Tracing KubeVirt traffic with Istio

Radim Hrazdil

rhrazdil@redhat.com

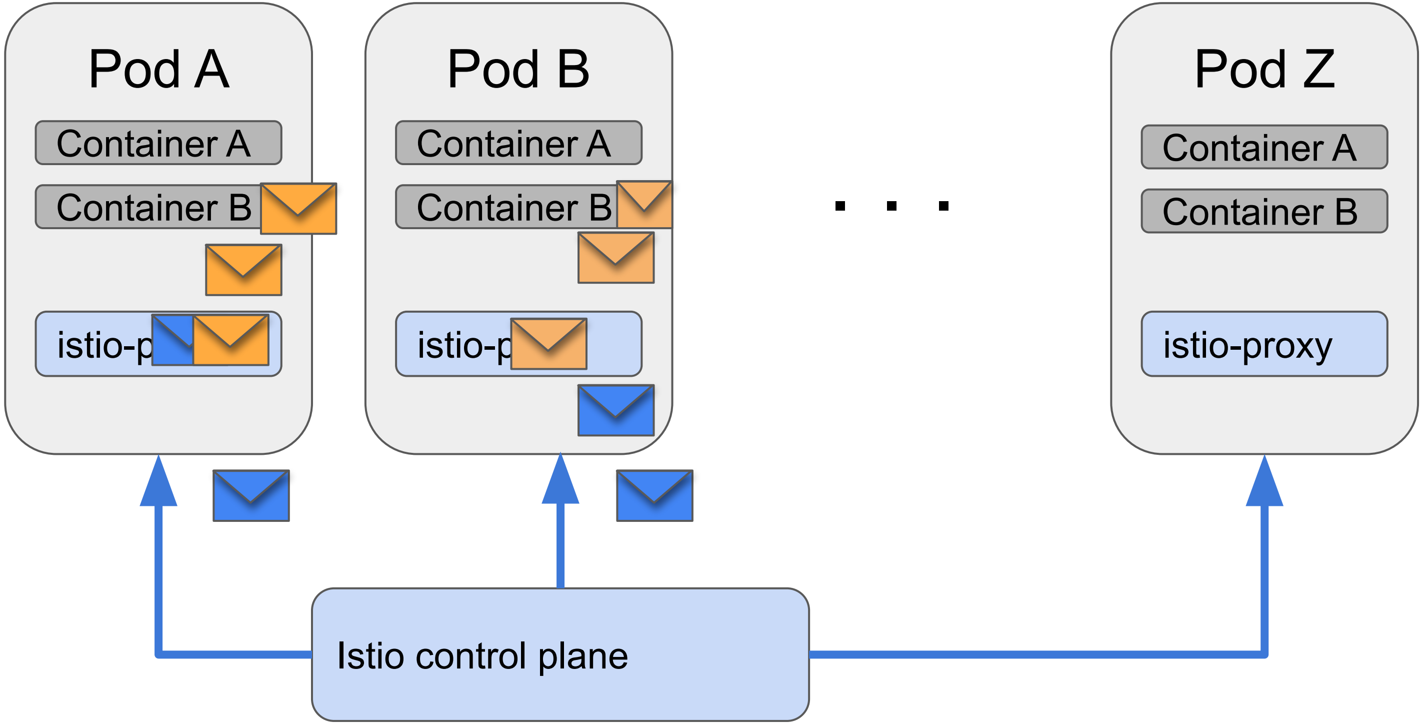
Agenda

1. Istio Service Mesh
2. KubeVirt
3. KubeVirt in-pod networking
4. Istio proxying
5. Adjusting NAT rules
6. Demo

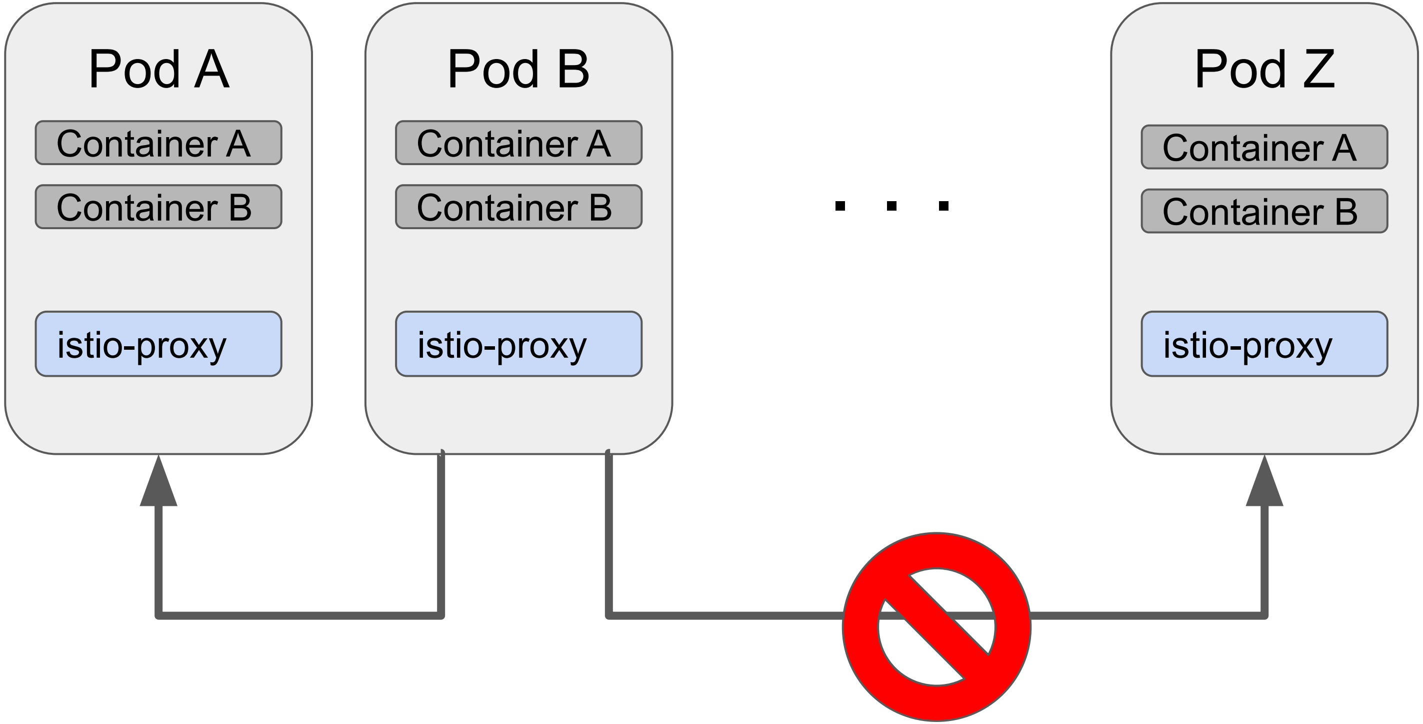
Istio Service Mesh

- Service Mesh is a layer of infrastructure transparent to the applications
- Istio is a service mesh distribution built on Envoy proxy data plane
- Provides additional features
    - Traffic management
    - Traffic monitoring
    - Load balancing
    - Security
    - Rate limiting
- These additional features may be vendored by other applications

