ASPEN MESH

How to Secure and Monitor External Service Access With a Service Mesh

Neeraj Poddar
Co-founder & Engineering Lead, Aspen Mesh

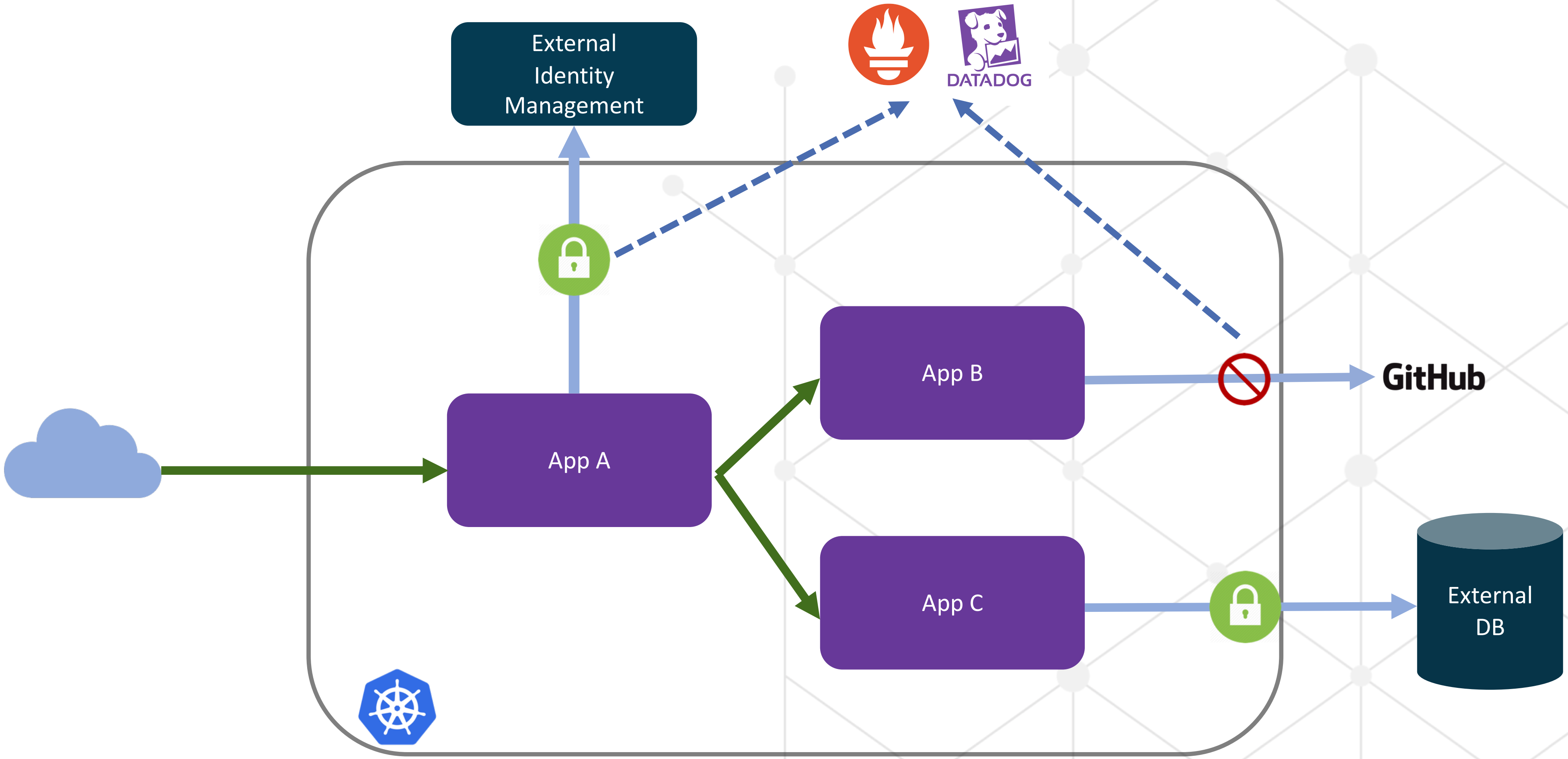
Why does this matter?

WE HAD A MASSIVE DATA BREACH. HACKERS GOT INTO THE PRIVATE DATA OF ALL OF OUR CUSTOMERS.
NO PROBLEM. WE'LL ISSUE A PRESS RELEASE THAT SAYS WE'RE SORRY AND IT WILL NEVER HAPPEN AGAIN.
THAT'S WHAT WE SAID THE LAST THREE TIMES IT HAPPENED.
OUR STRATEGY IS TO WEAR THEM DOWN.
DILBERT.COM @SCOTTADAMSSAYS
5-7-18 ©2018 Scott Adams, Inc./Dist. by Andrews McMeel

OWASP Top Ten

- Using components with Known Vulnerabilities. PyPI Example
- Insufficient Logging & Monitoring
- Security Misconfigurations

Desired State



Goals for External Service Access

- How do you know **what external services** you’re connecting?
- How can you **secure access** to those services?
- How can you **block unauthorized** access?

Ways to Control External Service Access

- Embed the logic in Application Code
- Use Third Party or OSS libraries
- Offload this functionality to the infrastructure layer

What’s a Service Mesh?

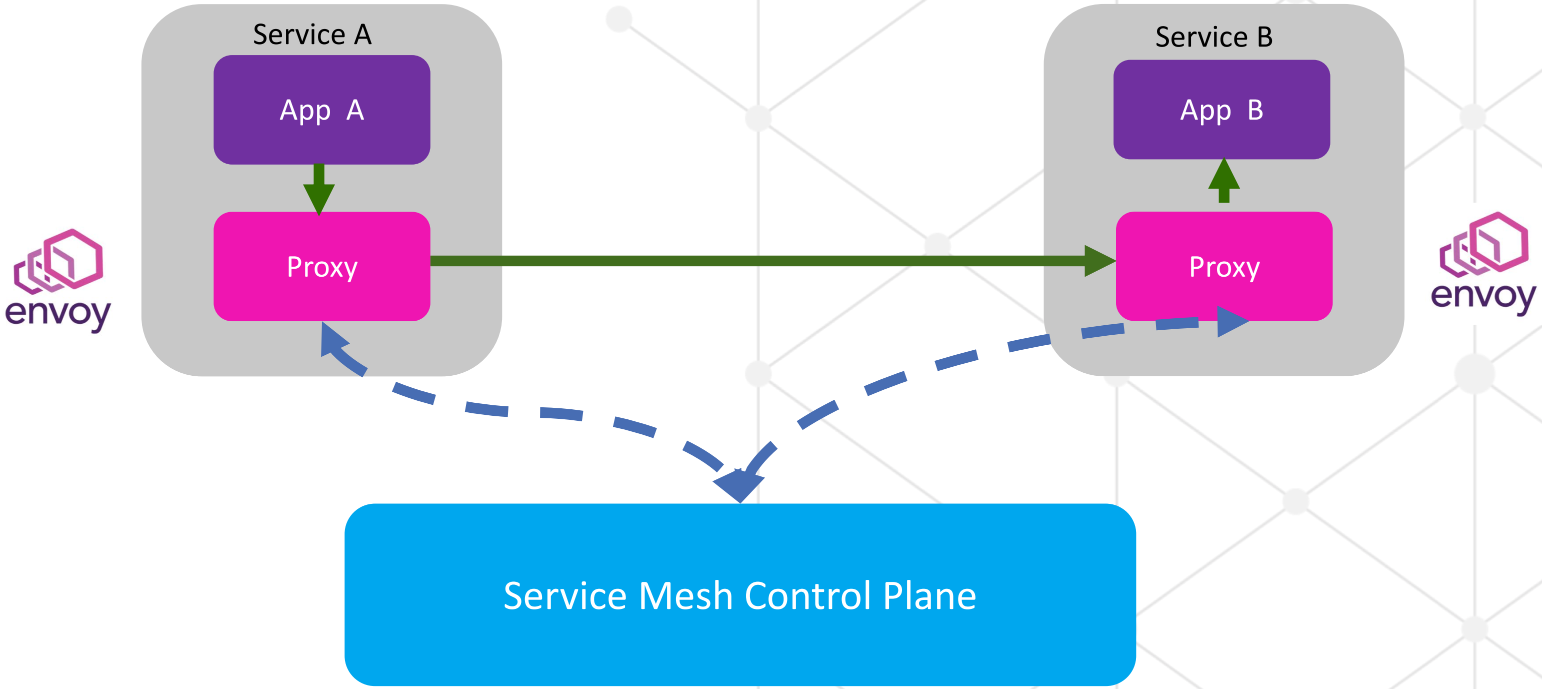
- A transparent **infrastructure layer** that manages communication between microservices

- So that **developers** can focus on business logic

- While **operators** work independent of dev cycles to provide a more resilient environment

