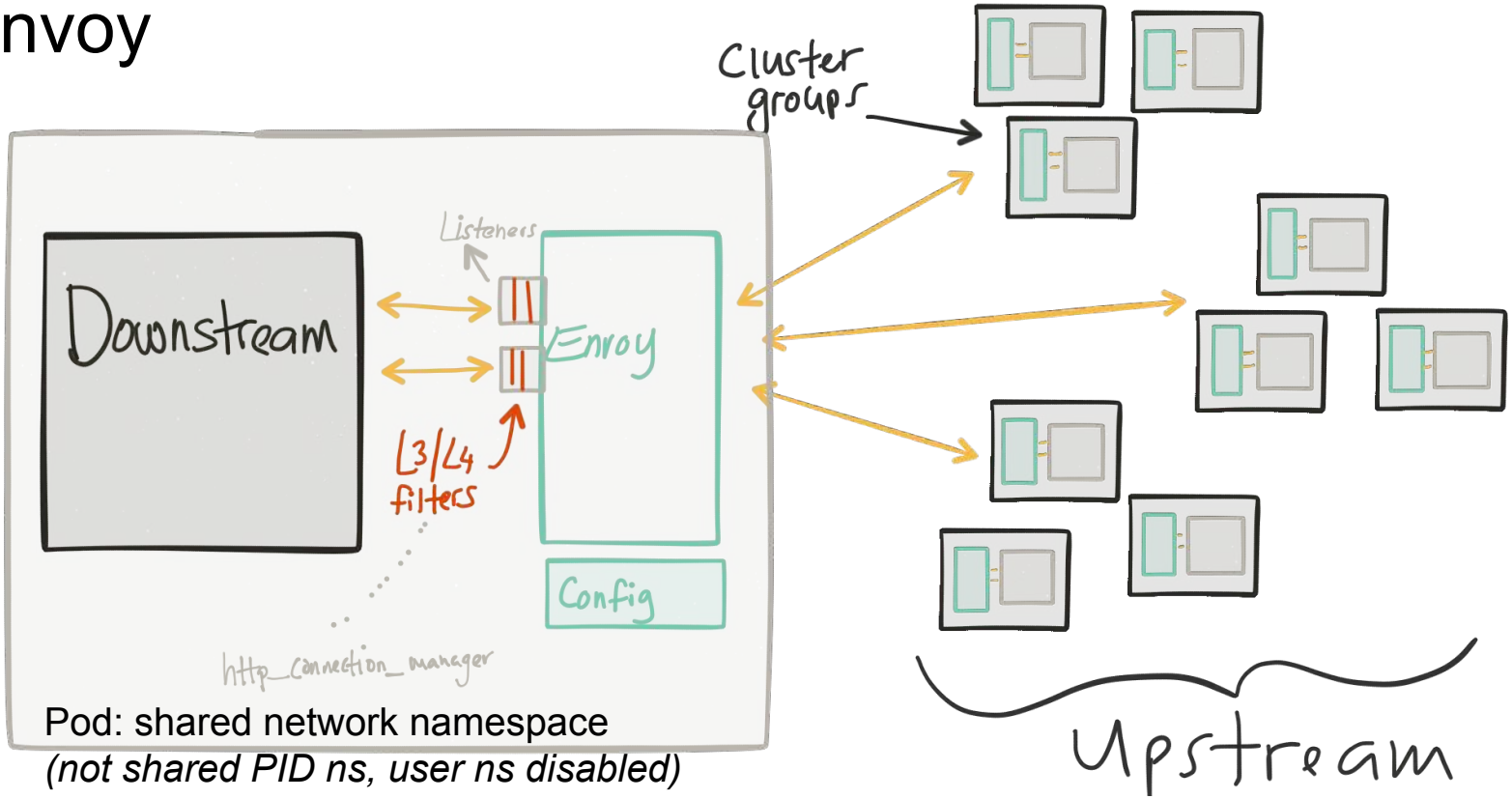
- gcr.io/istio-release/**pilot**:1.0.2
- gcr.io/istio-release/**mixer**:1.0.2
- gcr.io/istio-release/**citadel**:1.0.2
- gcr.io/istio-release/**galley**:1.0.2
- gcr.io/istio-release/**sidecar_injector**:1.0.2
- gcr.io/istio-release/**proxyv2**:1.0.2
- gcr.io/istio-release/**proxy_init**:1.0.2
- gcr.io/istio-release/**proxy_debug**:1.0.2
- gcr.io/istio-release/**grafana**:1.0.2
- docker.io/**jaegertracing**/all-in-one:1.5
- docker.io/prom/**prometheus**:v2.3.1
- docker.io/prom/**statsd-exporter**:v0.6.0

What Istio Installs

- gcr.io/istio-release/**pilot**:1.0.2
  - Configuration writer for envoy's API
- gcr.io/istio-release/**mixer**:1.0.2
  - Central metadata, metrics, and policy
- gcr.io/istio-release/**citadel**:1.0.2
  - The certificate authority, to issue and rotate certificates
- gcr.io/istio-release/**galley**:1.0.2
  - User-supplied config validation for the rest of the control plane
- gcr.io/istio-release/**sidecar_injector**:1.0.2
  - Kubernetes webhook for automatic Istio sidecar injection
- gcr.io/istio-release/**proxyv2**:1.0.2
  - Envoy
- gcr.io/istio-release/**proxy_debug**:1.0.2
  - Envoy plus debug symbols, sudo (to allow tcpdump)
- gcr.io/istio-release/**proxy_init**:1.0.2
  - Init container that configures iptables for sidecar networking
- gcr.io/istio-release/**grafana**:1.0.2
  - Dashboards add-on
- docker.io/**jaegertracing**/all-in-one:1.5
  - Tracing add-on
- docker.io/prom/**prometheus**:v2.3.1
  - Metrics add-on
- docker.io/prom/**statsd-exporter**:v0.6.0
  - statsd to prometheus bridge

Envoy

Envoy



Pod: shared network namespace
(not shared PID ns, user ns disabled)

http://blog.christianposta.com/istio-workshop/

controlplane

Envoy - Automatic Sidecar Injection

```yaml
apiVersion: v1
kind: Namespace
metadata:
 labels:
   istio-injection: enabled
 name: my-ns
```

Security: RBAC (aka Authorisation)

```
apiVersion: v1
items:
- apiVersion: admissionregistration.k8s.io/v1beta1
  kind: MutatingWebhookConfiguration
    name: istio-sidecar-injector
  webhooks:
  - clientConfig:
      caBundle: LS0tLS1CRUdJT...
      service:
         name: istio-sidecar-injector
         namespace: istio-system
         path: /inject
    failurePolicy: Fail
    name: sidecar-injector.istio.io
    namespaceSelector:
      matchLabels:
        istio-injection: enabled
    rules:
    - apiGroups:
      - ""
      apiVersions:
      - v1
      operations:
      - CREATE
      resources:
      - pods
```

controlplane

Envoy - Injection of Proxy Init and Proxy (Envoy)

```
# apiVersion: v1, kind: Pod
metadata:
  annotations:
    sidecar.istio.io/status:
'{"version":"42...129","initContainers":["istio-init"],"containers":["istio-proxy"],"vol
umes":["istio-envoy","istio-certs"],"imagePullSecrets":null}'
  spec:
    containers:
    - image: istio/examples-bookinfo-details-v1:1.8.0
# ...
    - image: gcr.io/istio-release/proxyv2:1.0.2
      name: istio-proxy
# ...
    - image: gcr.io/istio-release/proxy_init:1.0.2
      name: istio-init
```

controlplane

Envoy's initialisation container: `proxy_init`

Jérôme Petazzoni
@jpetazzo

Following

OH: "In any team you need a tank, a healer, a damage dealer, someone with crowd control abilities, and another who knows iptables"

6:41 PM - 27 Jun 2015 from Kansas City, MO

1,228 Retweets **1,589** Likes

29 1.2K 1.6K

Envoy - Proxy Init (Envoy)

https://github.com/istio/istio/blob/master/tools/deb/istio-iptables.sh

istio-iptables.sh

```
#!/bin/bash
#
# Copyright 2017, 2018 Istio Authors. All Rights Reserved.
#
# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
#
#    http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.
#
################################################################################
#
# Initialization script responsible for setting up port forwarding for Istio sidecar.
```

istio-iptables.sh

```
function usage() {
 echo "${0} -p PORT -u UID -g GID [-m mode] [-b ports] [-d ports] [-i CIDR] [-x CIDR] [-h]"
 echo ''
 echo '  -p: Specify the envoy port to which redirect all TCP traffic (default $ENVOY_PORT = 15001)'
 echo '  -u: Specify the UID of the user for which the redirection is not'
 echo '      applied. Typically, this is the UID of the proxy container'
 echo '      (default to uid of $ENVOY_USER, uid of istio_proxy, or 1337)'
 echo '  -g: Specify the GID of the user for which the redirection is not'
 echo '      applied. (same default value as -u param)'
 echo '  -m: The mode used to redirect inbound connections to Envoy, either "REDIRECT" or "TPROXY"'
 echo '      (default to $ISTIO_INBOUND_INTERCEPTION_MODE)'
 echo '  -b: Comma separated list of inbound ports for which traffic is to be redirected to Envoy (optional). The'
 echo '      wildcard character "*" can be used to configure redirection for all ports. An empty list will disable'
 echo '      all inbound redirection (default to $ISTIO_INBOUND_PORTS)'
 echo '  -d: Comma separated list of inbound ports to be excluded from redirection to Envoy (optional). Only applies'
 echo '      when all inbound traffic (i.e. "*") is being redirected (default to $ISTIO_LOCAL_EXCLUDE_PORTS)'
 echo '  -i: Comma separated list of IP ranges in CIDR form to redirect to envoy (optional). The wildcard'
 echo '      character "*" can be used to redirect all outbound traffic. An empty list will disable all outbound'
 echo '      redirection (default to $ISTIO_SERVICE_CIDR)'
 echo '  -x: Comma separated list of IP ranges in CIDR form to be excluded from redirection. Only applies when all '
 echo '      outbound traffic (i.e. "*") is being redirected (default to $ISTIO_SERVICE_EXCLUDE_CIDR).'
 echo ''
 echo 'Using environment variables in $ISTIO_SIDECAR_CONFIG (default: /var/lib/istio/envoy/sidecar.env)'
}
```

istio-iptables.sh

```
PROXY_PORT=${ENVOY_PORT:-15001}
PROXY_UID= #  defaults to ${ENVOY_USER} or “istio-proxy" or UID 1337
PROXY_GID=
INBOUND_INTERCEPTION_MODE=${ISTIO_INBOUND_INTERCEPTION_MODE}
INBOUND_TPROXY_MARK=${ISTIO_INBOUND_TPROXY_MARK:-1337}
INBOUND_TPROXY_ROUTE_TABLE=${ISTIO_INBOUND_TPROXY_ROUTE_TABLE:-133}
INBOUND_PORTS_INCLUDE=${ISTIO_INBOUND_PORTS-}
INBOUND_PORTS_EXCLUDE=${ISTIO_LOCAL_EXCLUDE_PORTS-}
OUTBOUND_IP_RANGES_INCLUDE=${ISTIO_SERVICE_CIDR-}
OUTBOUND_IP_RANGES_EXCLUDE=${ISTIO_SERVICE_EXCLUDE_CIDR-}
```