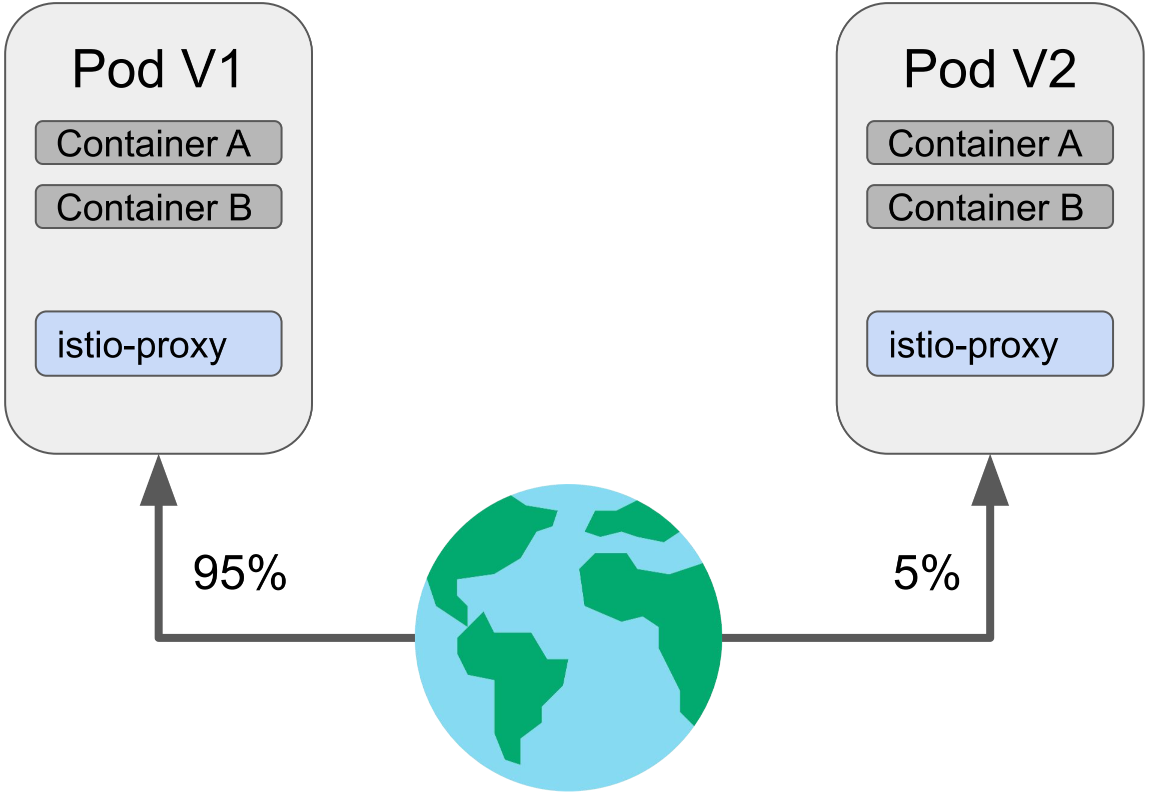
Istio - components



Istio - traffic management



Istio - traffic management



Pod V1

Container A

Container B

istio-proxy

Pod V2

Container A

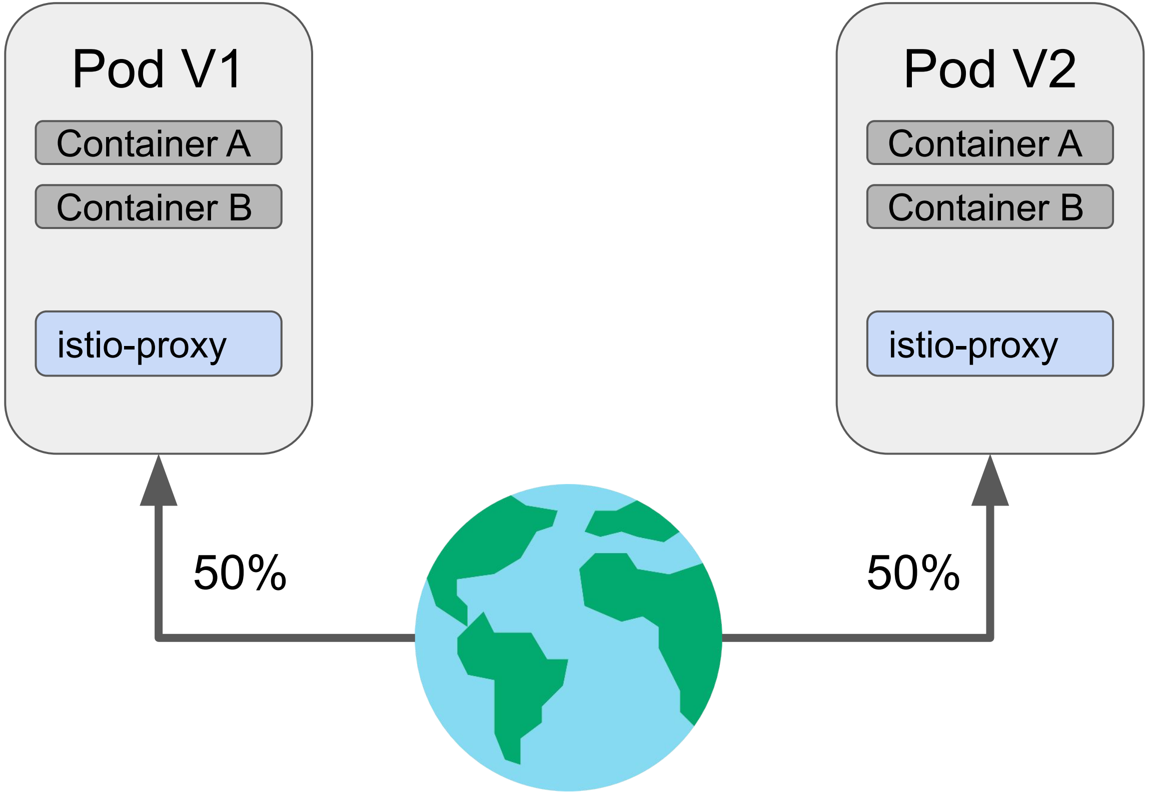
Container B

istio-proxy

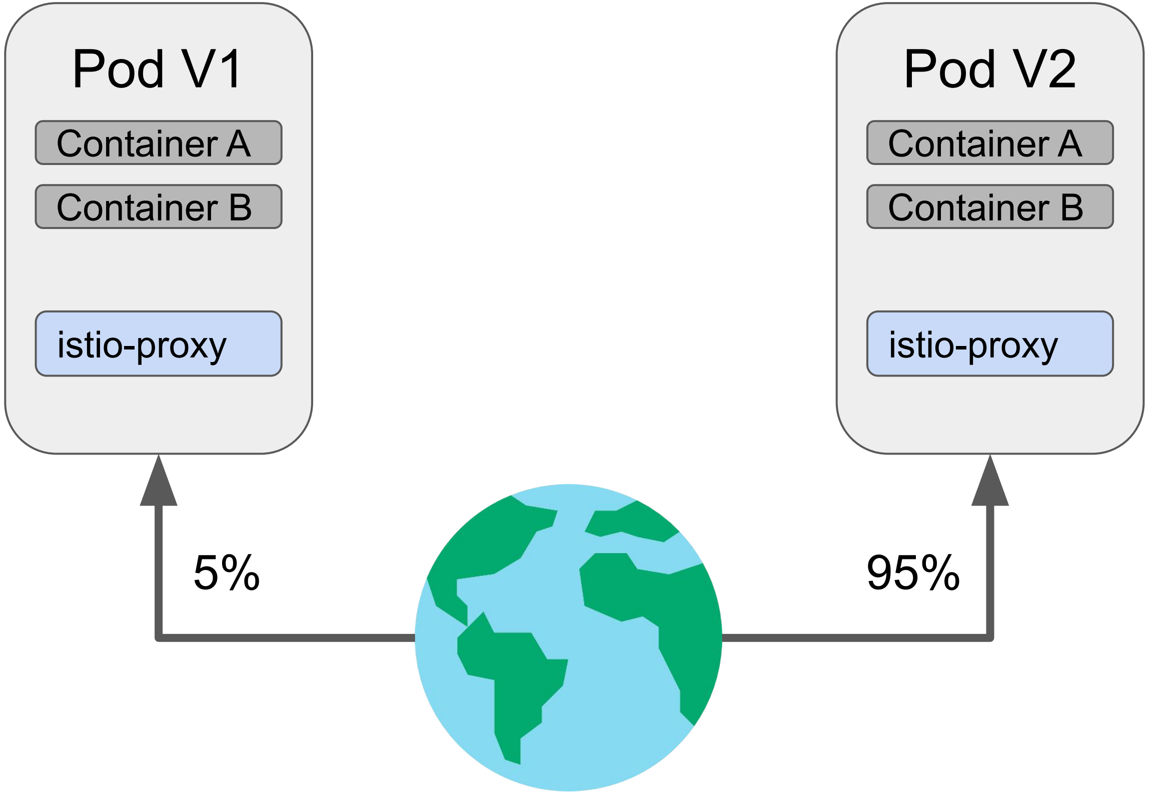
95%

5%

Istio - traffic management



Istio - traffic management

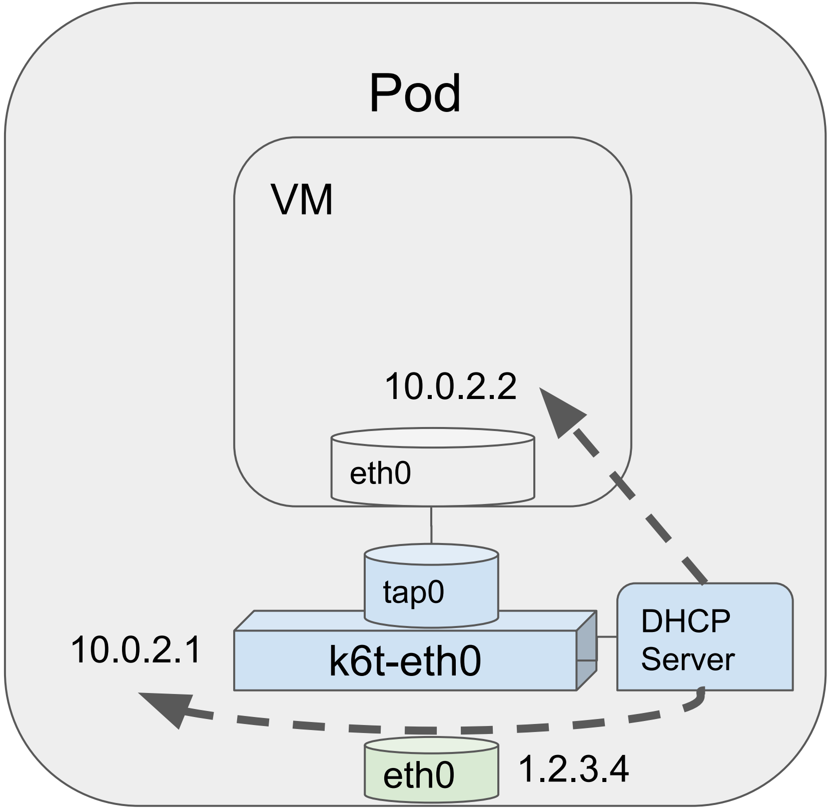


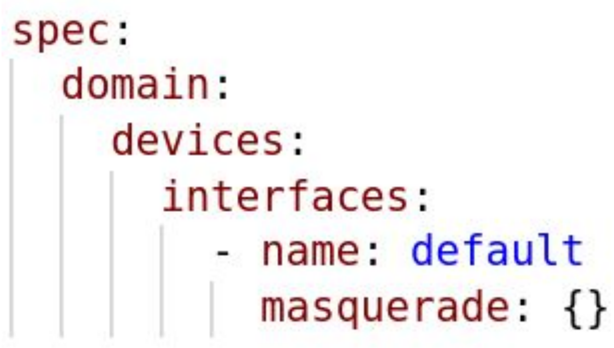
KubeVirt