Sidecar Proxy Architecture

Sidecar Proxy Architecture in

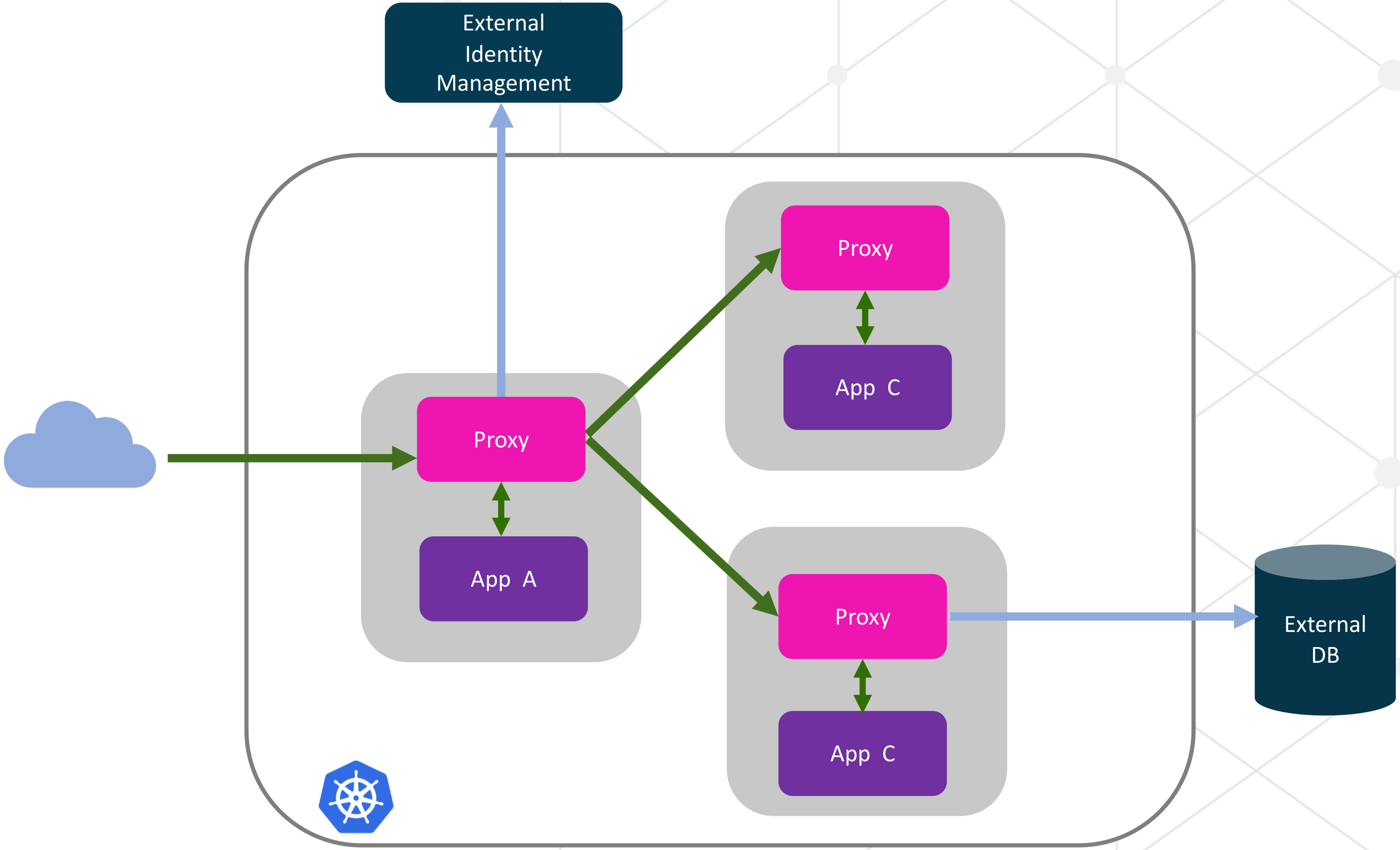


Traffic Management

Security

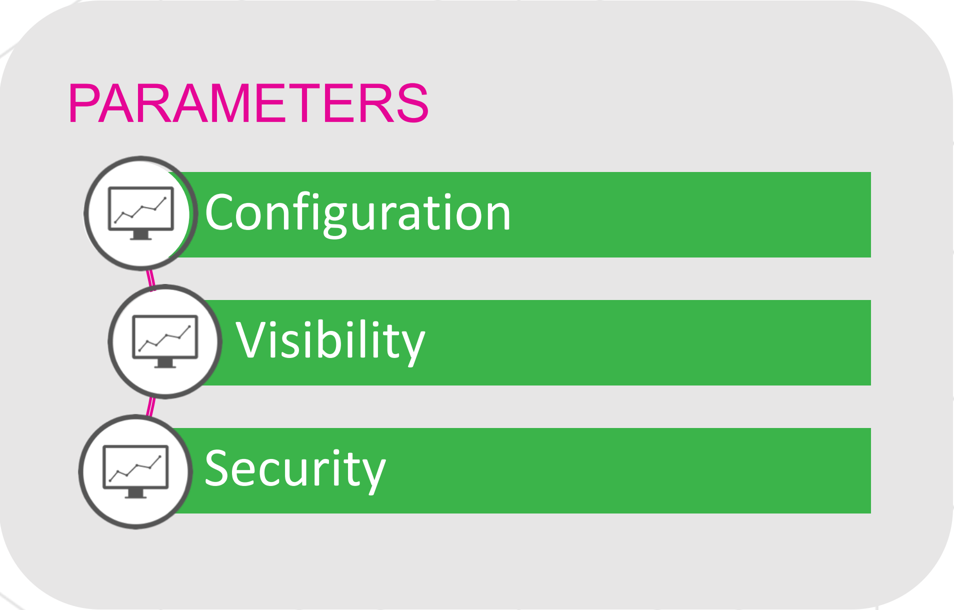
Observability

Service Mesh + External Services



Various Architecture Options in

- Allow any
- Restricted access with TLS passthrough
- Restricted access with TLS origination
- Egress gateway with TLS origination