istio-iptables.sh

```
# Remove the old chains, to generate new configs.
iptables -t nat -D PREROUTING -p tcp -j ISTIO_INBOUND 2>/dev/null
iptables -t mangle -D PREROUTING -p tcp -j ISTIO_INBOUND 2>/dev/null
iptables -t nat -D OUTPUT -p tcp -j ISTIO_OUTPUT 2>/dev/null

# Flush and delete the istio chains.
iptables -t nat -F ISTIO_OUTPUT 2>/dev/null
iptables -t nat -X ISTIO_OUTPUT 2>/dev/null
iptables -t nat -F ISTIO_INBOUND 2>/dev/null
iptables -t nat -X ISTIO_INBOUND 2>/dev/null
iptables -t mangle -F ISTIO_INBOUND 2>/dev/null
iptables -t mangle -X ISTIO_INBOUND 2>/dev/null
iptables -t mangle -F ISTIO_DIVERT 2>/dev/null
iptables -t mangle -X ISTIO_DIVERT 2>/dev/null
iptables -t mangle -F ISTIO_TPROXY 2>/dev/null
iptables -t mangle -X ISTIO_TPROXY 2>/dev/null

# Must be last, the others refer to it
iptables -t nat -F ISTIO_REDIRECT 2>/dev/null
iptables -t nat -X ISTIO_REDIRECT 2>/dev/null
iptables -t nat -F ISTIO_IN_REDIRECT 2>/dev/null
iptables -t nat -X ISTIO_IN_REDIRECT 2>/dev/null

if [ "${1:-}" = "clean" ]; then
 echo "Only cleaning, no new rules added"
 exit 0
fi
```

istio-iptables.sh

```
if [ "${INBOUND_PORTS_INCLUDE}" == "*" ]; then
  # Makes sure SSH is not redirected
  iptables -t ${table} -A ISTIO_INBOUND -p tcp --dport 22 -j RETURN
  # Apply any user-specified port exclusions.
  if [ -n "${INBOUND_PORTS_EXCLUDE}" ]; then
    for port in ${INBOUND_PORTS_EXCLUDE}; do
      iptables -t ${table} -A ISTIO_INBOUND -p tcp --dport "${port}" -j RETURN
    done
  fi
  # Redirect remaining inbound traffic to Envoy.
  if [ "${INBOUND_INTERCEPTION_MODE}" = "TPROXY" ]; then
    # If an inbound packet belongs to an established socket, route it to the
    # loopback interface.
    iptables -t mangle -A ISTIO_INBOUND -p tcp -m socket -j ISTIO_DIVERT || echo "No socket match support"
    # Otherwise, it's a new connection. Redirect it using TPROXY.
    iptables -t mangle -A ISTIO_INBOUND -p tcp -j ISTIO_TPROXY
  else
    iptables -t nat -A ISTIO_INBOUND -p tcp -j ISTIO_IN_REDIRECT
  fi
else
  # ...
```

istio-iptables.sh

```
# Skip redirection for Envoy-aware applications and
# container-to-container traffic both of which explicitly use
# localhost.
iptables -t nat -A ISTIO_OUTPUT -d 127.0.0.1/32 -j RETURN

# Apply outbound IP exclusions. Must be applied before inclusions.
if [ -n "${OUTBOUND_IP_RANGES_EXCLUDE}" ]; then
 for cidr in ${OUTBOUND_IP_RANGES_EXCLUDE}; do
   iptables -t nat -A ISTIO_OUTPUT -d "${cidr}" -j RETURN
 done
fi

# Apply outbound IP inclusions.
if [ "${OUTBOUND_IP_RANGES_INCLUDE}" == "*" ]; then
 # Wildcard specified. Redirect all remaining outbound traffic to Envoy.
 iptables -t nat -A ISTIO_OUTPUT -j ISTIO_REDIRECT
elif [ -n "${OUTBOUND_IP_RANGES_INCLUDE}" ]; then
 # User has specified a non-empty list of cidrs to be redirected to Envoy.
 for cidr in ${OUTBOUND_IP_RANGES_INCLUDE}; do
   iptables -t nat -A ISTIO_OUTPUT -d "${cidr}" -j ISTIO_REDIRECT
 done
 # All other traffic is not redirected.
 iptables -t nat -A ISTIO_OUTPUT -j RETURN
fi
```

istio-iptables.sh

```
# If ENABLE_INBOUND_IPV6 is unset (default unset), restrict IPv6 traffic.
set +o nounset
if [ -z "${ENABLE_INBOUND_IPV6}" ]; then
 # Drop all inbound traffic except established connections.
 # TODO: support receiving IPv6 traffic in the same way as IPv4.
 ip6tables -F INPUT || true
 ip6tables -A INPUT -m state --state ESTABLISHED -j ACCEPT || true
 ip6tables -A INPUT -j REJECT || true
fi
```

Envoy and The Mesh

Envoy - Mesh Proxy Status Report

```
$ istioctl proxy-status

PROXY                                                  CDS        LDS        EDS               RDS          PILOT                           VERSION
details-v1-586974b75-6l4qq.default                     SYNCED     SYNCED     SYNCED (100%)     SYNCED       istio-pilot-76b56cd46-pr9zg     1.0.2
istio-egressgateway-667fddb57-qr252.istio-system       SYNCED     SYNCED     SYNCED (100%)     NOT SENT     istio-pilot-76b56cd46-pr9zg     1.0.2
istio-egressgateway-667fddb57-trnzv.istio-system       SYNCED     SYNCED     SYNCED (100%)     NOT SENT     istio-pilot-76b56cd46-pr9zg     1.0.2
istio-ingressgateway-7998f6b7b8-28sz6.istio-system     SYNCED     SYNCED     SYNCED (100%)     SYNCED       istio-pilot-76b56cd46-pr9zg     1.0.2
istio-ingressgateway-7998f6b7b8-7l5jr.istio-system     SYNCED     SYNCED     SYNCED (100%)     SYNCED       istio-pilot-76b56cd46-pr9zg     1.0.2
istio-ingressgateway-7998f6b7b8-j7lw7.istio-system     SYNCED     SYNCED     SYNCED (100%)     SYNCED       istio-pilot-76b56cd46-pr9zg     1.0.2
istio-ingressgateway-7998f6b7b8-vxgsl.istio-system     SYNCED     SYNCED     SYNCED (100%)     SYNCED       istio-pilot-76b56cd46-pr9zg     1.0.2
istio-ingressgateway-7998f6b7b8-x9829.istio-system     SYNCED     SYNCED     SYNCED (100%)     SYNCED       istio-pilot-76b56cd46-pr9zg     1.0.2
productpage-v1-5bdbcbd659-gsgtd.default                SYNCED     SYNCED     SYNCED (100%)     SYNCED       istio-pilot-76b56cd46-pr9zg     1.0.2
ratings-v1-588c545875-w6v64.default                    SYNCED     SYNCED     SYNCED (100%)     SYNCED       istio-pilot-76b56cd46-pr9zg     1.0.2
reviews-v1-6b5b46bb47-jv6f4.default                    SYNCED     SYNCED     SYNCED (100%)     SYNCED       istio-pilot-76b56cd46-pr9zg     1.0.2
reviews-v2-6c565864fc-ppfns.default                    SYNCED     SYNCED     SYNCED (100%)     SYNCED       istio-pilot-76b56cd46-pr9zg     1.0.2
reviews-v3-65849d49f5-wgld7.default                    SYNCED     SYNCED     SYNCED (100%)     SYNCED       istio-pilot-76b56cd46-pr9zg     1.0.2
```

Envoy - Proxy Configuration

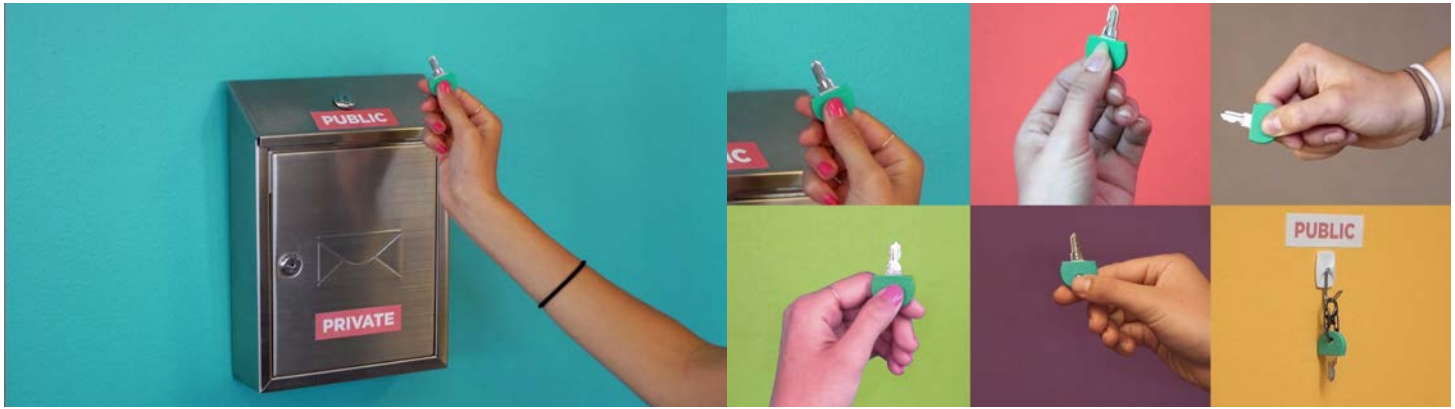
```
$ istioctl proxy-config cluster productpage-v1-55d65d9c4-f52c5

SERVICE FQDN                                                     PORT      SUBSET     DIRECTION     TYPE
BlackHoleCluster                                                 -         -          -             STATIC
custom-metrics-stackdriver-adapter.custom-metrics.svc.cluster.local     443       -          outbound      EDS
default-http-backend.kube-system.svc.cluster.local               80        -          outbound      EDS
details.default.svc.cluster.local                                9080      -          outbound      EDS
grafana.istio-system.svc.cluster.local                           3000      -          outbound      EDS
heapster.kube-system.svc.cluster.local                           80        -          outbound      EDS
istio-citadel.istio-system.svc.cluster.local                     8060      -          outbound      EDS
istio-citadel.istio-system.svc.cluster.local                     9093      -          outbound      EDS
istio-egressgateway.istio-system.svc.cluster.local               80        -          outbound      EDS
istio-egressgateway.istio-system.svc.cluster.local               443       -          outbound      EDS
istio-galley.istio-system.svc.cluster.local                      443       -          outbound      EDS
istio-galley.istio-system.svc.cluster.local                      9093      -          outbound      EDS
istio-ingressgateway.istio-system.svc.cluster.local              80        -          outbound      EDS
istio-ingressgateway.istio-system.svc.cluster.local              443       -          outbound      EDS
istio-ingressgateway.istio-system.svc.cluster.local              853       -          outbound      EDS
...
istio-pilot.istio-system.svc.cluster.local                       8080      -          outbound      EDS
...
productpage.default.svc.cluster.local                            9080      -          inbound       STATIC
productpage.default.svc.cluster.local                            9080      -          outbound      EDS
ratings.default.svc.cluster.local                                9080      -          outbound      EDS
reviews.default.svc.cluster.local                                9080      -          outbound      EDS
xds-grpc                                                         -         -          -             STRICT_DNS
zipkin                                                           -         -          -             STRICT_DNS
zipkin.istio-system.svc.cluster.local                            9411      -          outbound      EDS
```