- KubeVirt is a kubernetes plugin that allows running traditional Virtual Machines side by side with other Kubernetes containerized workloads

- Provides common roof for running containerized and legacy applications running in Virtual Machines

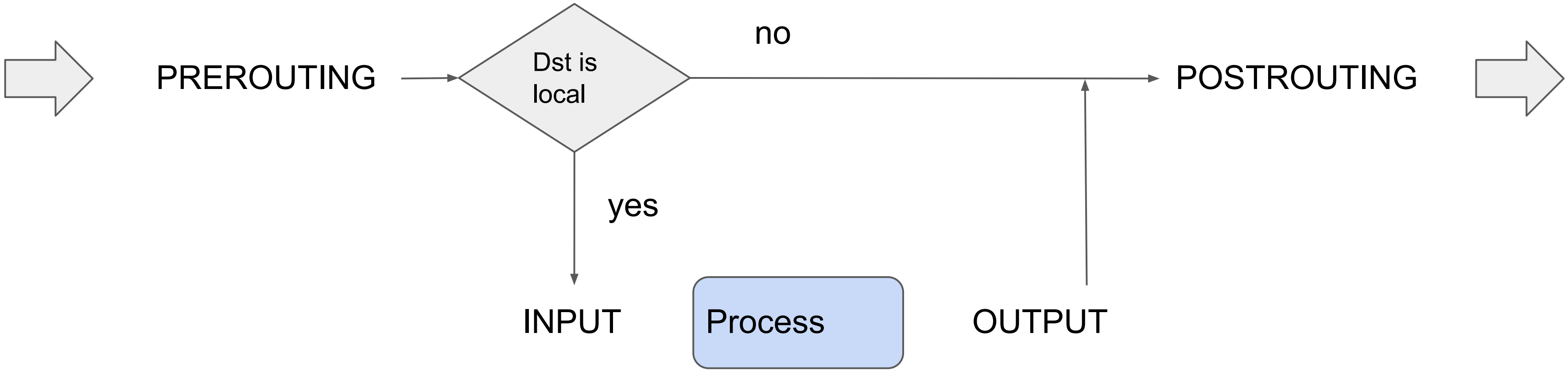
KubeVirt masquerade in-pod networking



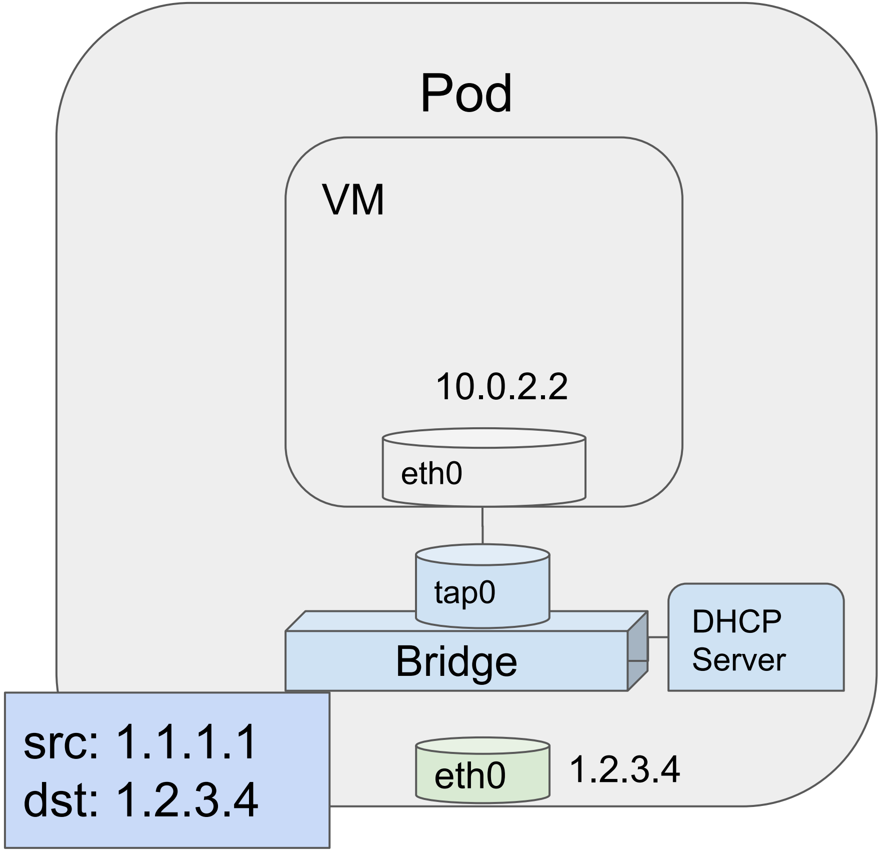


```
spec:
  domain:
    devices:
      interfaces:
        - name: default
          masquerade: {}
```