PARAMETERS

- Configuration
- Visibility
- Security

Option 1: Allow Any

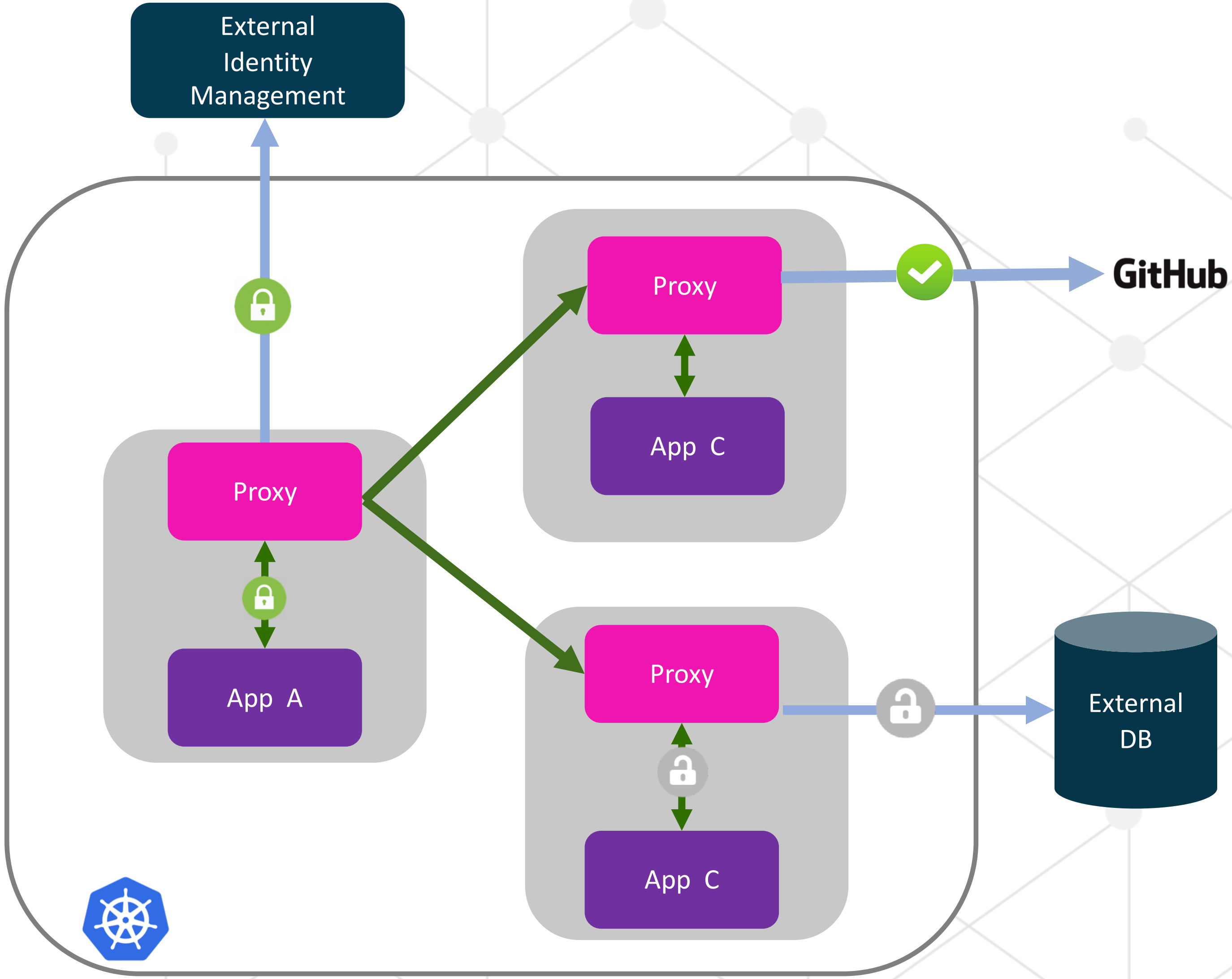
PROS

Configuration: Zero config 🟩

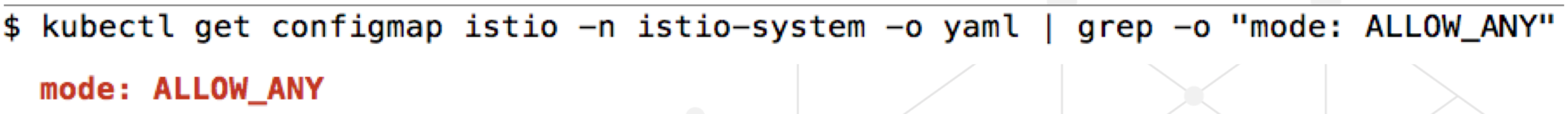
CONS

Visibility: TCP metrics 🟥

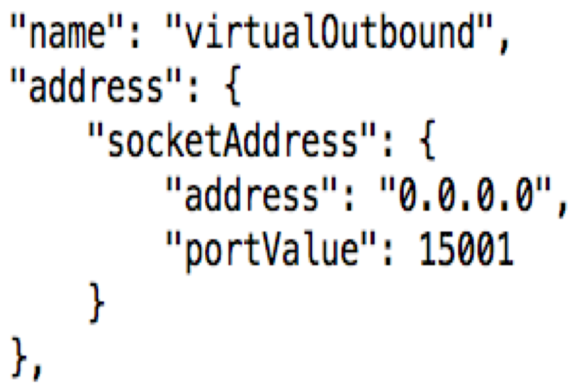
Security: Not secure 🟥



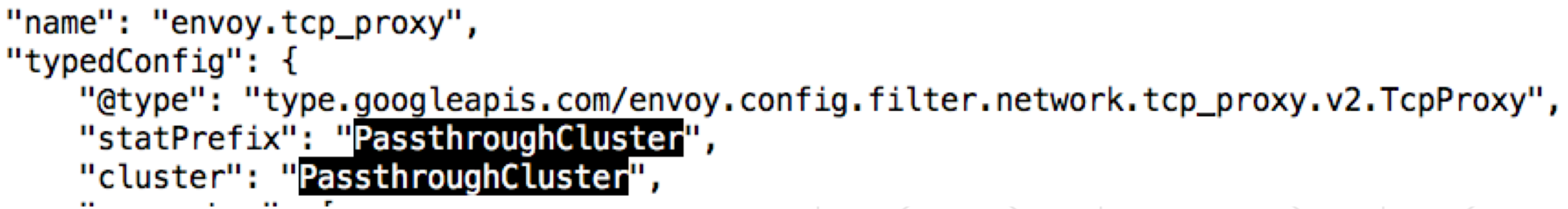
Option 1: Allow Any in



```
$ kubectl get configmap istio -n istio-system -o yaml | grep -o "mode: ALLOW_ANY"
  mode: ALLOW_ANY
```



```
"name": "virtualOutbound",
"address": {
    "socketAddress": {
        "address": "0.0.0.0",
        "portValue": 15001
    }
},
```



```
"name": "envoy.tcp_proxy",
"typedConfig": {
    "@type": "type.googleapis.com/envoy.config.filter.network.tcp_proxy.v2.TcpProxy",
    "statPrefix": "PassthroughCluster",
    "cluster": "PassthroughCluster",
```

Option 2: Restricted Access with TLS Passthrough

PROS

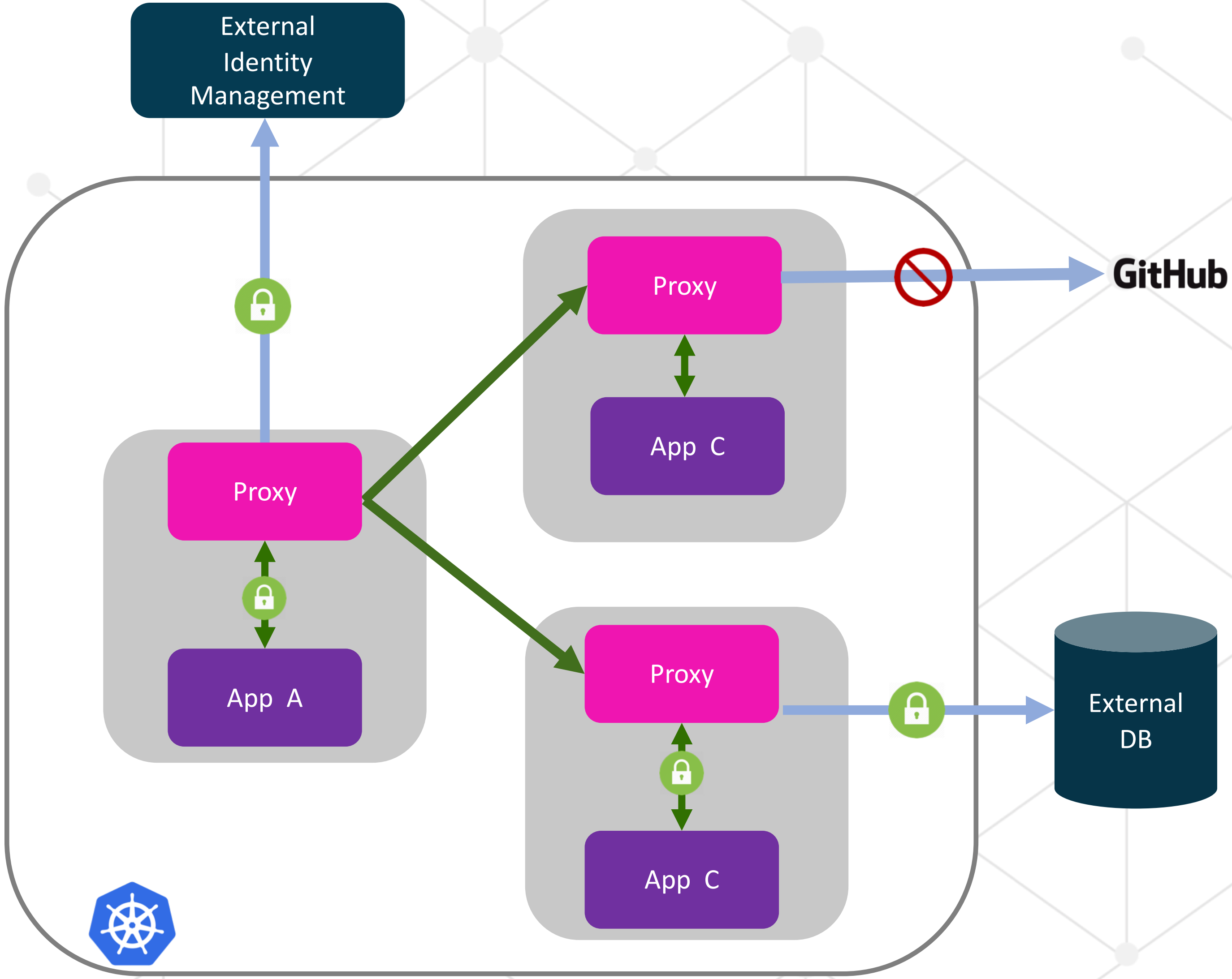
Security: Secure 🟩

CONS

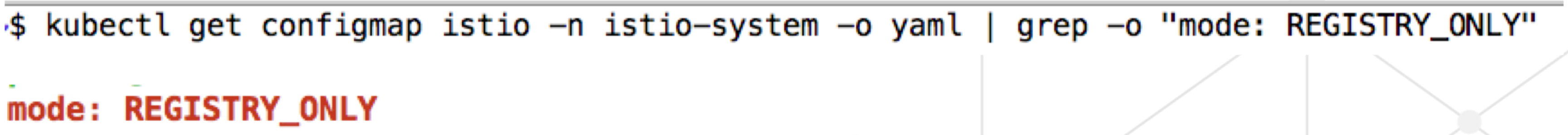
Configuration: Per service 🟥

Visibility: TCP metrics 🟥

Security: Application TLS stack 🟥

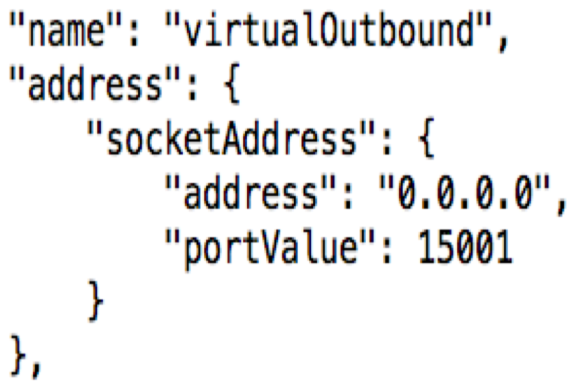


Option 2: Restricted Access with TLS Passthrough in

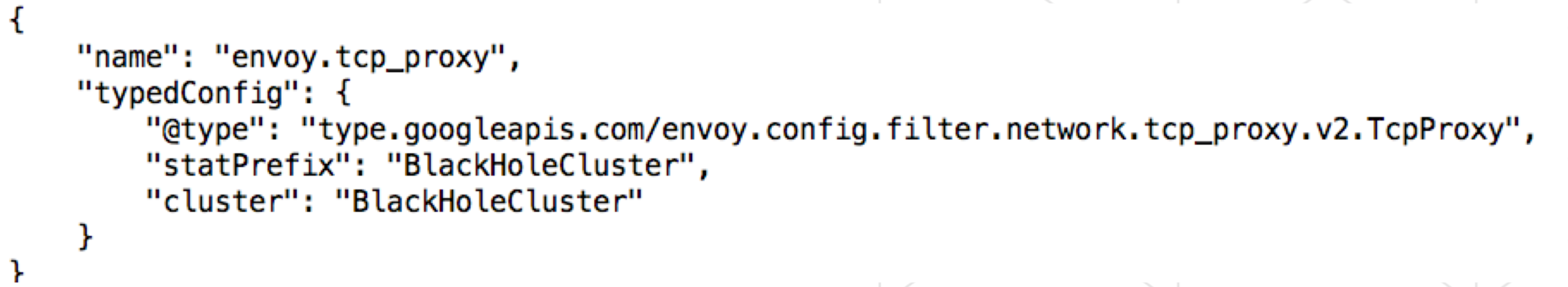


```
$ kubectl get configmap istio -n istio-system -o yaml | grep -o "mode: REGISTRY_ONLY"

mode: REGISTRY_ONLY
```