Mutual TLS

Public Key Cryptography

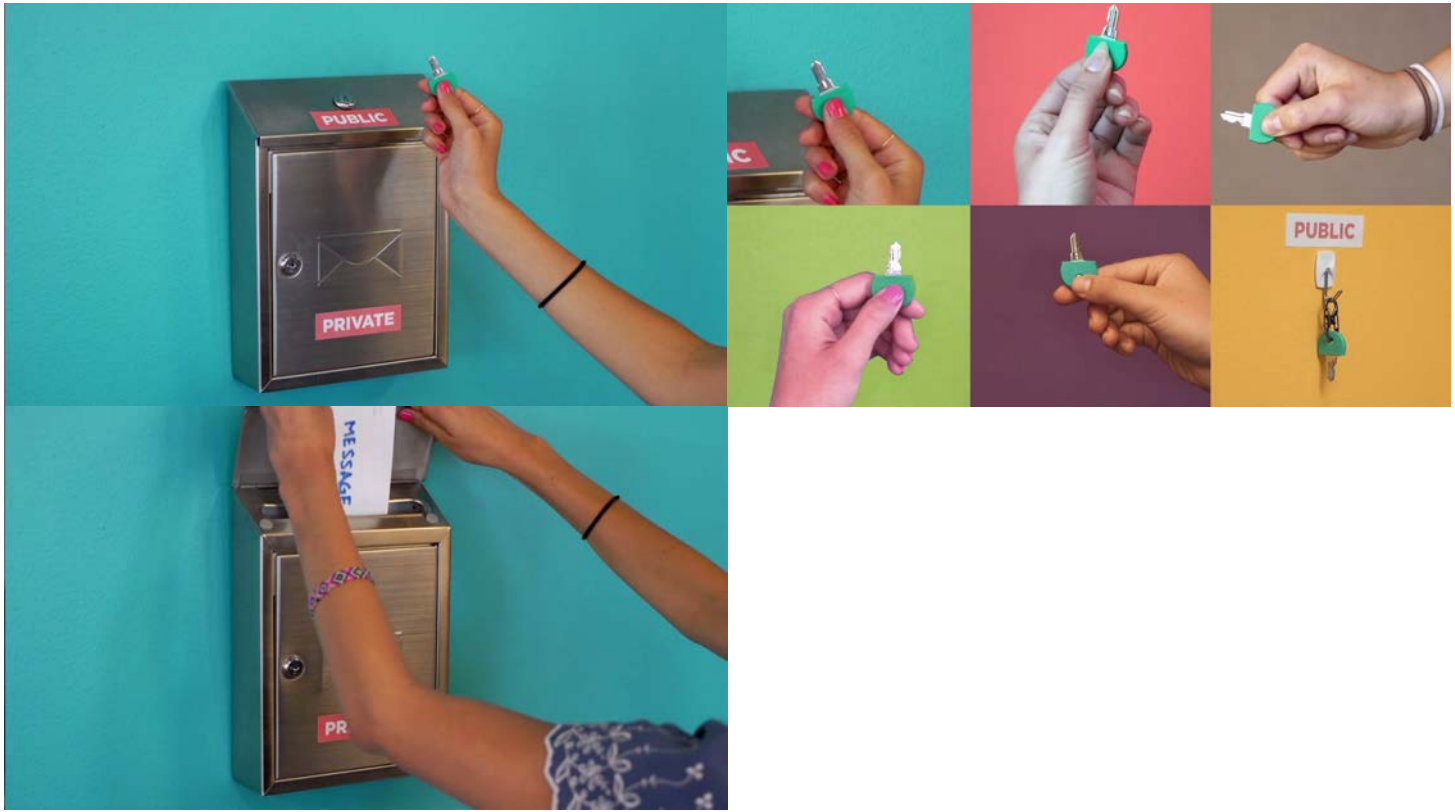
controlplane

Public Key Cryptography



controlplane

Public Key Cryptography

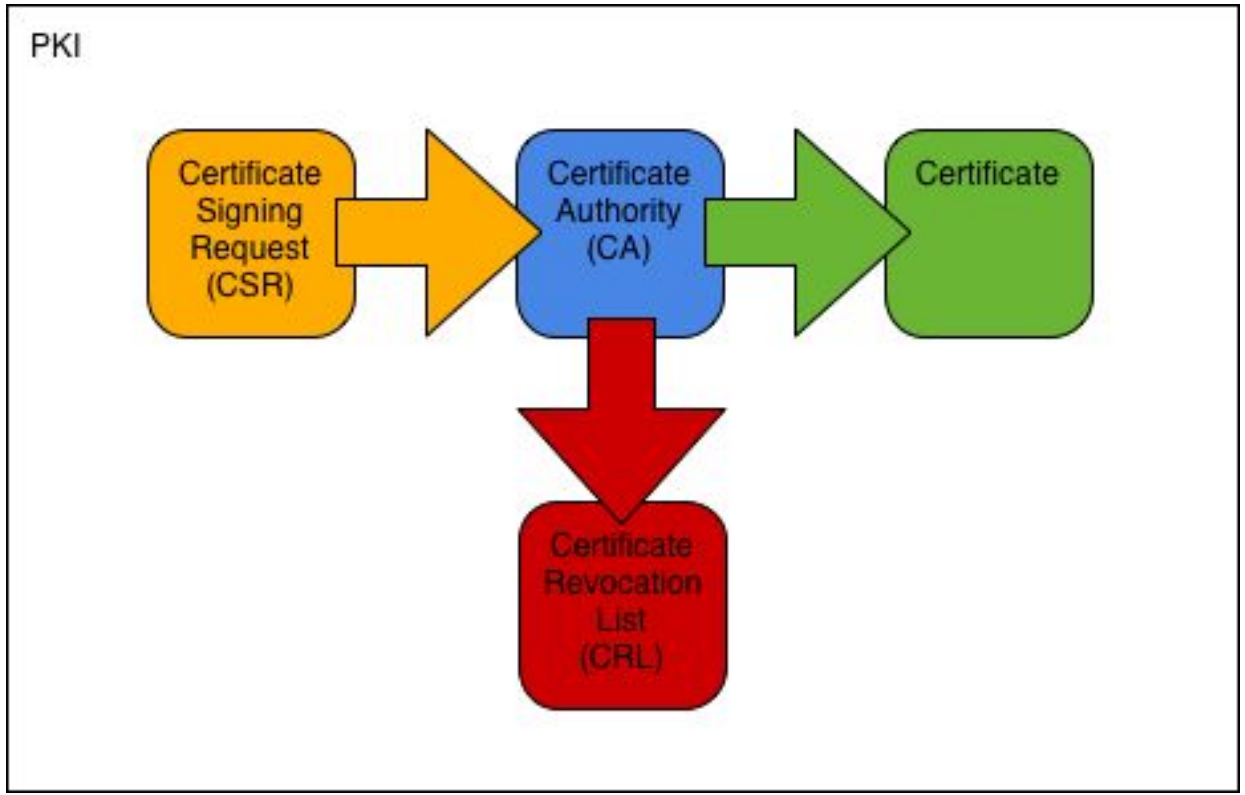


controlplane

Public Key Cryptography

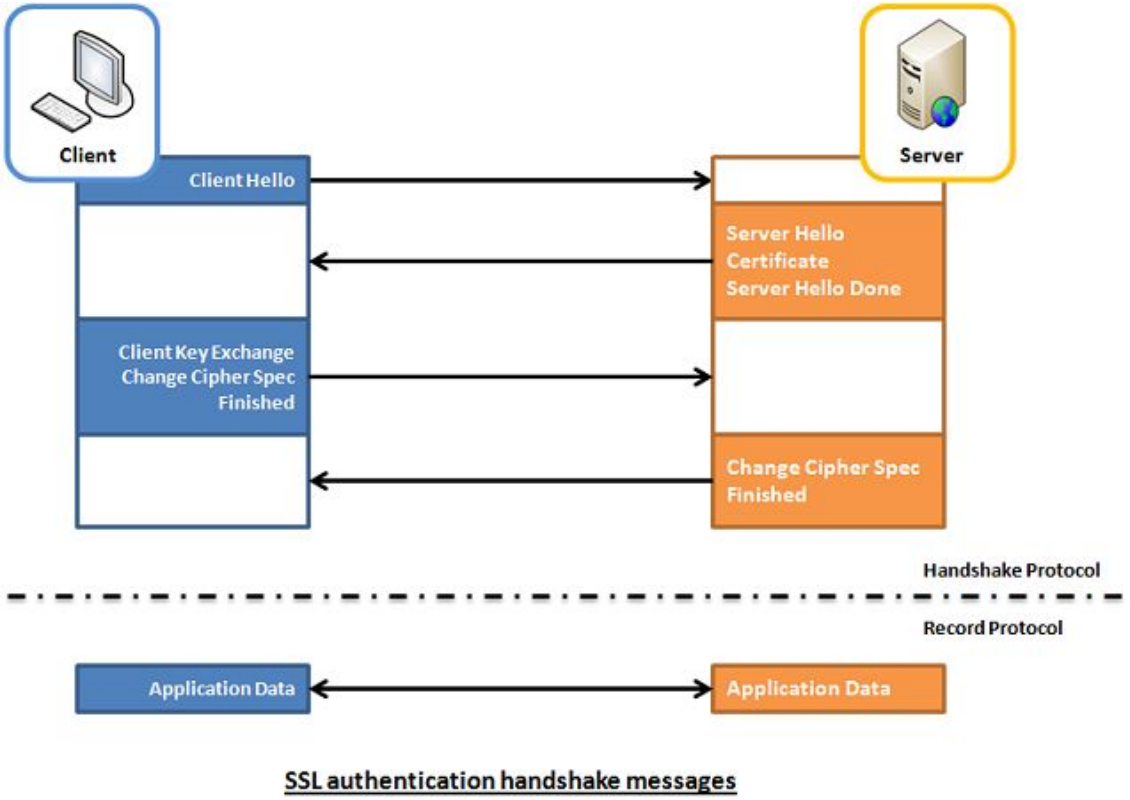
controlplane

Self Signed Certs aka Signing Your Own Homework



controlplane

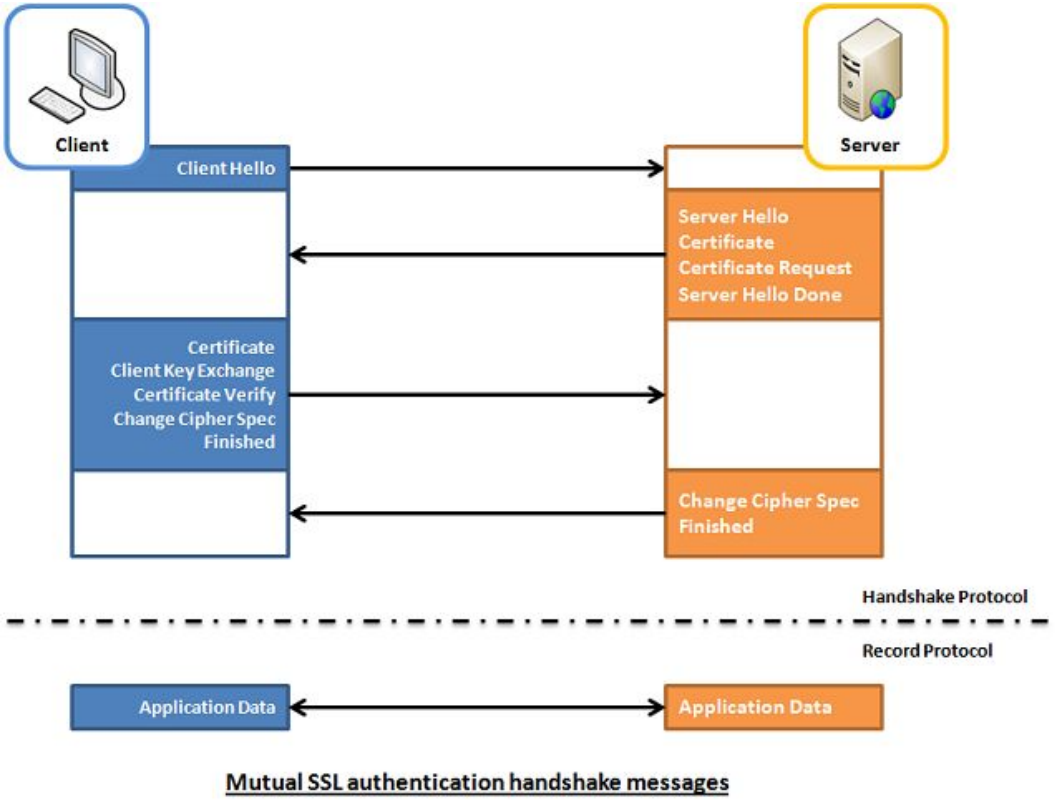
One-Way (Traditional) TLS Handshake



SSL authentication handshake messages

controlplane

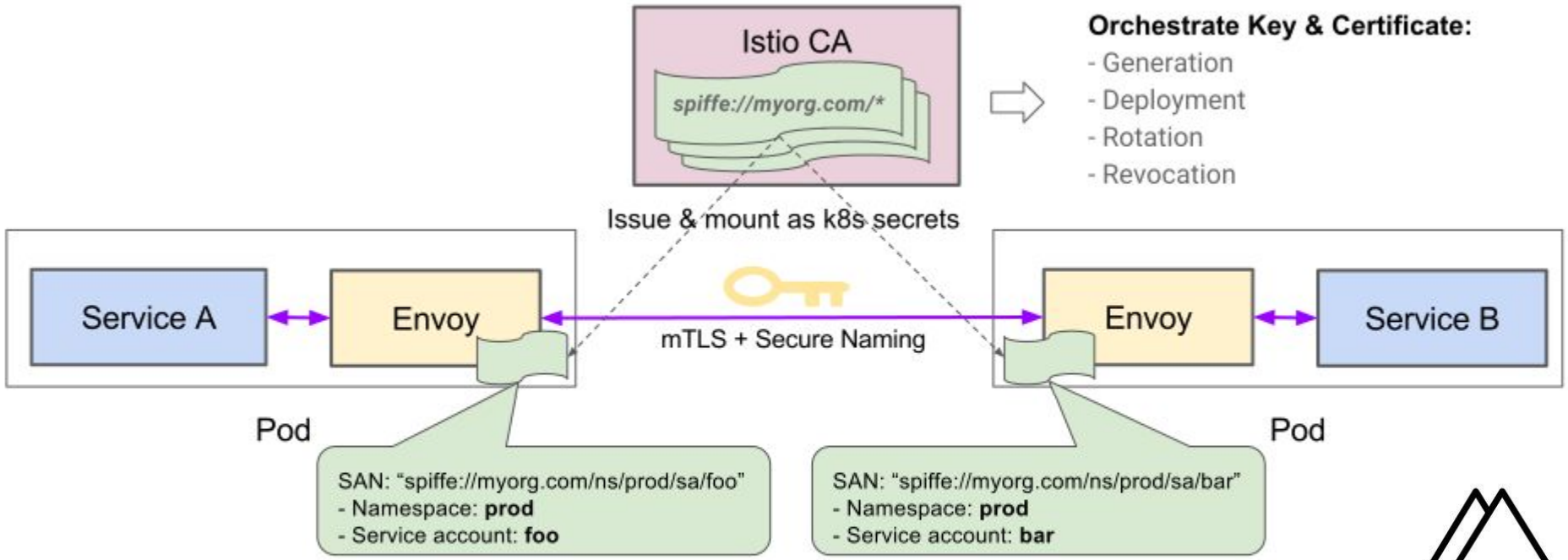
Mutual TLS Handshake (mTLS)



Mutual SSL authentication handshake messages

controlplane

Secure Naming



controlplane

Citadel

Citadel Image Filesystem

```
/mnt/docker/docker/overlay/.../root
├── etc
│   └── ssl
│       └── certs
│           └── ca-certificates.crt
├── tmp
└── usr
    ├── local
    │   └── bin
    │       └── istio_ca
    └── share
        └── doc
            └── ca-certificates
                └── copyright
```

```
/mnt/docker/docker/overlay/.../merged
/mnt/docker/docker/overlay/.../upper
├── dev
│   ├── console
│   ├── pts
│   └── shm
├── etc
│   ├── hostname
│   ├── hosts
│   ├── mtab -> /proc/mounts
│   └── resolv.conf
├── proc
└── sys
/mnt/docker/docker/overlay/.../work
└── work
```

Citadel Image Filesystem

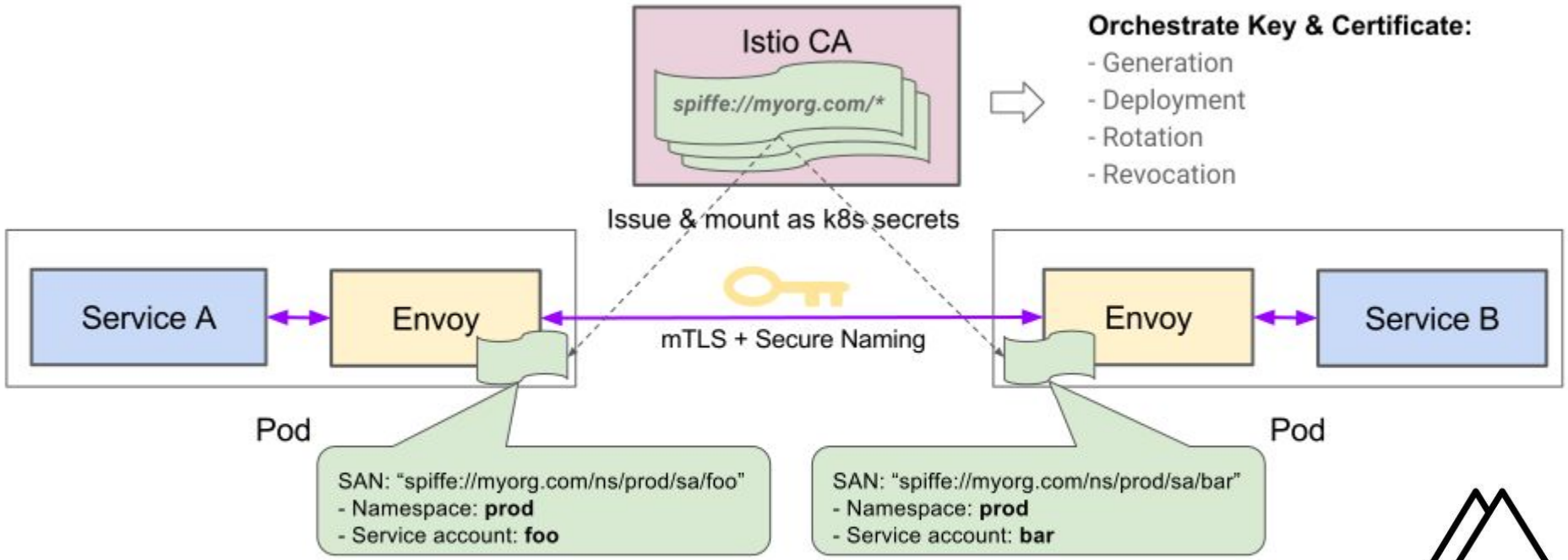
```
$ sudo sysdig 'proc.name=istio_ca and
evt.type=clone' --verbose | tr ' ' '\n'

702390
08:17:42.953506485
2
istio_ca
(20713)
<
clone
res=0
exe=/usr/local/bin/istio_ca
args=--self-signed-ca.
tid=20713(istio_ca)
pid=20650(istio_ca)
ptid=20632(docker-containe)
cwd=
# ...

fdlimit=4096
pgft_maj=0
pgft_min=0
vm_size=37380
vm_rss=22072
vm_swap=0
comm=istio_ca
cgroups=cpuset=/docker/61a4c7a10a431e7f227
1396633fb0757034be69f8eddd825280f3e9d0eb96
333...
flags=47107(CLONE_FILES|CLONE_FS|CLONE_SIG
HAND|CLONE_SYSVSEM|CLONE_THREAD|CLONE_VM)
uid=0
gid=0
vtid=12
vpid=1(systemd)
```

controlplane

Citadel Flow

1. Get the istio-system secret `istio-ca-secret`
2. Gets all service accounts and generates SPIFFE identifiers for them
   a. spiffe://cluster.local/ns/**kube-system**/sa/**service-controller**
   b. spiffe://cluster.local/ns/**default**/sa/**frontend-web-orders**
   c. spiffe://cluster.local/ns/**default**/sa/**frontend-web-addressbook**
   d. spiffe://cluster.local/ns/**istio-system**/sa/**istio-citadel-service-account**
3. Gets all services
4. Gets all secrets of type `istio.io/key-and-cert` for already-issued certificates