Netfilter - NAT table chains



KubeVirt - inbound traffic

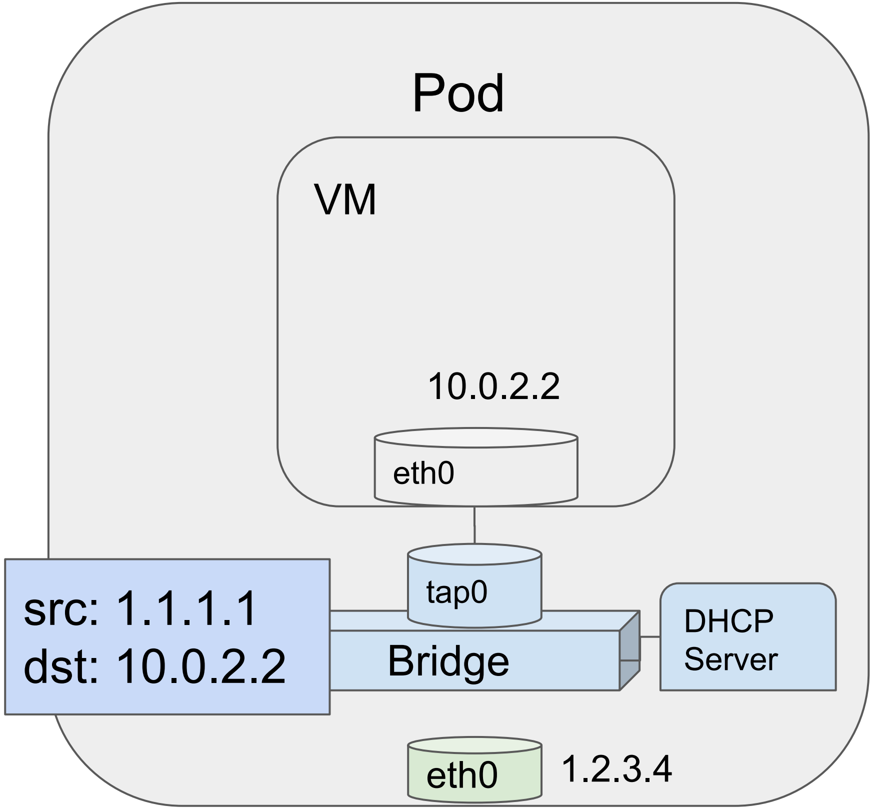


PREROUTING

iifname "eth0" DNAT to 10.0.2.2

. . .

KubeVirt - inbound traffic



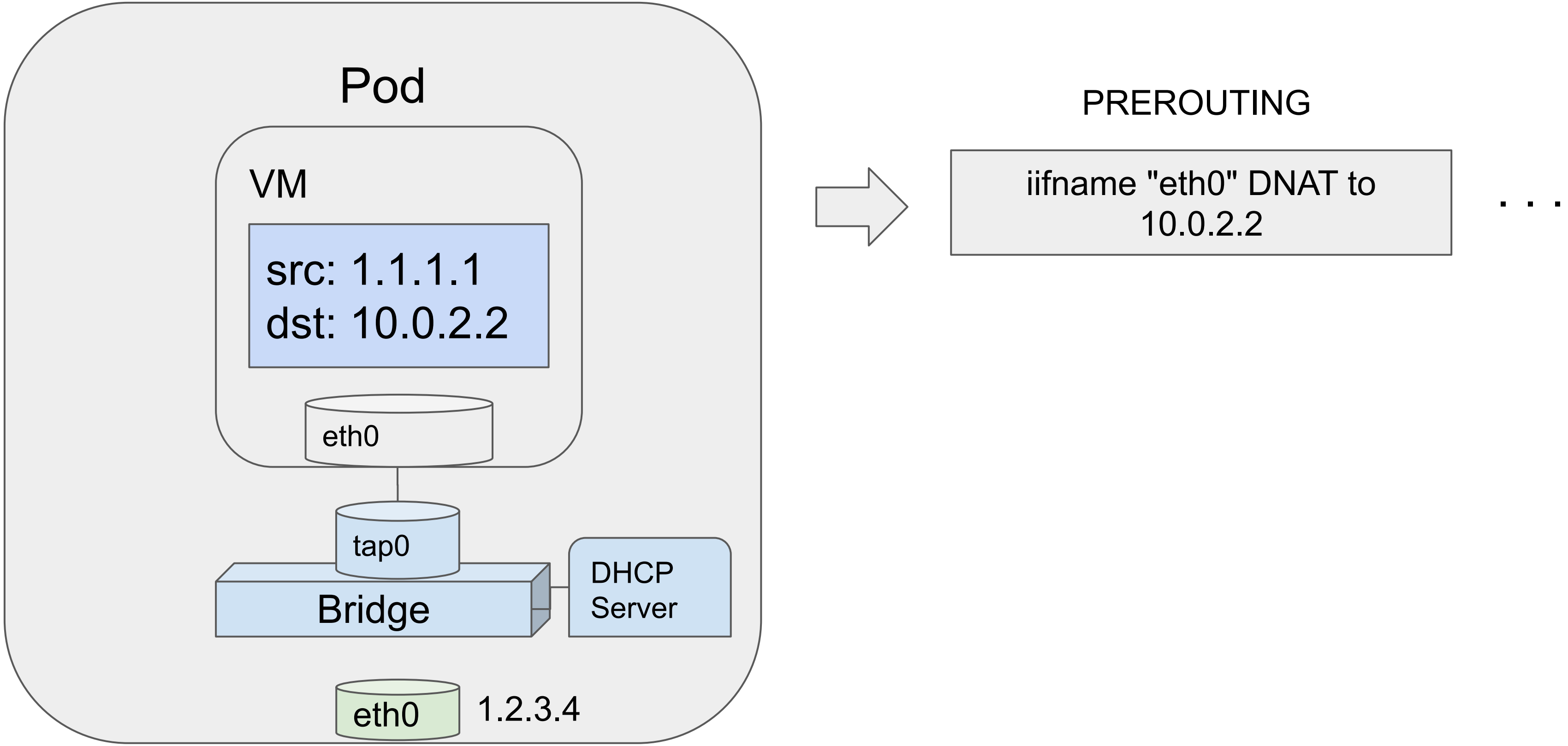
src: 1.1.1.1
dst: 10.0.2.2

PREROUTING

iifname "eth0" DNAT to 10.0.2.2

. . .