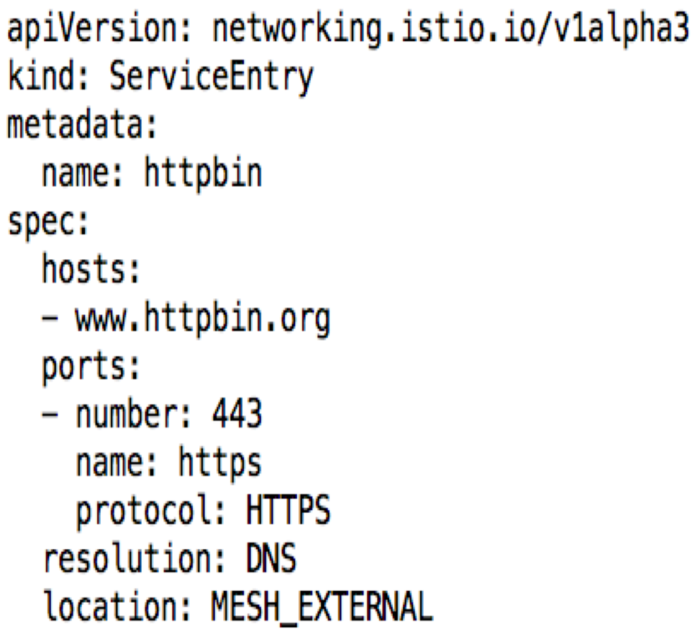


```
"name": "virtualOutbound",
"address": {
    "socketAddress": {
        "address": "0.0.0.0",
        "portValue": 15001
    }
},
```

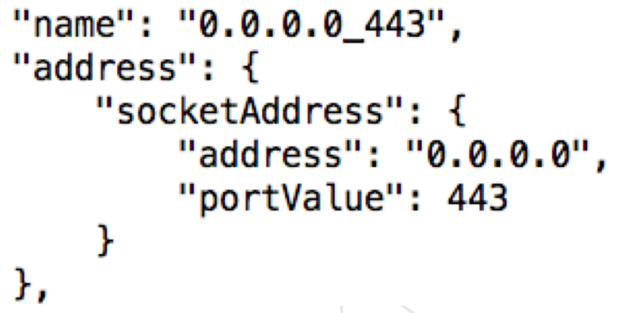


```
{
    "name": "envoy.tcp_proxy",
    "typedConfig": {
        "@type": "type.googleapis.com/envoy.config.filter.network.tcp_proxy.v2.TcpProxy",
        "statPrefix": "BlackHoleCluster",
        "cluster": "BlackHoleCluster"
    }
}
```

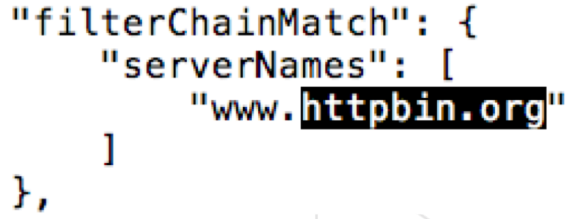
Option 2: Restricted Access with TLS Passthrough in



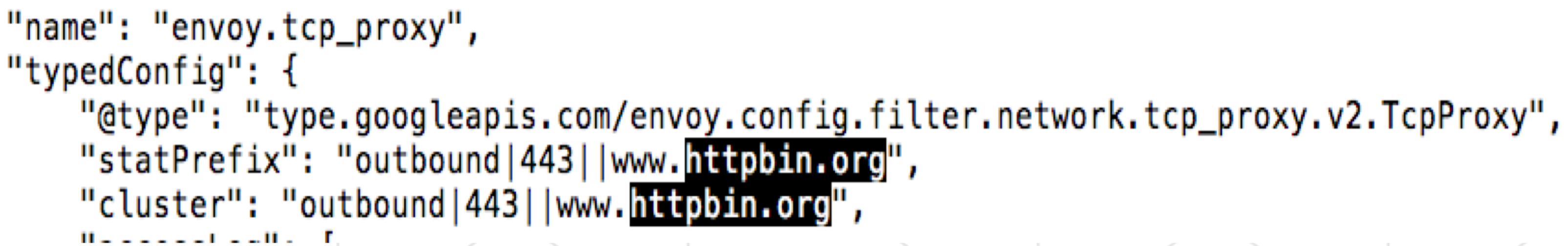
```yaml
apiVersion: networking.istio.io/v1alpha3
kind: ServiceEntry
metadata:
  name: httpbin
spec:
  hosts:
  - www.httpbin.org
  ports:
  - number: 443
    name: https
    protocol: HTTPS
  resolution: DNS
  location: MESH_EXTERNAL
```



```
"name": "0.0.0.0_443",
"address": {
    "socketAddress": {
        "address": "0.0.0.0",
        "portValue": 443
    }
},
```



```
"filterChainMatch": {
    "serverNames": [
        "www.httpbin.org"
    ]
},
```



```
"name": "envoy.tcp_proxy",
"typedConfig": {
    "@type": "type.googleapis.com/envoy.config.filter.network.tcp_proxy.v2.TcpProxy",
    "statPrefix": "outbound|443||www.httpbin.org",
    "cluster": "outbound|443||www.httpbin.org",
    "accessLog": [
```