5. Sets a watch on the serviceaccounts, services, and secrets
6. Issue certs!

Secure Naming



controlplane

Bootstrapping identity with SPIFFE

SPIFFE ID

`spiffe://acme.com/billing/payments`

Trust Domain *Workload Identifier*

controlplane

SPIFFE ID: Istio

`spiffe://cluster.local/ns/`**`my-ns`**`/sa/`**`my-serviceaccount`**

Trust Domain

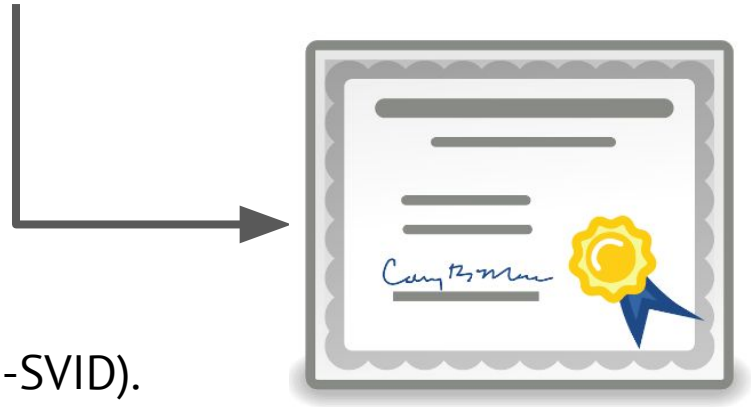
Workload Identifier

controlplane

SPIFFE Verifiable Identity Document (SVID)

`spiffe://acme.com/billing/payments`

Typically short-lived



Today only one form of SVID (X509-SVID).
Other document types under consideration
(including JWT-SVID)

controlplane

X.509 RFC Format

```
Certificate  ::=  SEQUENCE  {
     tbsCertificate       TBSCertificate,
     signatureAlgorithm   AlgorithmIdentifier,
     signatureValue       BIT STRING  }

TBSCertificate  ::=  SEQUENCE  {
     version         [0]  EXPLICIT Version DEFAULT v1,
     serialNumber         CertificateSerialNumber,
     signature            AlgorithmIdentifier,
     issuer               Name,
     validity             Validity,
     subject              Name,
     subjectPublicKeyInfo SubjectPublicKeyInfo,
     issuerUniqueID  [1]  IMPLICIT UniqueIdentifier OPTIONAL,
                          -- If present, version MUST be v2 or v3
     subjectUniqueID [2]  IMPLICIT UniqueIdentifier OPTIONAL,
                          -- If present, version MUST be v2 or v3
     extensions      [3]  EXPLICIT Extensions OPTIONAL
                          -- If present, version MUST be v3
     }

Version  ::=  INTEGER  {  v1(0), v2(1), v3(2)  }

CertificateSerialNumber  ::=  INTEGER
```

```
Validity ::= SEQUENCE {
     notBefore      Time,
     notAfter       Time }

Time ::= CHOICE {
     utcTime        UTCTime,
     generalTime    GeneralizedTime }

UniqueIdentifier  ::=  BIT STRING

SubjectPublicKeyInfo  ::=  SEQUENCE  {
     algorithm            AlgorithmIdentifier,
     subjectPublicKey     BIT STRING  }

Extensions  ::=  SEQUENCE SIZE (1..MAX) OF Extension

Extension  ::=  SEQUENCE  {
     extnID      OBJECT IDENTIFIER,
     critical    BOOLEAN DEFAULT FALSE,
     extnValue   OCTET STRING
                 -- contains the DER encoding of an ASN.1 value
                 -- corresponding to the extension type identified
                 -- by extnID
     }
```

https://github.com/spiffe/spiffe/blob/master/standards/X509-SVID.md#appendix-a-x509-field-reference