KubeVirt - inbound traffic



Pod

VM

src: 1.1.1.1
dst: 10.0.2.2

eth0

tap0

Bridge

DHCP Server

eth0

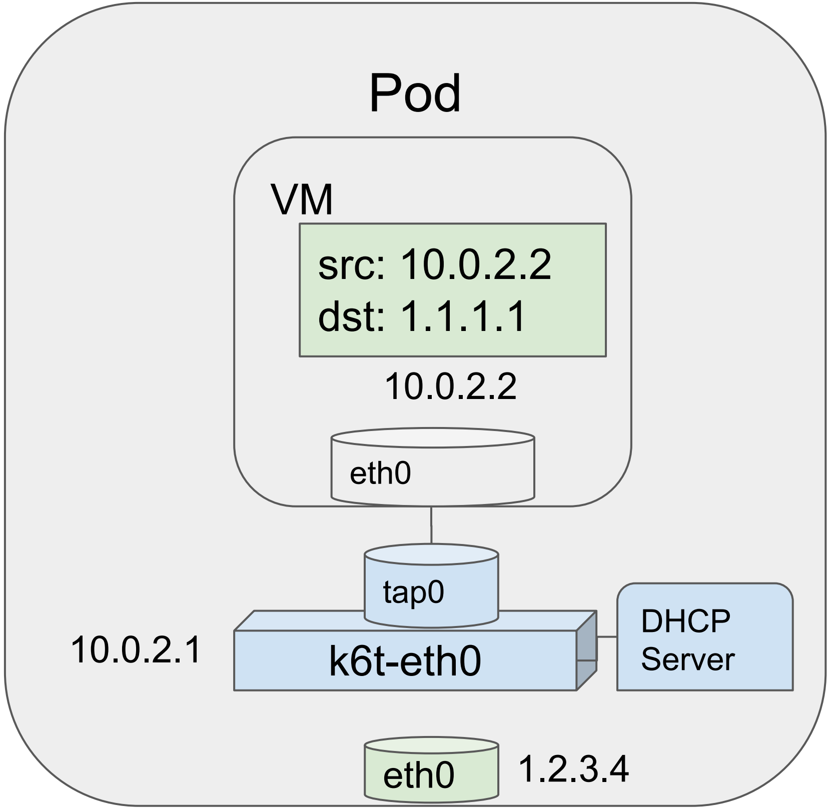
1.2.3.4

PREROUTING

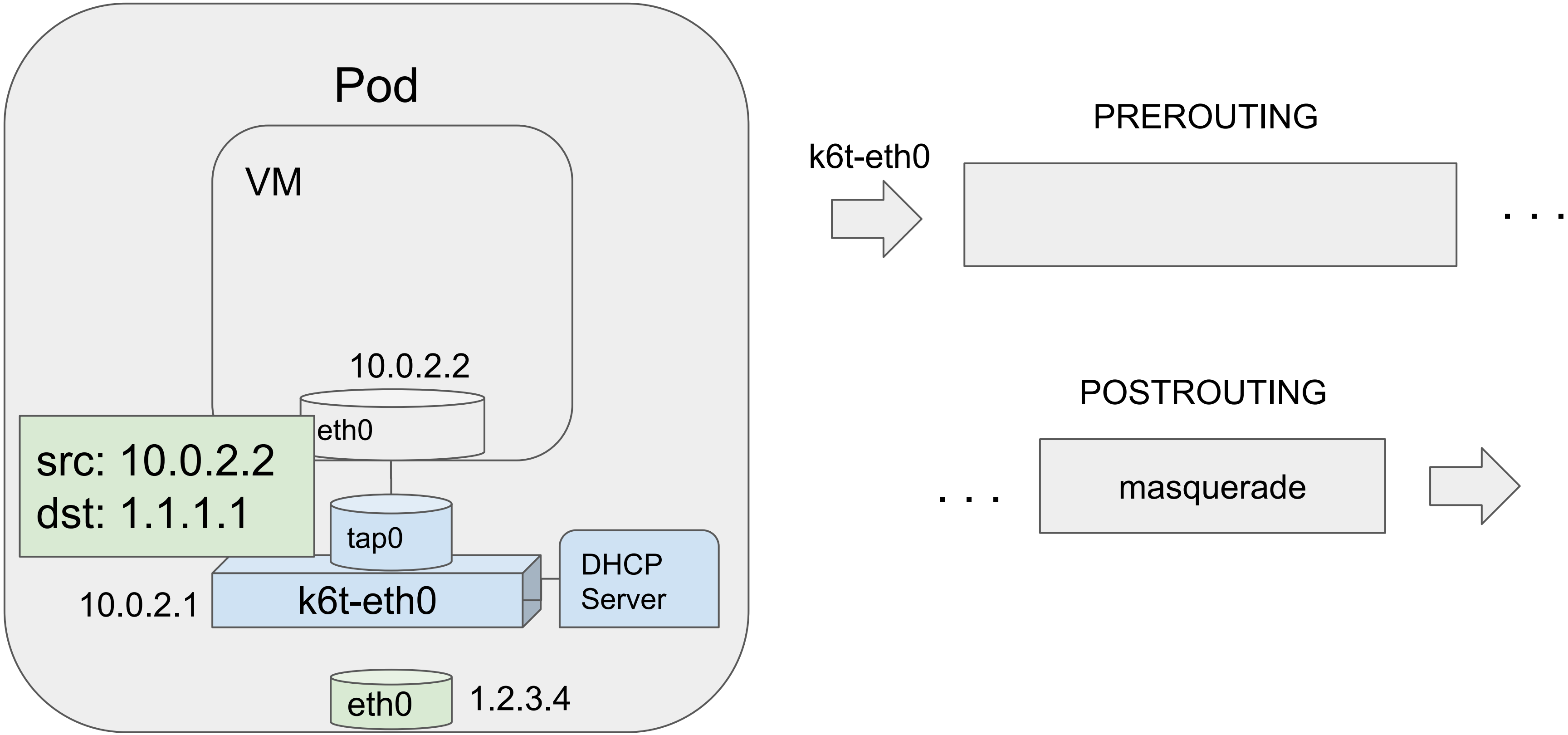
iifname "eth0" DNAT to 10.0.2.2

. . .