Option 3: Restricted Access with TLS Origination

PROS

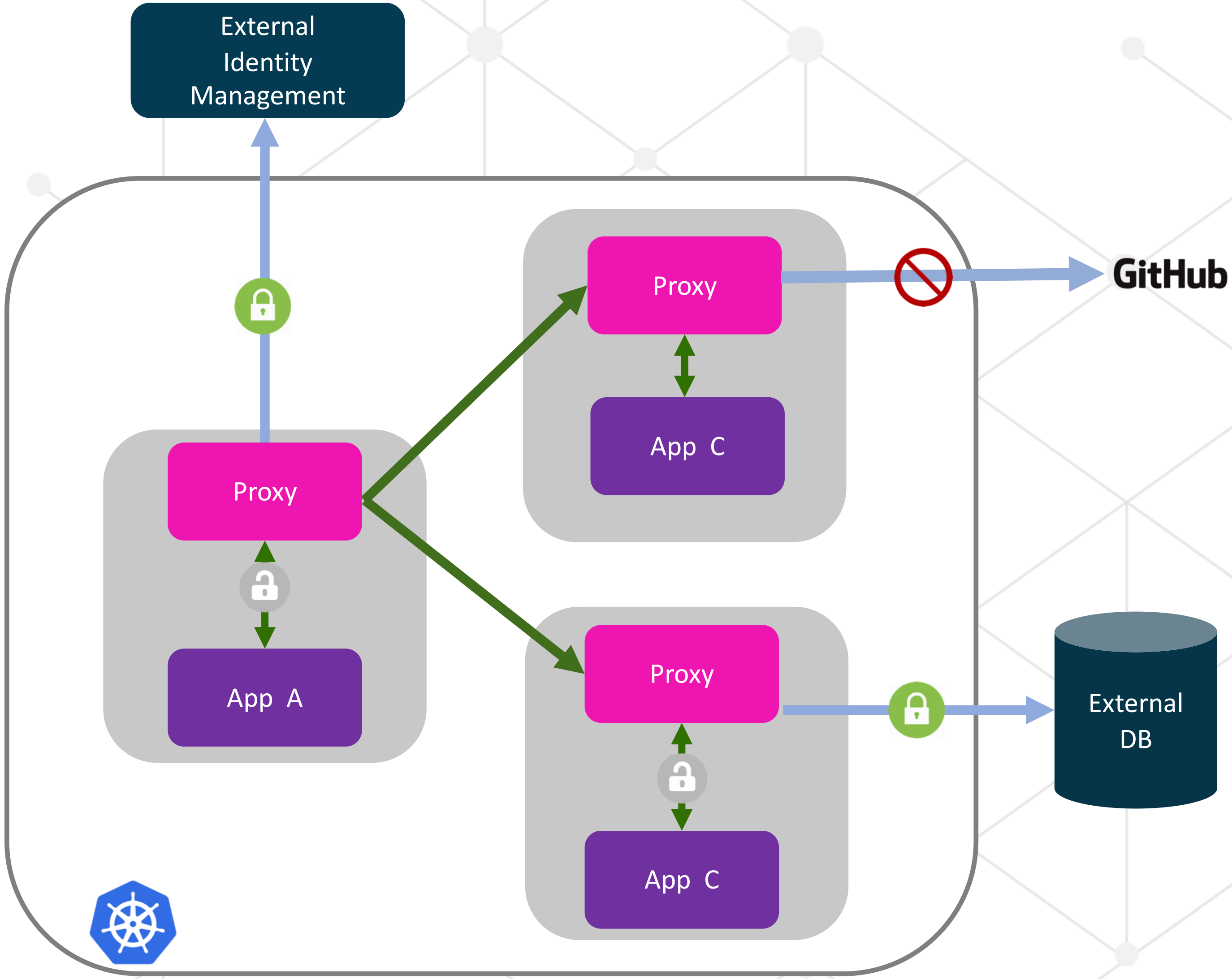
Security:
Secure
L7 policy

Visibility:
HTTP metrics
Access logging
Tracing

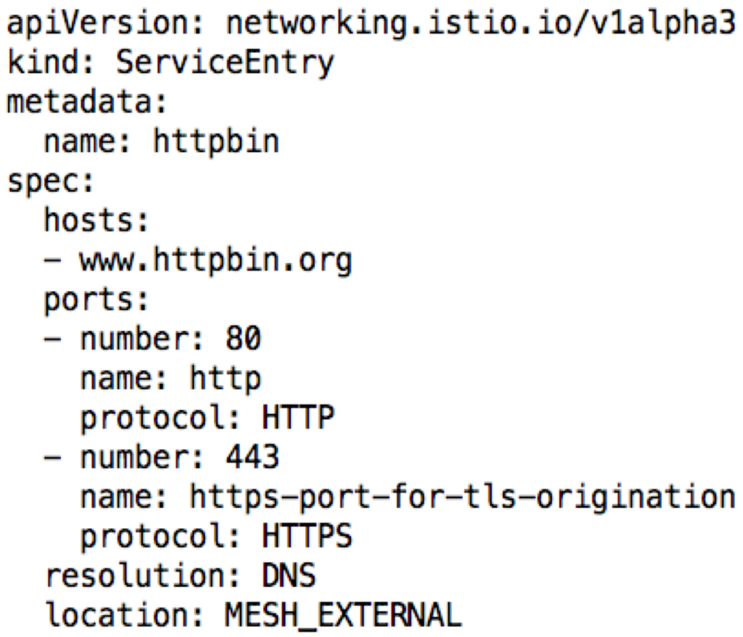
CONS

Configuration:
Lots of configuration

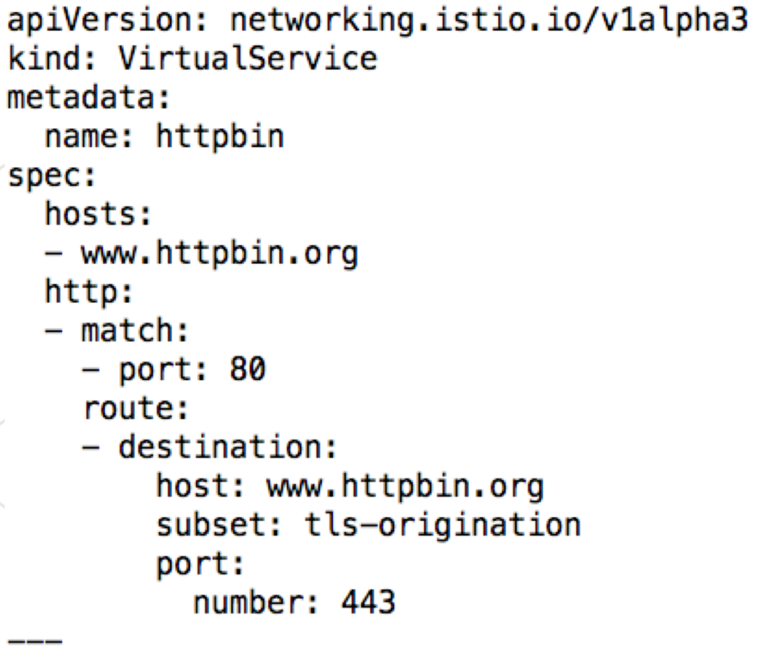
Security:
Unencrypted traffic
b/w Application &
Proxy



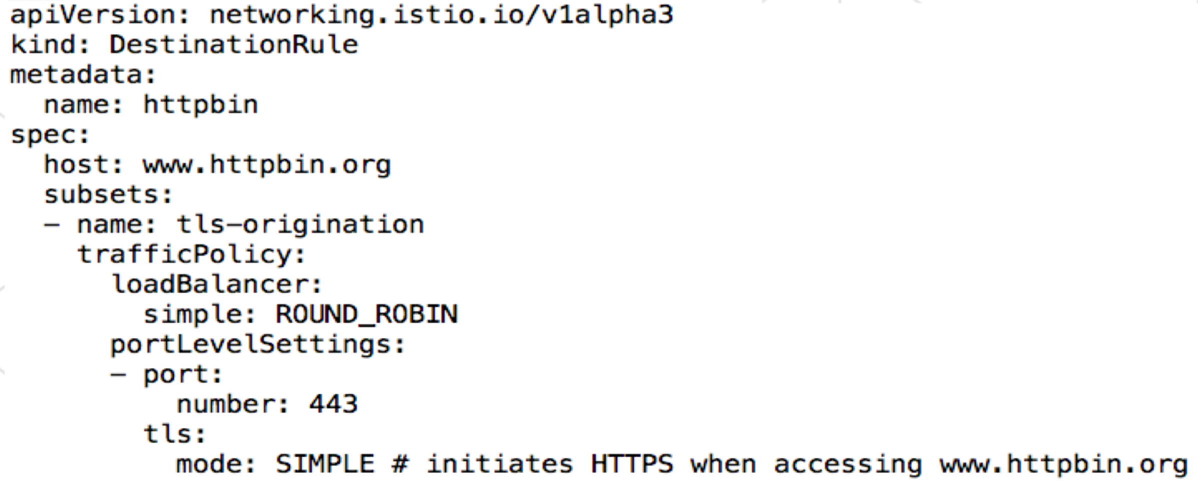
Option 3: Restricted Access with TLS Origination in



```yaml
apiVersion: networking.istio.io/v1alpha3
kind: ServiceEntry
metadata:
  name: httpbin
spec:
  hosts:
  - www.httpbin.org
  ports:
  - number: 80
    name: http
    protocol: HTTP
  - number: 443
    name: https-port-for-tls-origination
    protocol: HTTPS
  resolution: DNS
  location: MESH_EXTERNAL
```

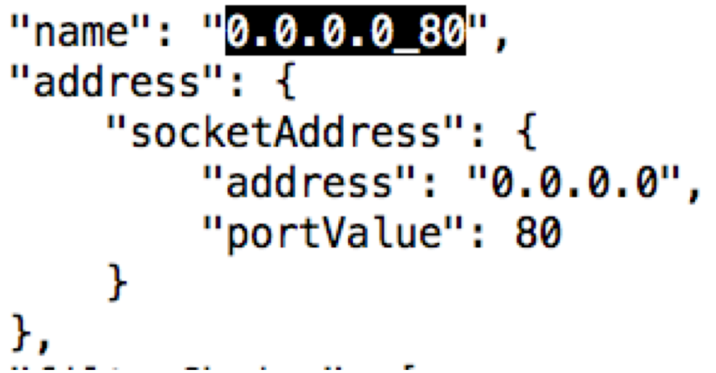


```yaml
apiVersion: networking.istio.io/v1alpha3
kind: VirtualService
metadata:
  name: httpbin
spec:
  hosts:
  - www.httpbin.org
  http:
  - match:
    - port: 80
    route:
    - destination:
        host: www.httpbin.org
        subset: tls-origination
        port:
          number: 443
```

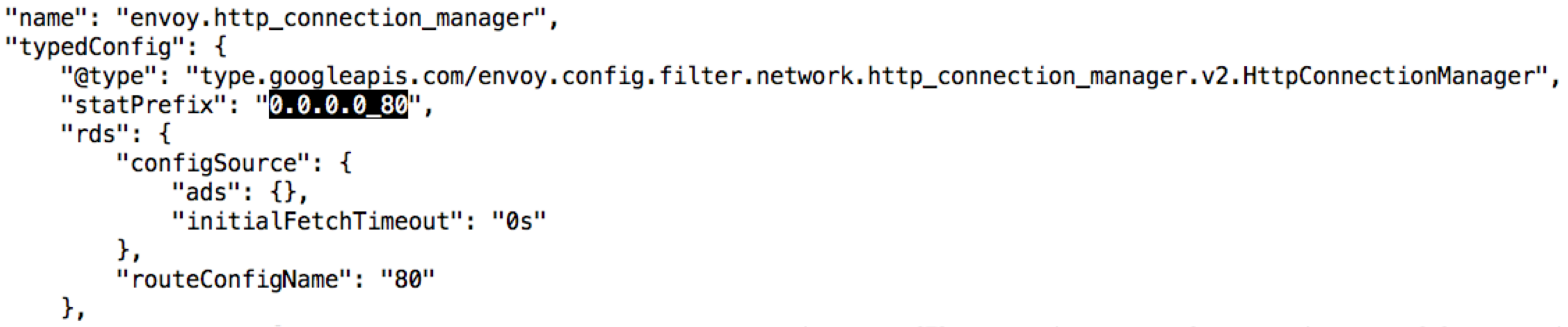


```yaml
---
apiVersion: networking.istio.io/v1alpha3
kind: DestinationRule
metadata:
  name: httpbin
spec:
  host: www.httpbin.org
  subsets:
  - name: tls-origination
    trafficPolicy:
      loadBalancer:
        simple: ROUND_ROBIN
      portLevelSettings:
      - port:
          number: 443
        tls:
          mode: SIMPLE # initiates HTTPS when accessing www.httpbin.org
```