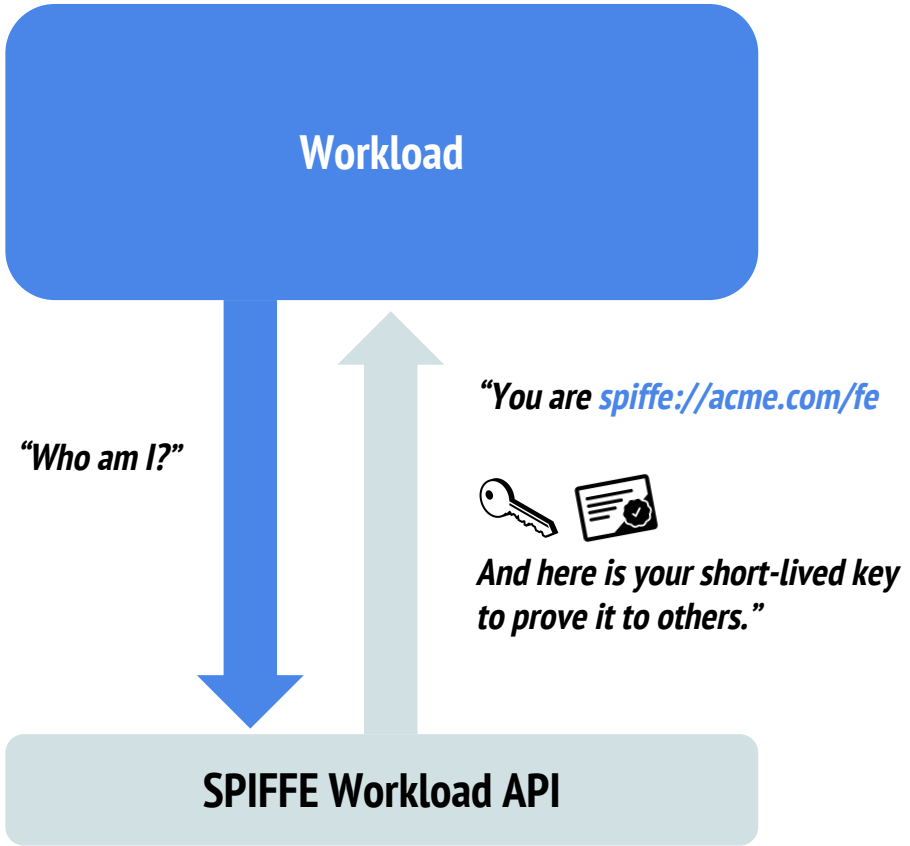
controlplane

spiffe

SCYTALE

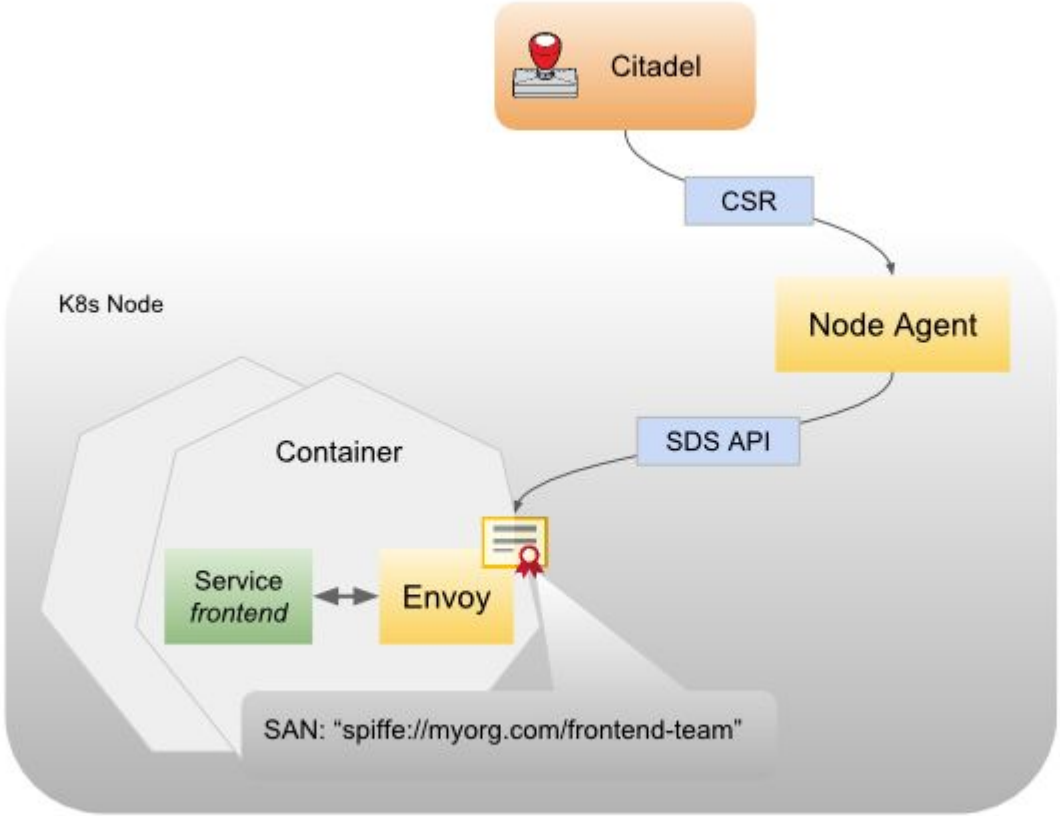
TiGERA

heptio



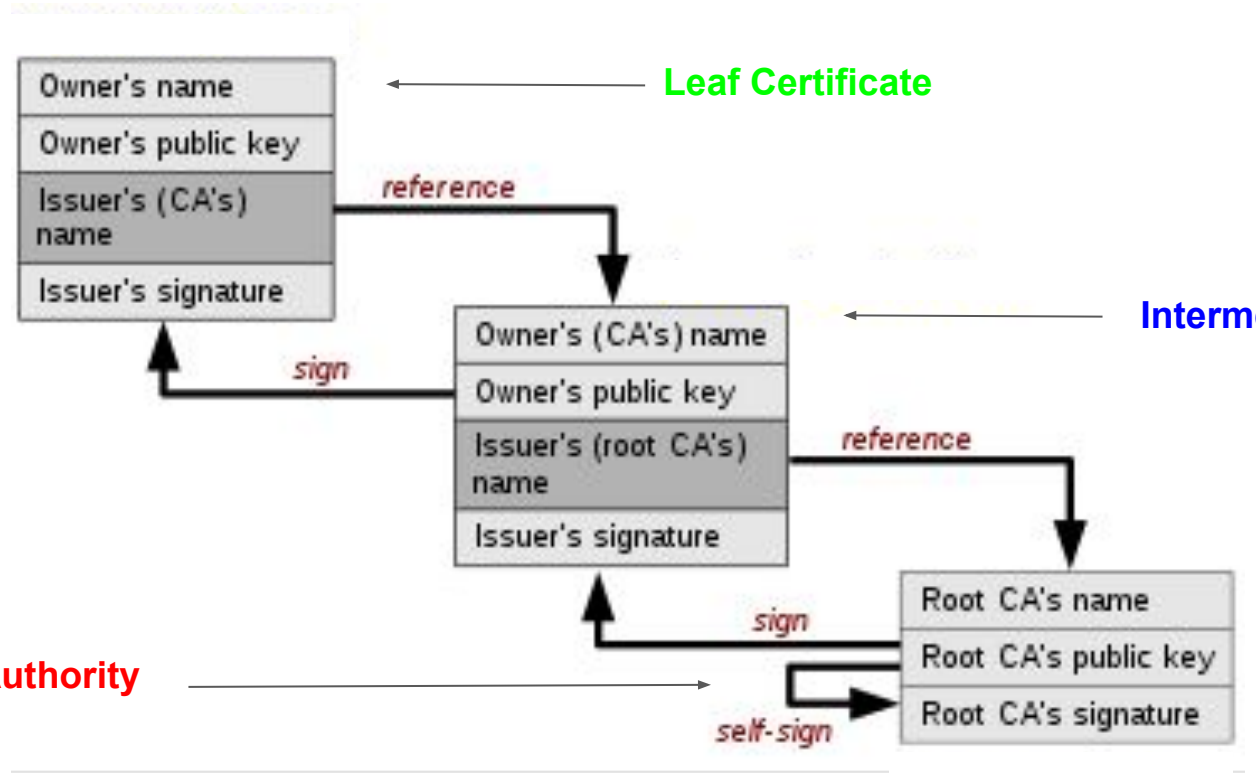
controlplane

Istio: Phase Two Attestation



controlplane

Certificate Path Validation

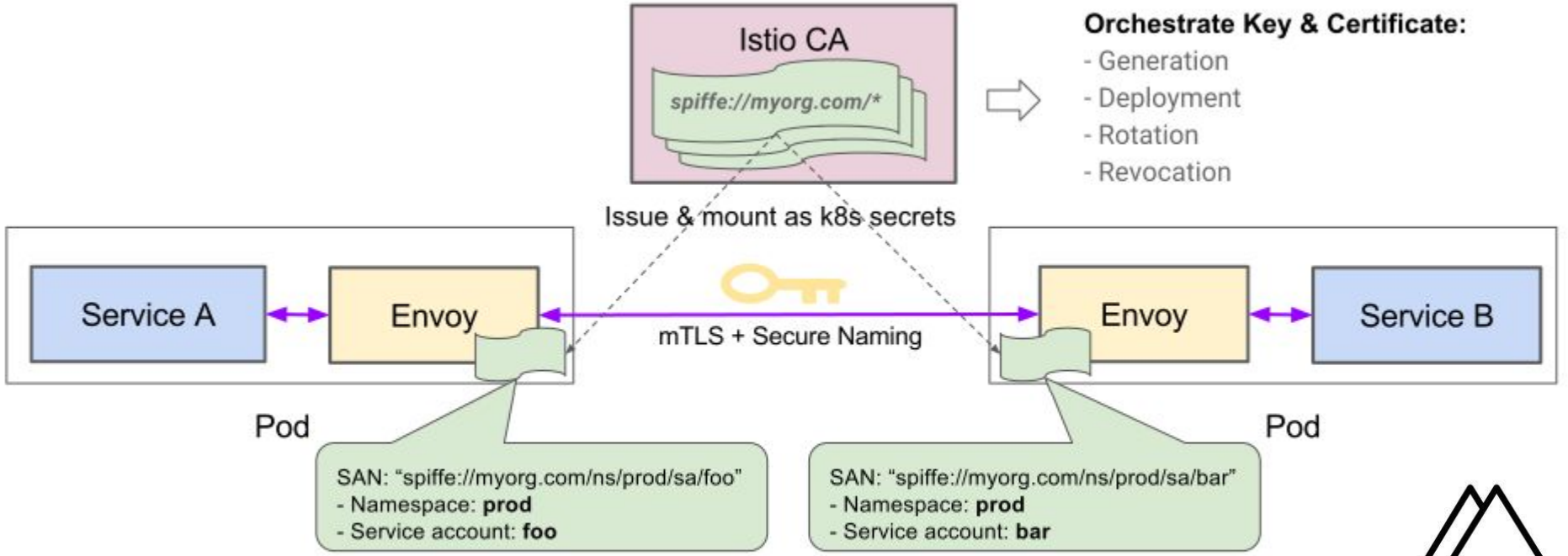


controlplane

What SPIFFE is not

- **Authorization** (however it provides identities upon which authorization schemes can be deployed)
- **Transport level security** (however SVIDs can be used to facilitate things like TLS or JWT signing)