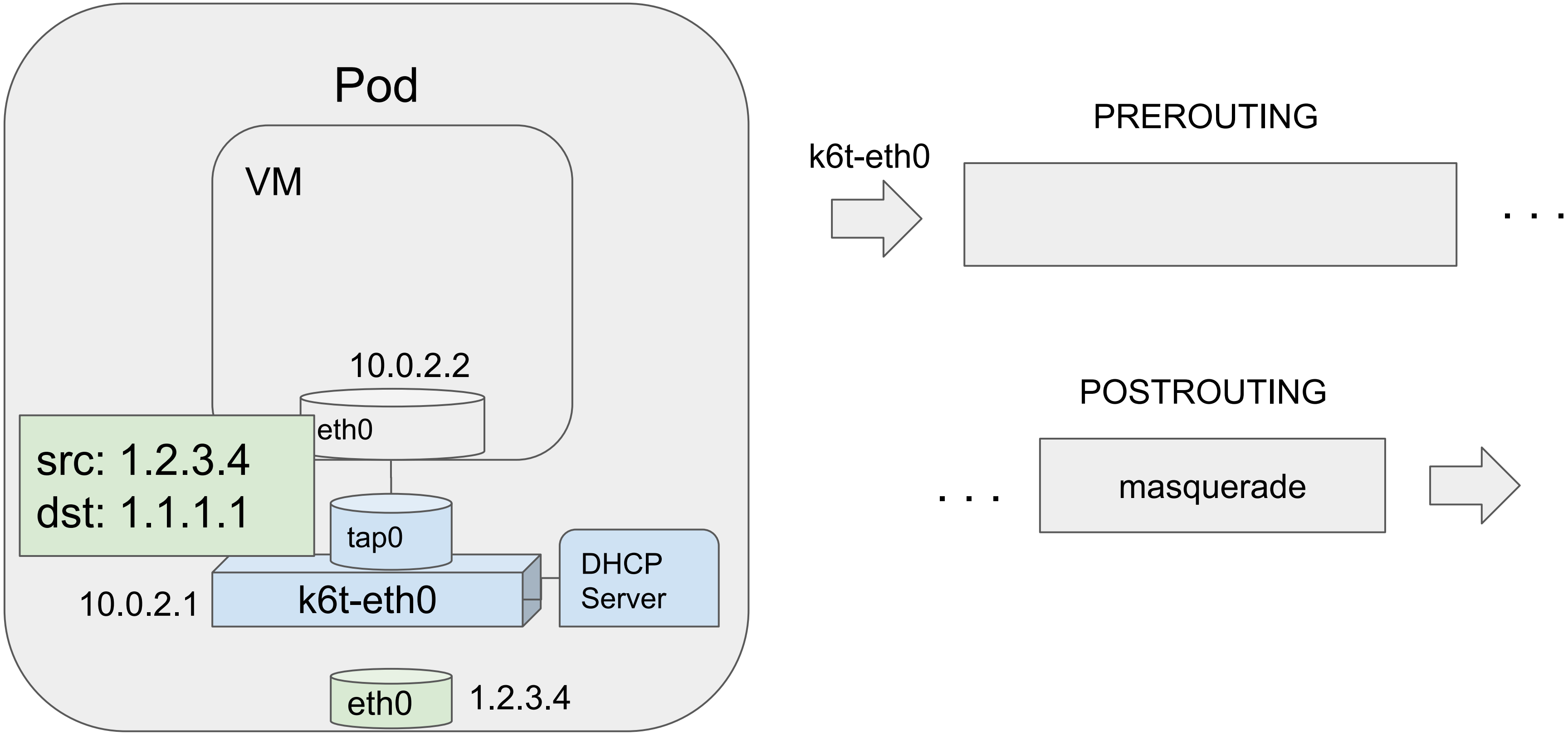
KubeVirt - outbound traffic



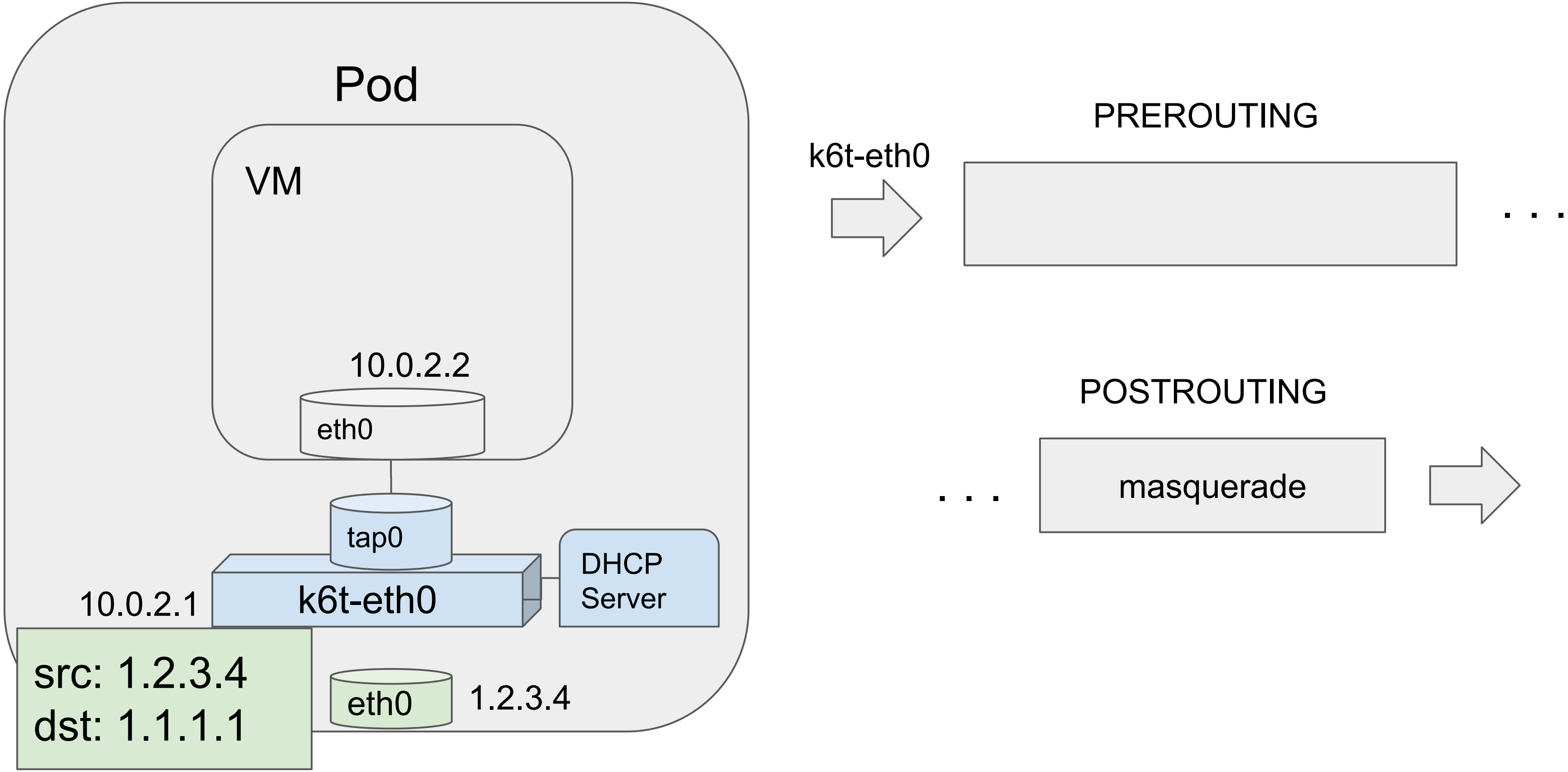
KubeVirt - outbound traffic



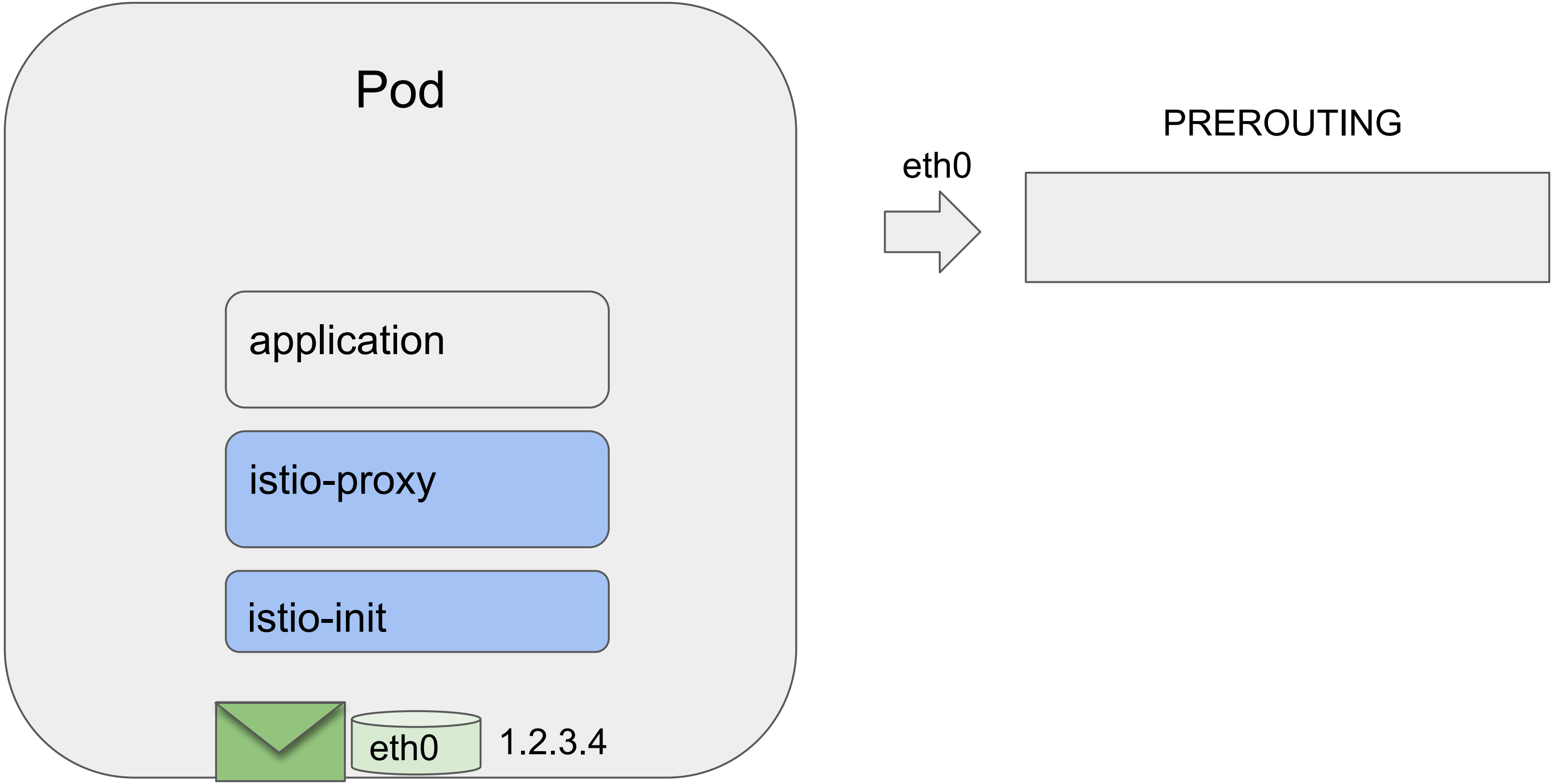
KubeVirt - outbound traffic



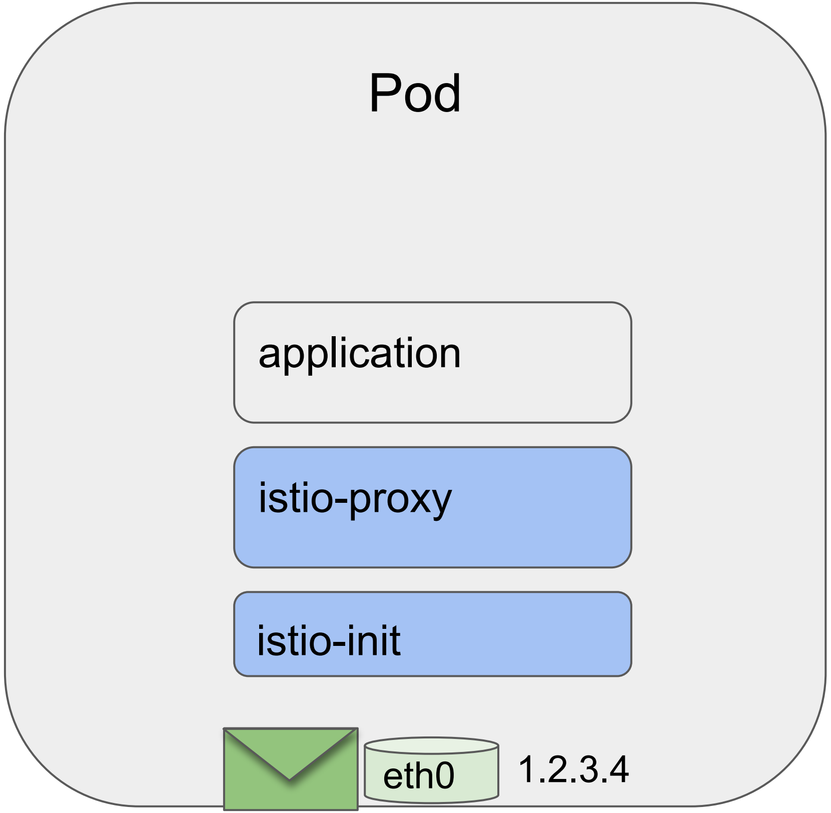