Option 3: Restricted Access with TLS Origination in

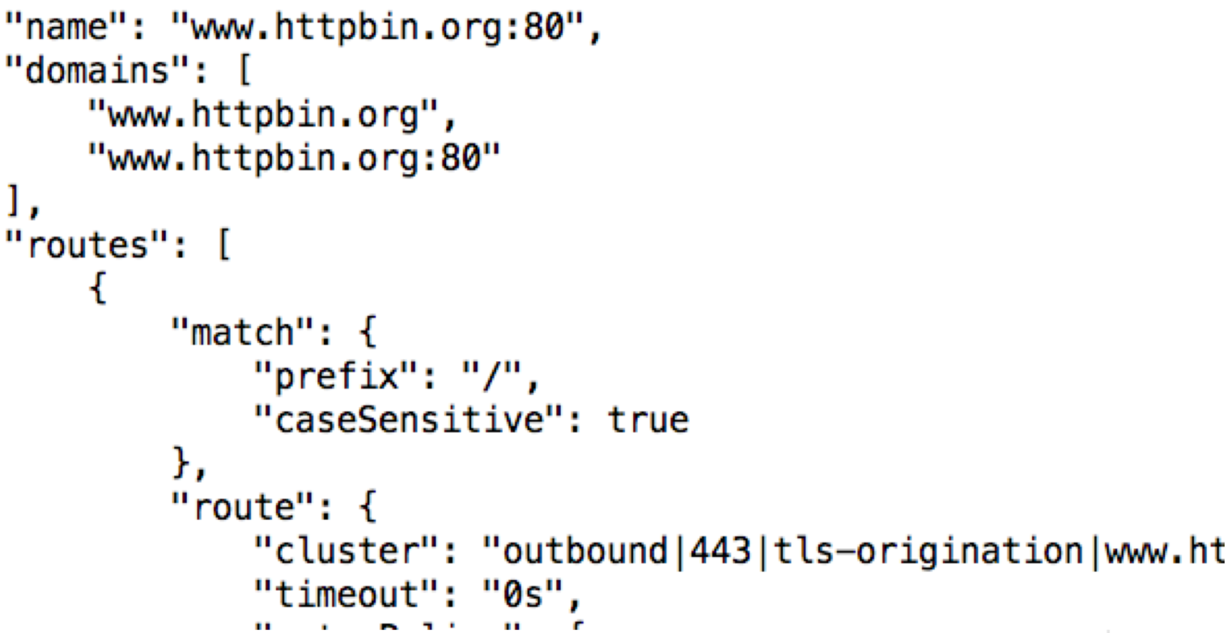


```
"name": "0.0.0.0_80",
"address": {
    "socketAddress": {
        "address": "0.0.0.0",
        "portValue": 80
    }
},
```

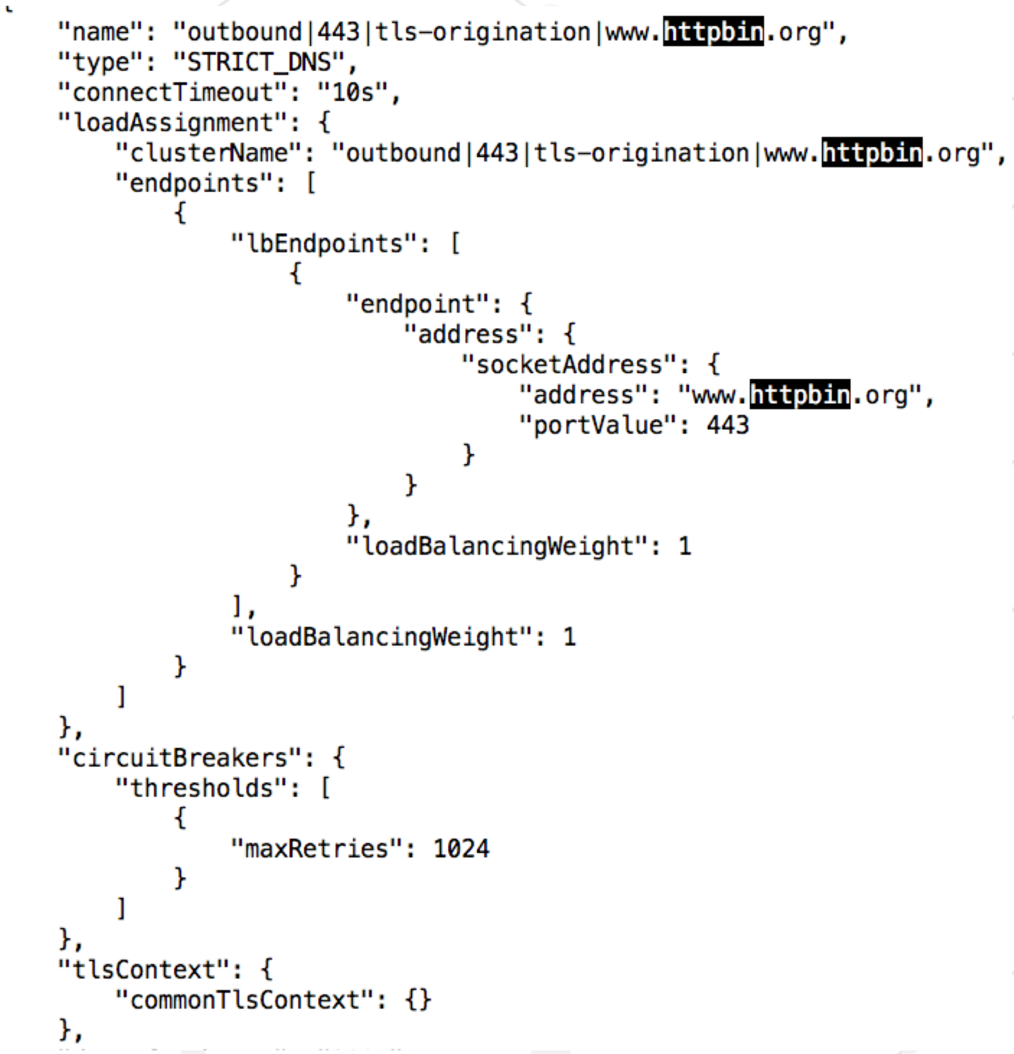


```
"name": "envoy.http_connection_manager",
"typedConfig": {
    "@type": "type.googleapis.com/envoy.config.filter.network.http_connection_manager.v2.HttpConnectionManager",
    "statPrefix": "0.0.0.0_80",
    "rds": {
        "configSource": {
            "ads": {},
            "initialFetchTimeout": "0s"
        },
        "routeConfigName": "80"
    },
```

Option 3: Restricted Access with TLS Origination in



```
"name": "www.httpbin.org:80",
"domains": [
    "www.httpbin.org",
    "www.httpbin.org:80"
],
"routes": [
    {
        "match": {
            "prefix": "/",
            "caseSensitive": true
        },
        "route": {
            "cluster": "outbound|443|tls-origination|www.ht
            "timeout": "0s",
```



```
"name": "outbound|443|tls-origination|www.httpbin.org",
"type": "STRICT_DNS",
"connectTimeout": "10s",
"loadAssignment": {
    "clusterName": "outbound|443|tls-origination|www.httpbin.org",
    "endpoints": [
        {
            "lbEndpoints": [
                {
                    "endpoint": {
                        "address": {
                            "socketAddress": {
                                "address": "www.httpbin.org",
                                "portValue": 443
                            }
                        }
                    },
                    "loadBalancingWeight": 1
                }
            ],
            "loadBalancingWeight": 1
        }
    ]
},
"circuitBreakers": {
    "thresholds": [
        {
            "maxRetries": 1024
        }
    ]
},
"tlsContext": {
    "commonTlsContext": {}
},
```