Using SPIFFE in TLS Certificates



controlplane

TLS Config in Envoy

```
$ kubectl exec -it productpage-v1-55d65d9c4-f52c5 --namespace default -c istio-proxy -- curl localhost:15000/config_dump

{
 "configs": {
  "listeners": {
   "@type": "type.googleapis.com/envoy.admin.v2alpha.ListenersConfigDump",
   "version_info": "2018-09-11T11:53:29Z",
   "dynamic_active_listeners": [
    {
     "listener": {
      "name": "10.12.77.7_9080",
      "address": {
       "socket_address": { "address": "10.12.77.7", "port_value": 9080 }
      },
      "filter_chains": [
       {
        "tls_context": {
         "common_tls_context": {
          "tls_certificates": [
           {
            "certificate_chain": {
             "filename": "/etc/certs/cert-chain.pem"
            },
            "private_key": {
             "filename": "/etc/certs/key.pem"
            }
           }
          ],

# ...
```

```
# ...

   "validation_context": {
    "trusted_ca": {
     "filename": "/etc/certs/root-cert.pem"
    }
   },
   "alpn_protocols": [
    "h2",
    "http/1.1"
   ]
  },
  "require_client_certificate": true
 },
```

controlplane

Secure By Default

Security: RBAC (aka Authorisation)

```yaml
apiVersion: "rbac.istio.io/v1alpha1"
kind: RbacConfig
metadata:
 name: default
spec:
 mode: 'ON_WITH_INCLUSION'
 inclusion:
   namespaces: ["my-istio-ns"]
```

controlplane

Security: Namespace RBAC

```yaml
apiVersion: "rbac.istio.io/v1alpha1"
kind: ServiceRole
metadata:
  name: service-viewer
  namespace: default
spec:
  rules:
  - services: ["*"]
    methods: ["GET"]
    constraints:
    - key: "destination.labels[app]"
      values: ["productpage", "details",
"reviews", "ratings"]

# ...
```

```yaml
---
apiVersion: "rbac.istio.io/v1alpha1"
kind: ServiceRoleBinding
metadata:
  name: bind-service-viewer
  namespace: default
spec:
  subjects:
  - properties:
      source.namespace: "istio-system"
  - properties:
      source.namespace: "my-istio-ns"
  roleRef:
    kind: ServiceRole
    name: "service-viewer"
```

controlplane

Security: Per-Service RBAC

```yaml
apiVersion: "rbac.istio.io/v1alpha1"
kind: ServiceRole
metadata:
  name: productpage-viewer
  namespace: default
spec:
  rules:
  - services: ["productpage.default.svc.cluster.local"]
    methods: ["GET", "HEAD"]
    paths: ["*"]
    constraints:
    - key: request.headers[version]
      values: ["v1", "v2"]

# ...
```

```yaml
---
apiVersion: "rbac.istio.io/v1alpha1"
kind: ServiceRoleBinding
metadata:
  name: bind-productpage-viewer
  namespace: default
spec:
  subjects:
  - user: "*"
  roleRef:
    kind: ServiceRole
    name: "productpage-viewer"
```

controlplane

Security: Per-Service RBAC in Envoy

```
$ kubectl exec -it productpage-v1-55d65d9c4-f52c5 --namespace default -c istio-proxy -- curl localhost:15000/config_dump
```

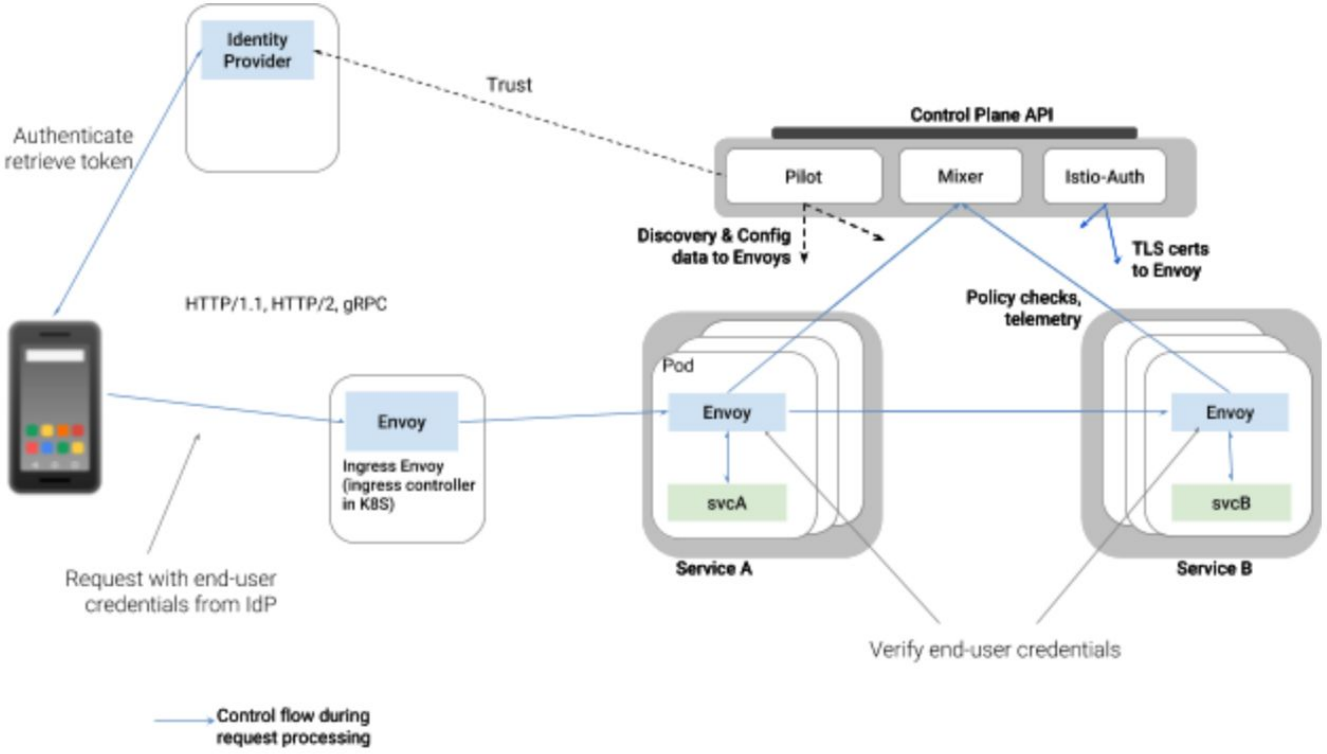

```
{
  "http_filters": [
    {
      "name": "istio_authn",
      "config": {
        "policy": {
          "peers": [
            {
              "mtls": {}
            }
          ]
        }
      }
    },
    {
      "name": "envoy.filters.http.rbac",
      "config": {
        "rules": {
          "policies": {
            "productpage-viewer": {
              "permissions": [
                {
                  "and_rules": {
                    "rules": [
                      {
                        "or_rules": {
                          "rules": [
                            {
                              "header": {
                                "exact_match": "GET",
                                "name": ":method"
                              }
                            },
                            {
                              "header": {
                                "name": ":method",
                                "exact_match": "HEAD"
                              }}]},
                            {
                              "or_rules": {
                                "rules": [
                                  {
                                    "header": {
                                      "present_match": true,
                                      "name": ":path"
                                    }}]},
                            {
                              "or_rules": {
                                "rules": [
                                  {
                                    "header": {
                                      "name": "version",
                                      "exact_match": "v1"
                                    }},
                                  {
                                    "header": {
                                      "exact_match": "v2",
                                      "name": "version"
                                    }}]}]}],
                    "principals": [
                      {
                        "and_ids": {
                          "ids": [{ "any": true }]}}
                    ]}}},
              "shadow_rules": {
                "policies": {}
              }}}
  ]
}
```


controlplane

Security: End User Authentication

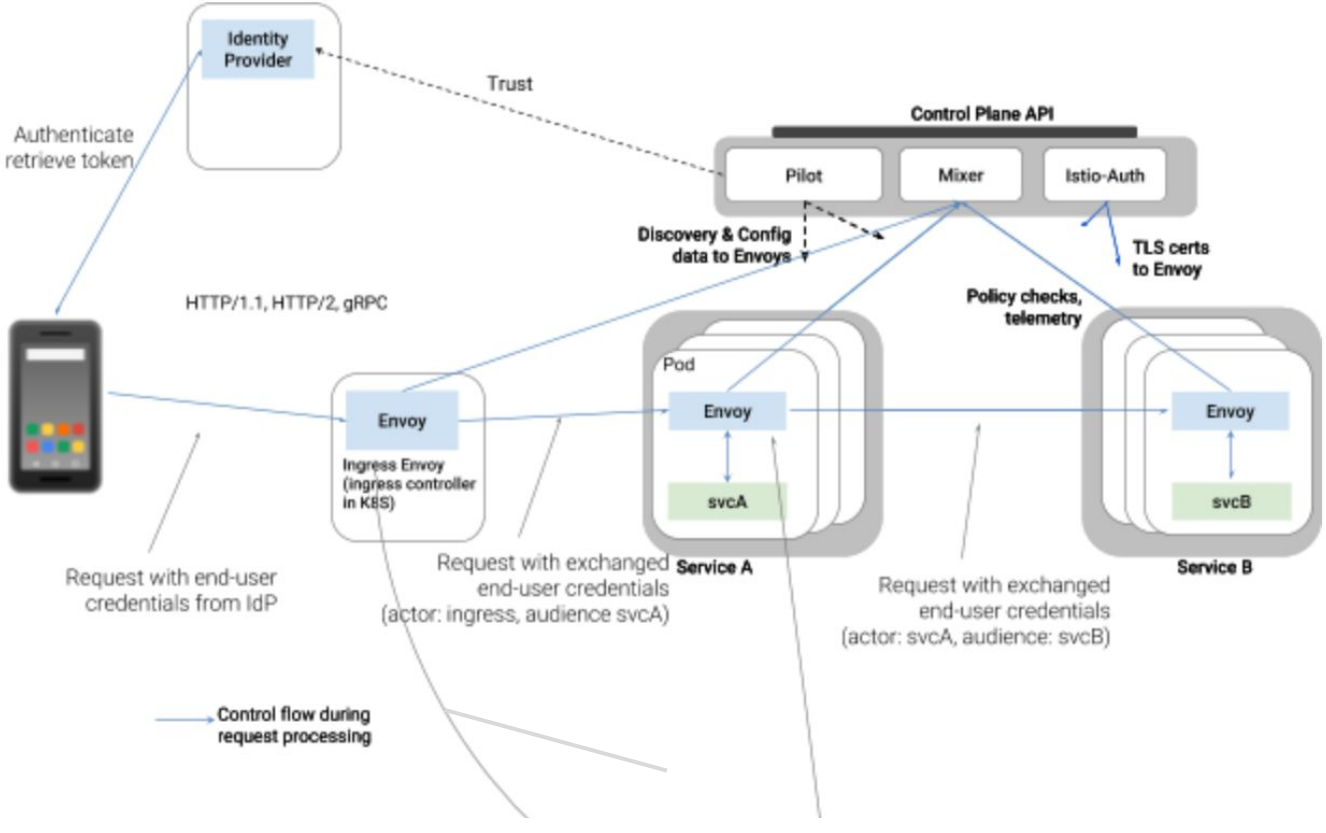
- Provide an unified form of identity associated with each request
- Provide a delegation mechanisms for service-to-service requests which originate from end-users

Security: End User Authentication (Phase 1)