KubeVirt - outbound traffic



Istio - Envoy proxying



Istio - Envoy proxying

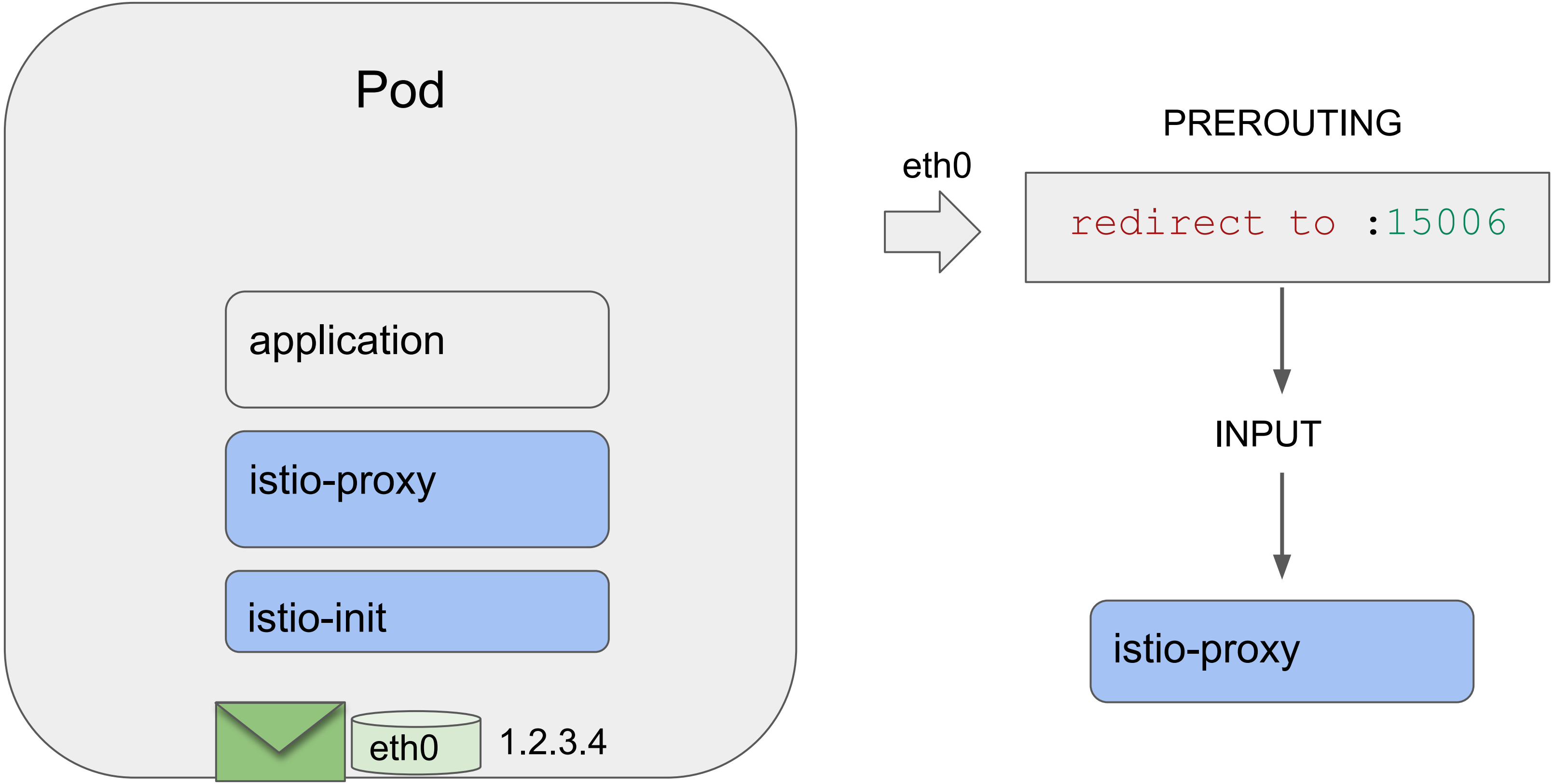


eth0

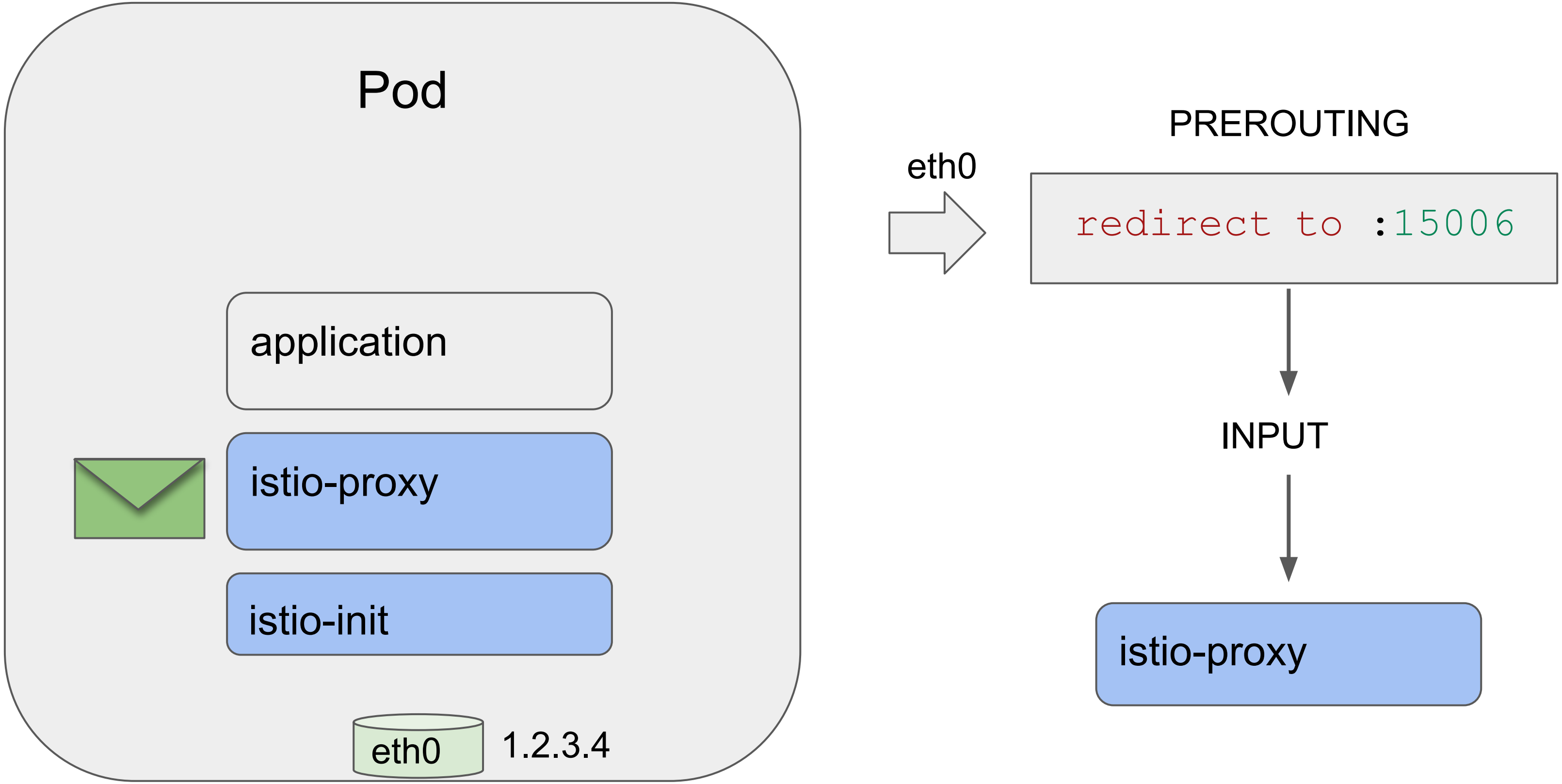
PREROUTING

```
redirect to :15006
```

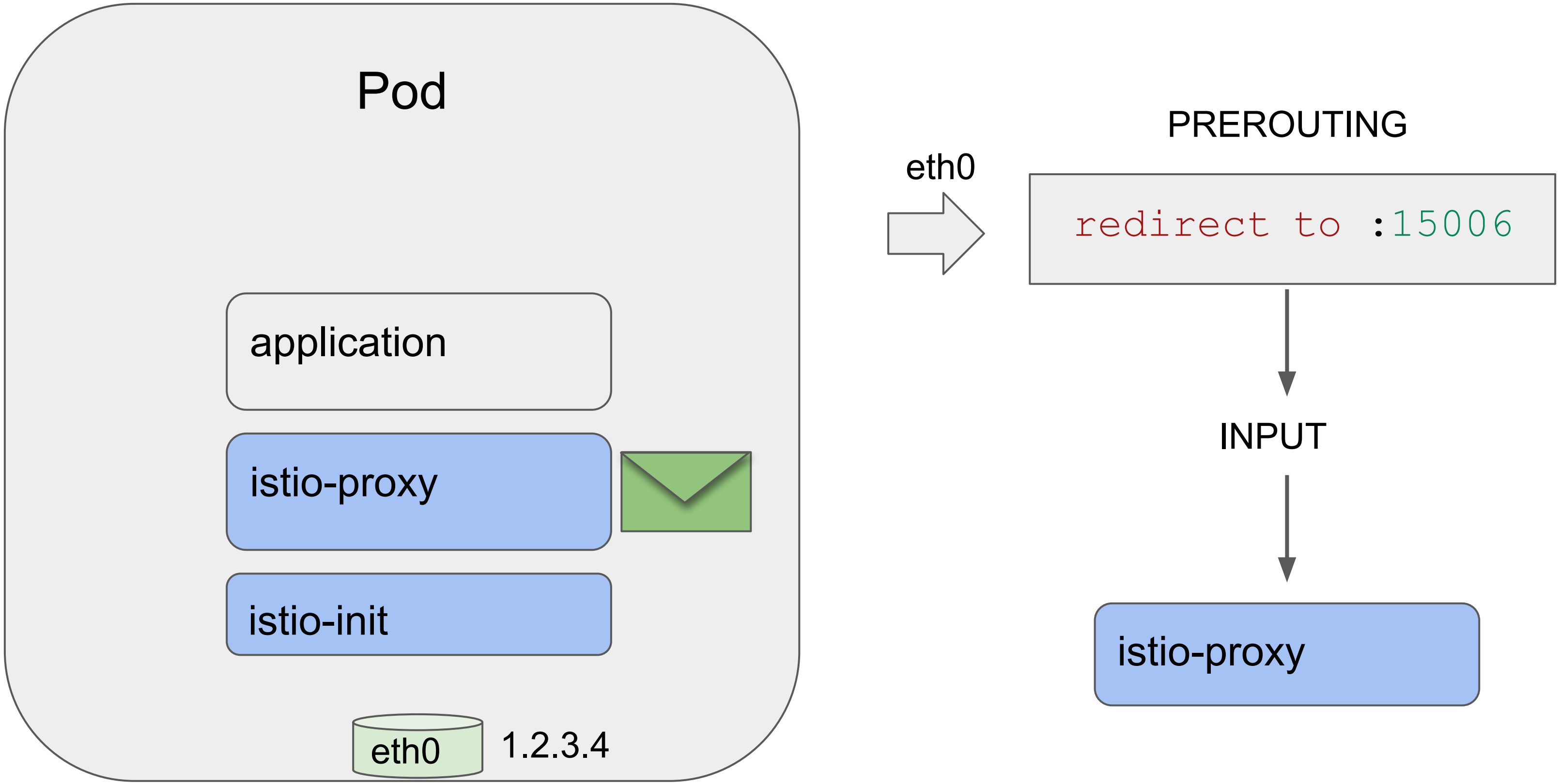