Option 4: Egress Gateway with TLS Origination

PROS

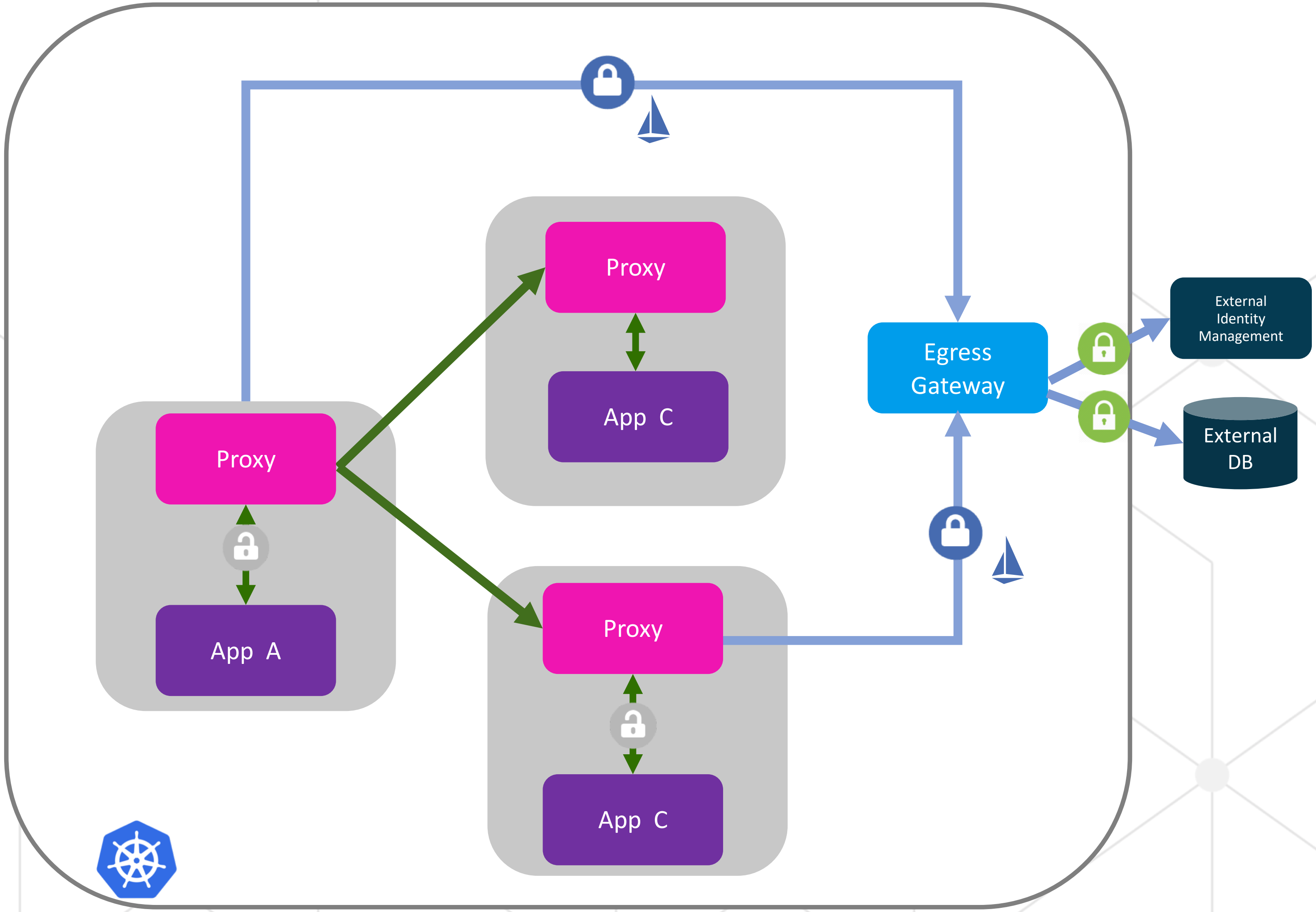
Security:
Secure
L7 policy
Public IP control

Visibility:
HTTP metrics
Access logging
Tracing

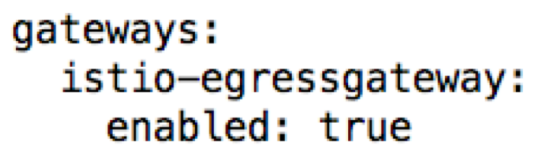
CONS

Configuration:
Most configuration

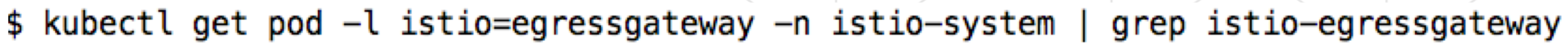
Security:
mTLS required



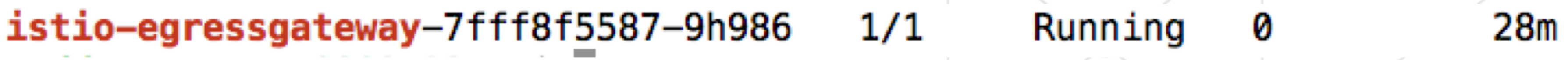
Option 4: Egress Gateway with TLS Origination in



```
gateways:
  istio-egressgateway:
    enabled: true
```

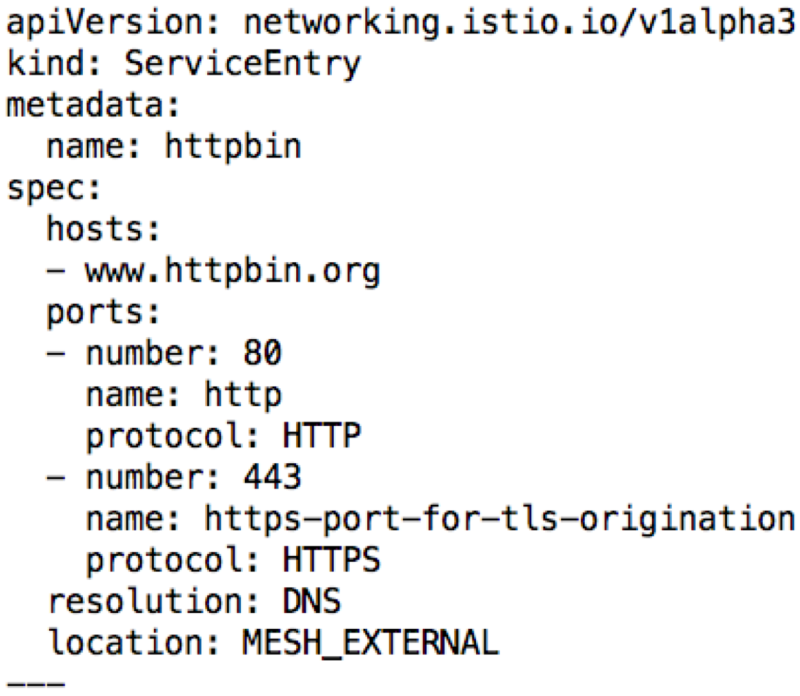


```
$ kubectl get pod -l istio=egressgateway -n istio-system | grep istio-egressgateway
```

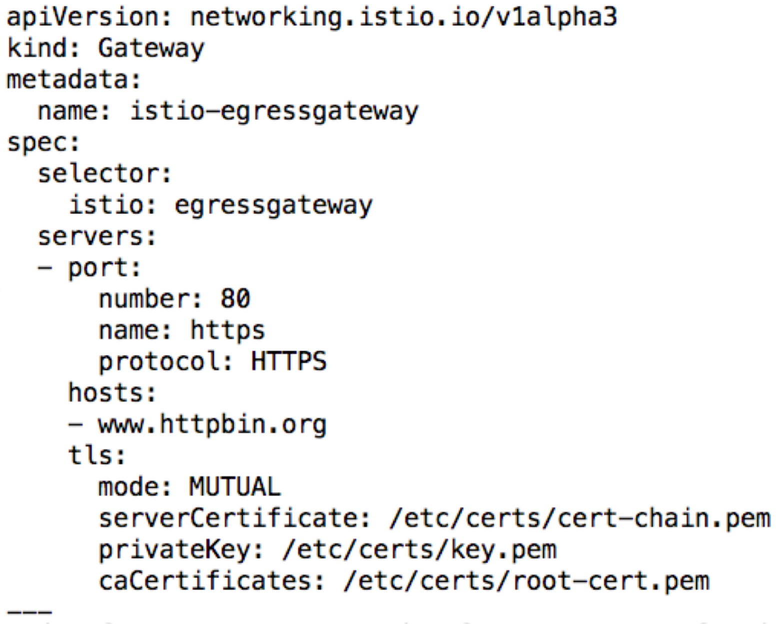


```
istio-egressgateway-7fff8f5587-9h986   1/1     Running   0          28m
```

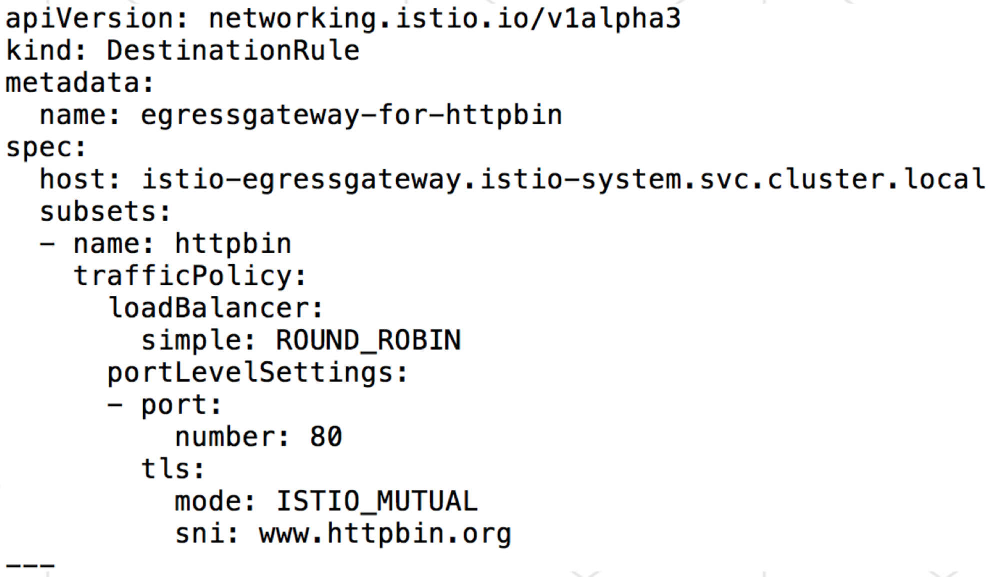
Option 4: Egress Gateway with TLS Origination in



```
apiVersion: networking.istio.io/v1alpha3
kind: ServiceEntry
metadata:
  name: httpbin
spec:
  hosts:
  - www.httpbin.org
  ports:
  - number: 80
    name: http
    protocol: HTTP
  - number: 443
    name: https-port-for-tls-origination
    protocol: HTTPS
  resolution: DNS
  location: MESH_EXTERNAL
---
```