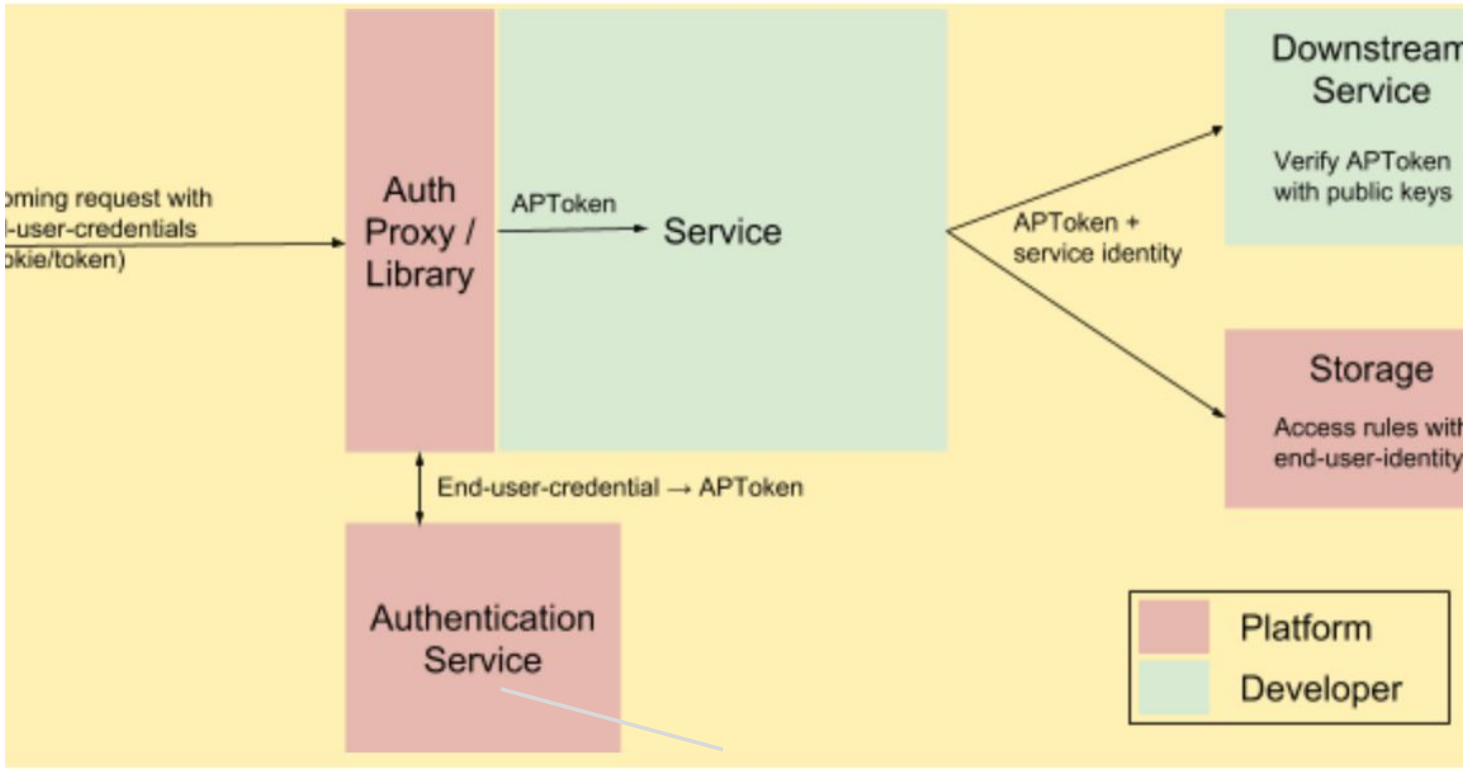
controlplane

Security: End User Authentication (Phase 2)



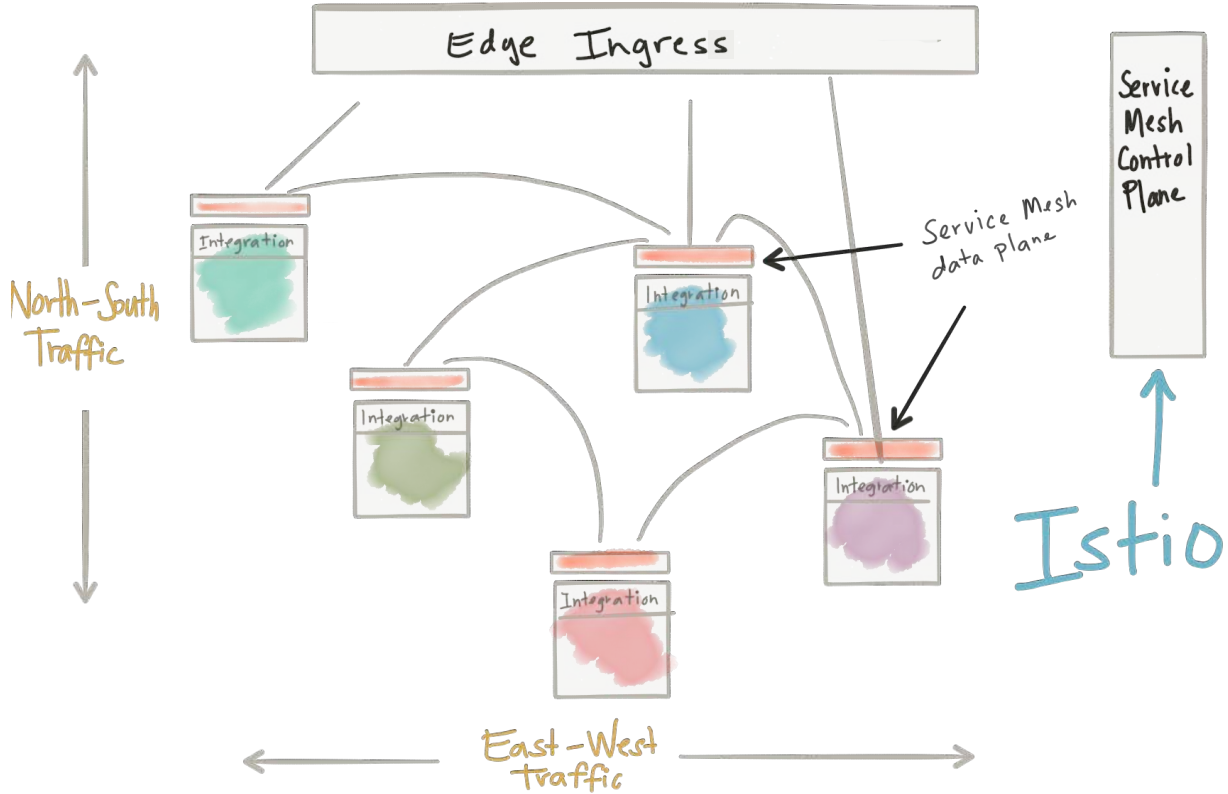
controlplane

Security: Authentication Proof Token



controlplane

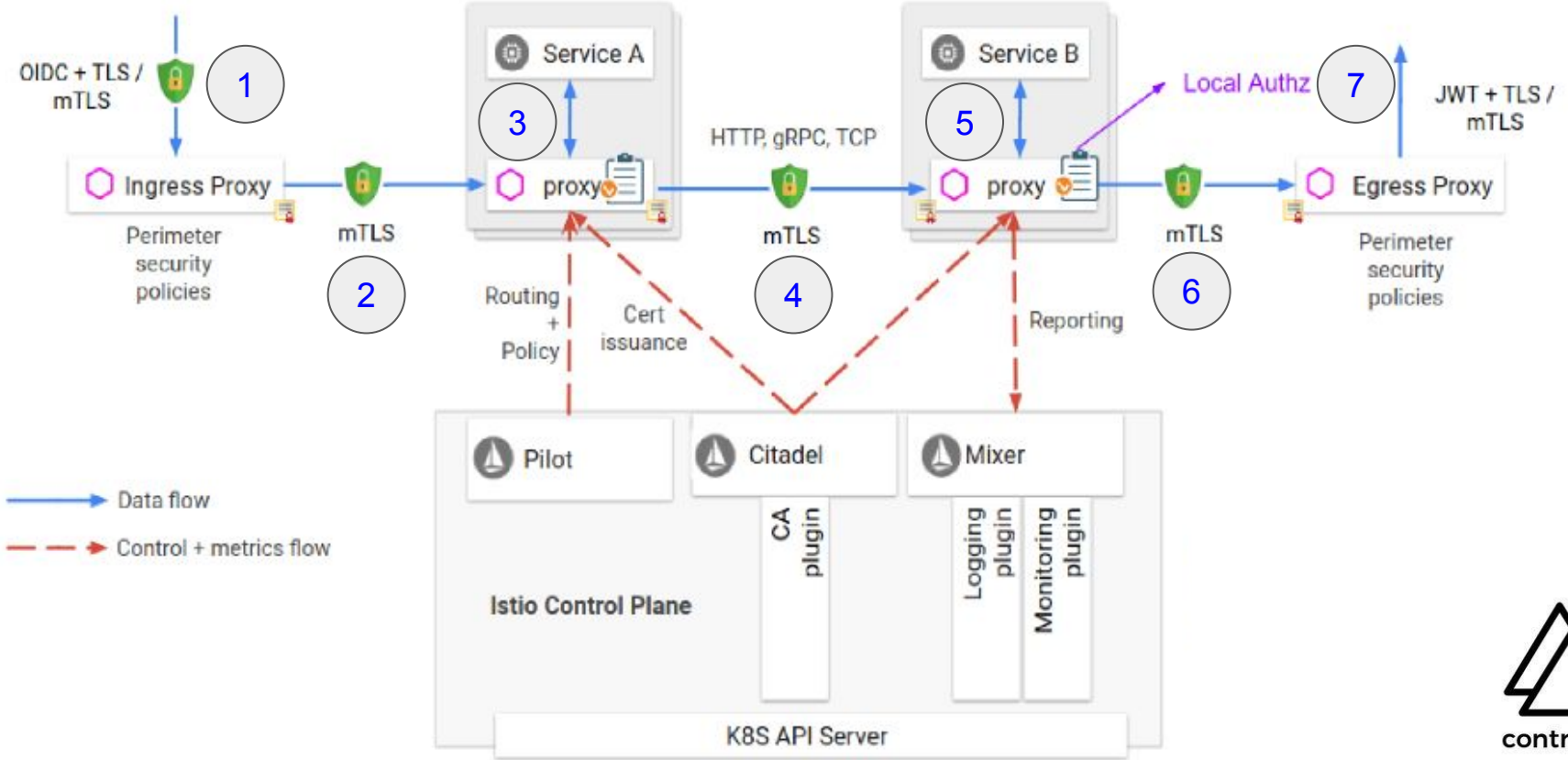
Traffic Management: Ingress Traffic



http://blog.christianposta.com/how-a-service-mesh-can-help-with-microservices-security/