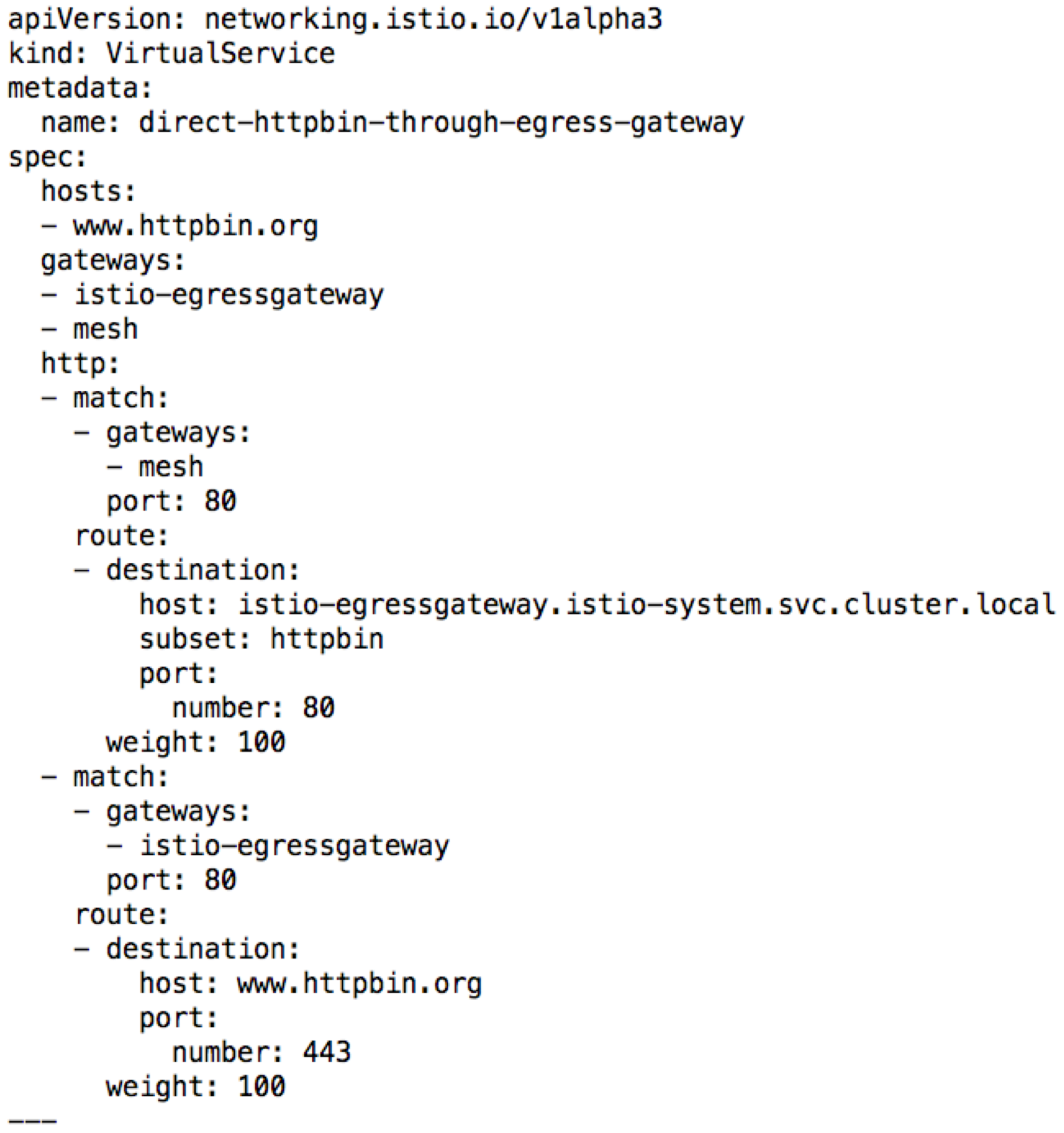


```
apiVersion: networking.istio.io/v1alpha3
kind: Gateway
metadata:
  name: istio-egressgateway
spec:
  selector:
    istio: egressgateway
  servers:
  - port:
      number: 80
      name: https
      protocol: HTTPS
    hosts:
    - www.httpbin.org
    tls:
      mode: MUTUAL
      serverCertificate: /etc/certs/cert-chain.pem
      privateKey: /etc/certs/key.pem
      caCertificates: /etc/certs/root-cert.pem
---
```

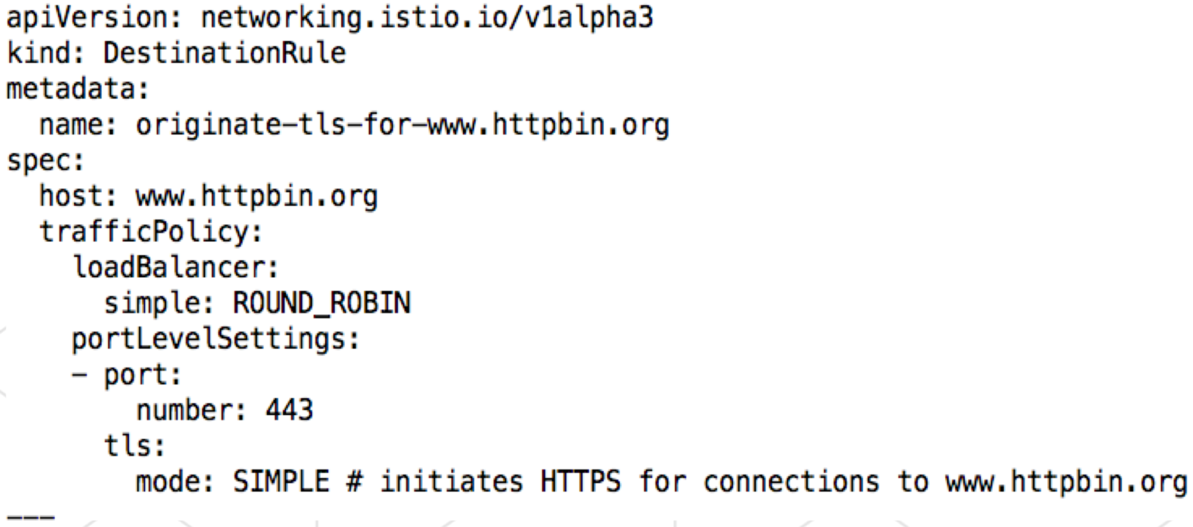


```
apiVersion: networking.istio.io/v1alpha3
kind: DestinationRule
metadata:
  name: egressgateway-for-httpbin
spec:
  host: istio-egressgateway.istio-system.svc.cluster.local
  subsets:
  - name: httpbin
    trafficPolicy:
      loadBalancer:
        simple: ROUND_ROBIN
      portLevelSettings:
      - port:
          number: 80
        tls:
          mode: ISTIO_MUTUAL
          sni: www.httpbin.org
---
```

Option 4: Egress Gateway with TLS Origination in



```
apiVersion: networking.istio.io/v1alpha3
kind: VirtualService
metadata:
  name: direct-httpbin-through-egress-gateway
spec:
  hosts:
  - www.httpbin.org
  gateways:
  - istio-egressgateway
  - mesh
  http:
  - match:
    - gateways:
      - mesh
      port: 80
    route:
    - destination:
        host: istio-egressgateway.istio-system.svc.cluster.local
        subset: httpbin
        port:
          number: 80
      weight: 100
  - match:
    - gateways:
      - istio-egressgateway
      port: 80
    route:
    - destination:
        host: www.httpbin.org
        port:
          number: 443
      weight: 100
---
```



```
apiVersion: networking.istio.io/v1alpha3
kind: DestinationRule
metadata:
  name: originate-tls-for-www.httpbin.org
spec:
  host: www.httpbin.org
  trafficPolicy:
    loadBalancer:
      simple: ROUND_ROBIN
    portLevelSettings:
    - port:
        number: 443
      tls:
        mode: SIMPLE # initiates HTTPS for connections to www.httpbin.org
---
```

Summary

Most Visibility

Egress Gateway

MORDAC, THE PREVENTER OF INFORMATION SERVICES.
SECURITY IS MORE IMPORTANT THAN USABILITY.
scottadams@aol.com
www.dilbert.com
IN A PERFECT WORLD, NO ONE WOULD BE ABLE TO USE ANYTHING.
©2007 Scott Adams, Inc./Dist. by UFS, Inc.
To complete the log-in procedure, stare directly at the sun.

Least Visibility

The Road to Secure External Services with

Allow Any

Capture Destination IP

Service Entry for External Services

Update Application to use HTTP

Use TLS Origination Capture HTTP metrics

Secure External Services

Questions?

Neeraj Poddar
@nrjpoddar
✉ neeraj@aspenmesh.io