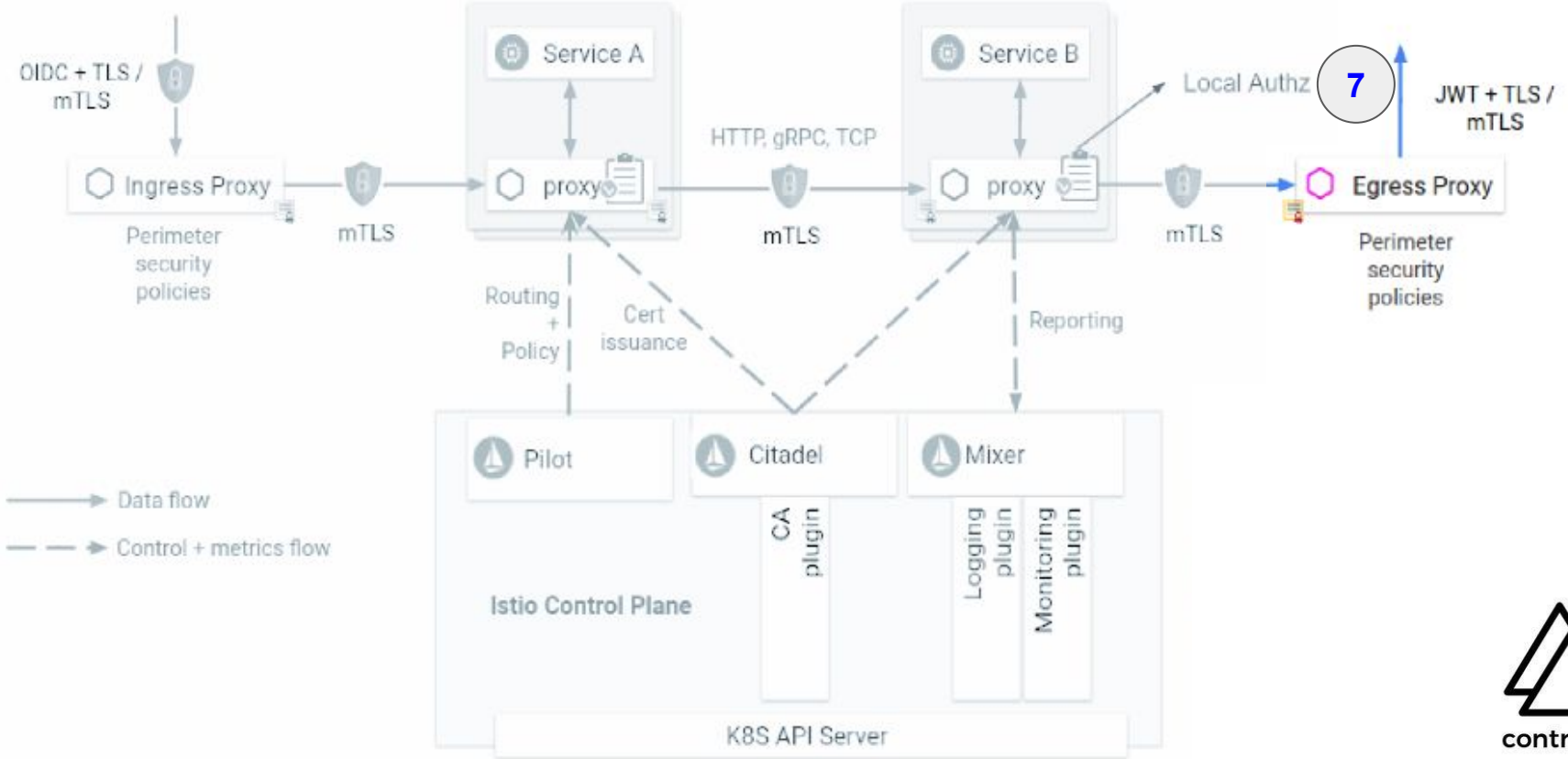
controlplane

Traffic Management: Egress Traffic



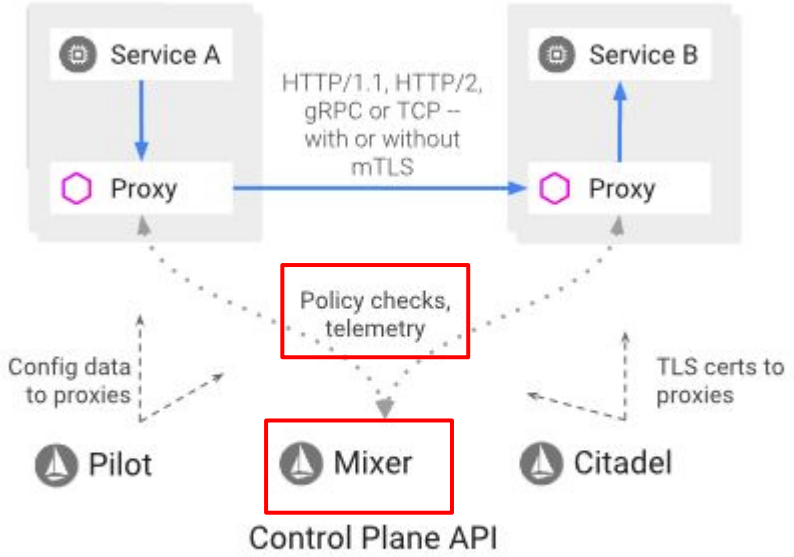
controlplane

Traffic Management: Egress Traffic



controlplane

Policy: Rate Limiting



controlplane

Policy: Blacklists and Whitelists

```yaml
apiVersion: config.istio.io/v1alpha2
kind: denier
metadata:
 name: denyreviewsv3handler
spec:
 status:
   code: 7
   message: Not allowed

---
apiVersion: config.istio.io/v1alpha2
kind: checknothing
metadata:
 name: denyreviewsv3request
spec:

---
apiVersion: config.istio.io/v1alpha2
kind: rule
metadata:
 name: denyreviewsv3
spec:
 match: destination.labels["app"] == "ratings" && source.labels["app"]=="reviews" && source.labels["version"] == "v3"
 actions:
 - handler: denyreviewsv3handler.denier
   instances: [ denyreviewsv3request.checknothing ]
```

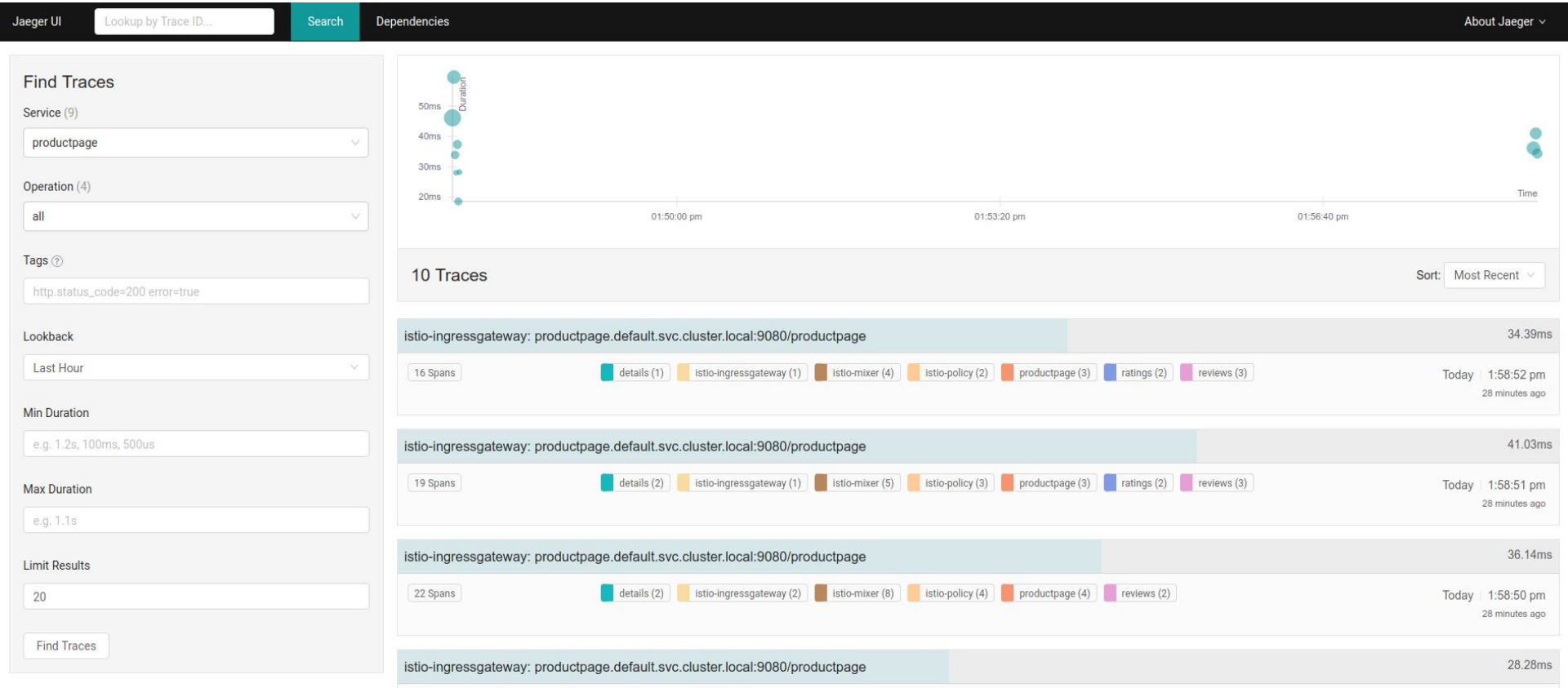
controlplane

Policy: Blacklists and Whitelists

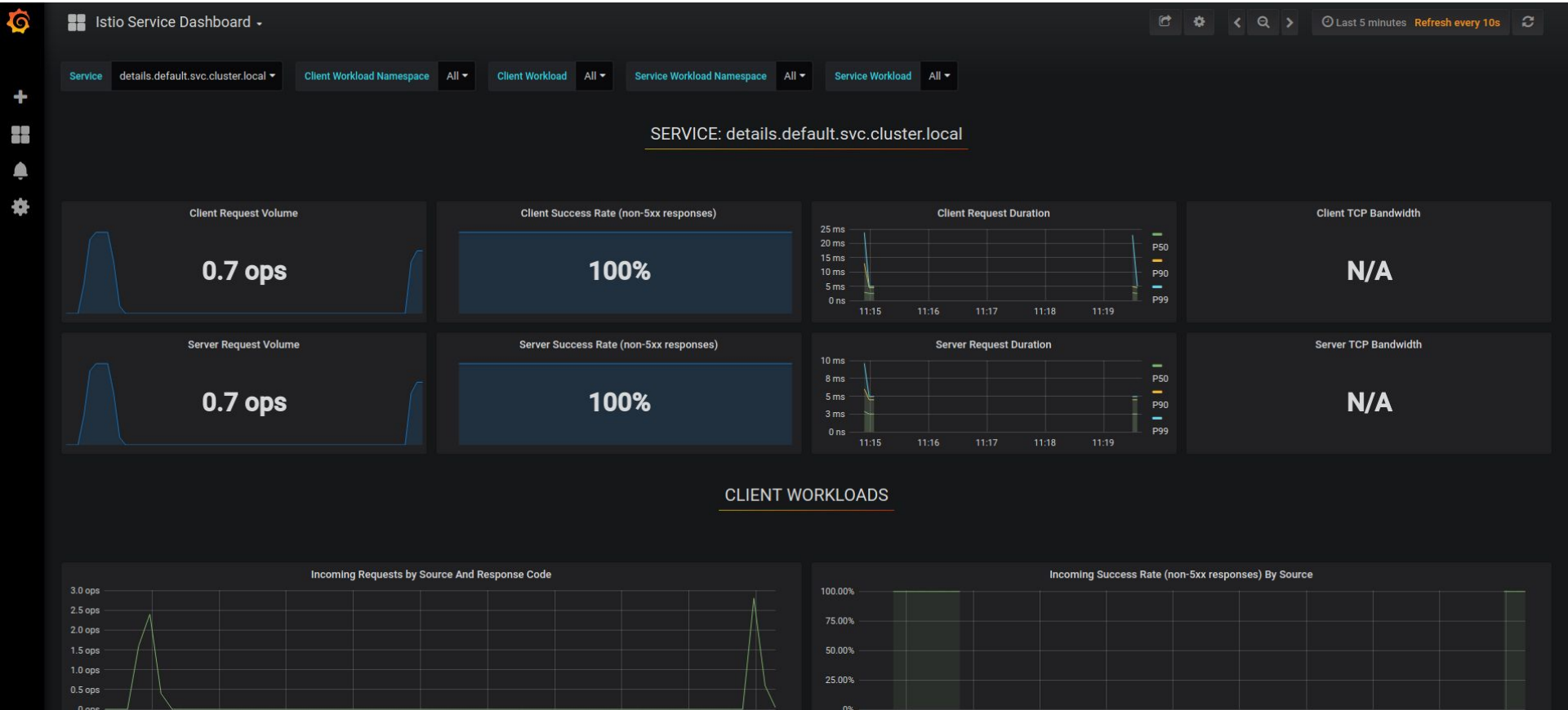
```yaml
apiVersion: config.istio.io/v1alpha2
kind: listchecker
metadata:
 name: whitelist
spec:
 # providerUrl: ordinarily black and white lists are maintained
 # externally and fetched asynchronously using the providerUrl.
 overrides: ["v1", "v2"]  # overrides provide a static list
 blacklist: false

---

apiVersion: config.istio.io/v1alpha2
kind: listentry
metadata:
 name: appversion
spec:
 value: source.labels["version"]

# ...
```

```yaml
# ...

---
apiVersion: config.istio.io/v1alpha2
kind: rule
metadata:
 name: checkversion
spec:
 match: destination.labels["app"] == "ratings"
 actions:
 - handler: whitelist.listchecker
   instances:
   - appversion.listentry
```

Extras: Telemetry



Extras: Metrics



What’s Next?

- Multi-cloud and multi-environment
- Reduction of component privilege (proxy init)
- Networking (extension models, UDP, QUIC, more performance)
- Further integrations (ACLs, telemetry, audit, policy)
- Enhanced security (HSM, Cert & Key stores, federation)
- Extended Authentication Policy (end-user authn, mTLS & JWT, federation)
- API Management

Thanks! For more...

Speaker: Pi Unnerup
@piunnerup
Talk author: Andrew Martin
@sublimino

controlplane
@controlplaneio
https://control-plane.io