Istio - Envoy proxying



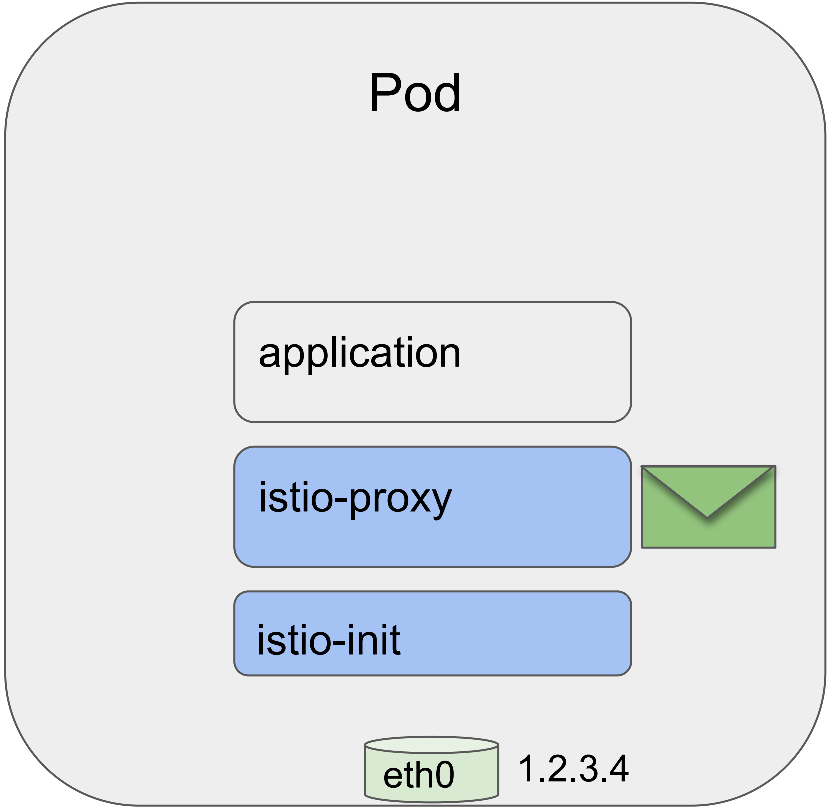
Istio - Envoy proxying



Istio - Envoy proxying

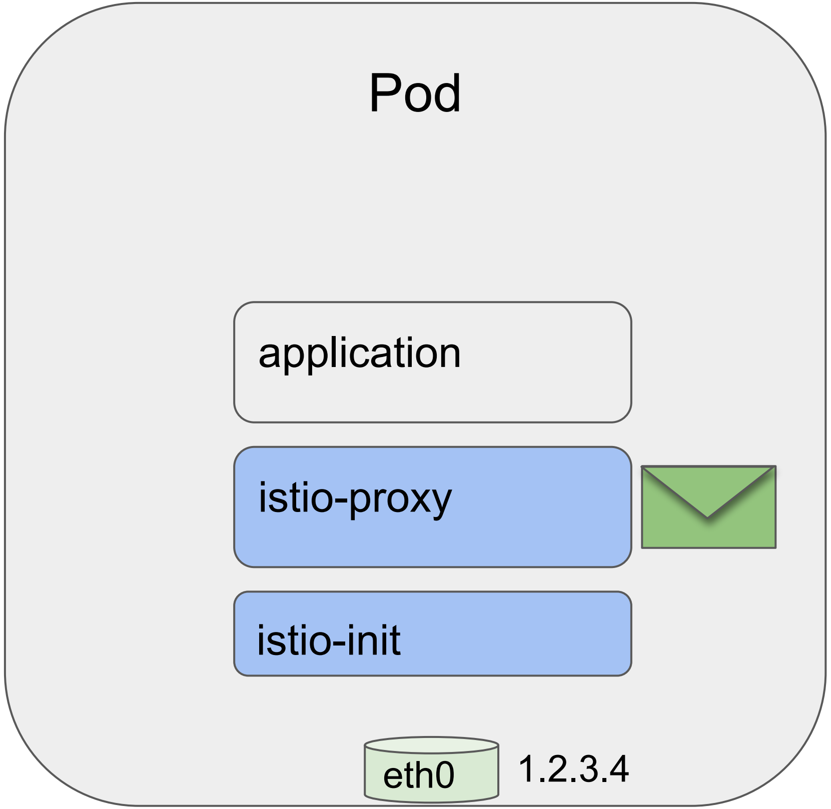


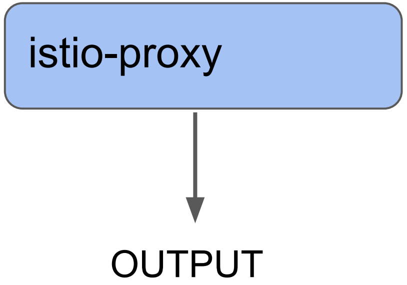
Istio - Envoy proxying



SO_ORIGINAL_DST

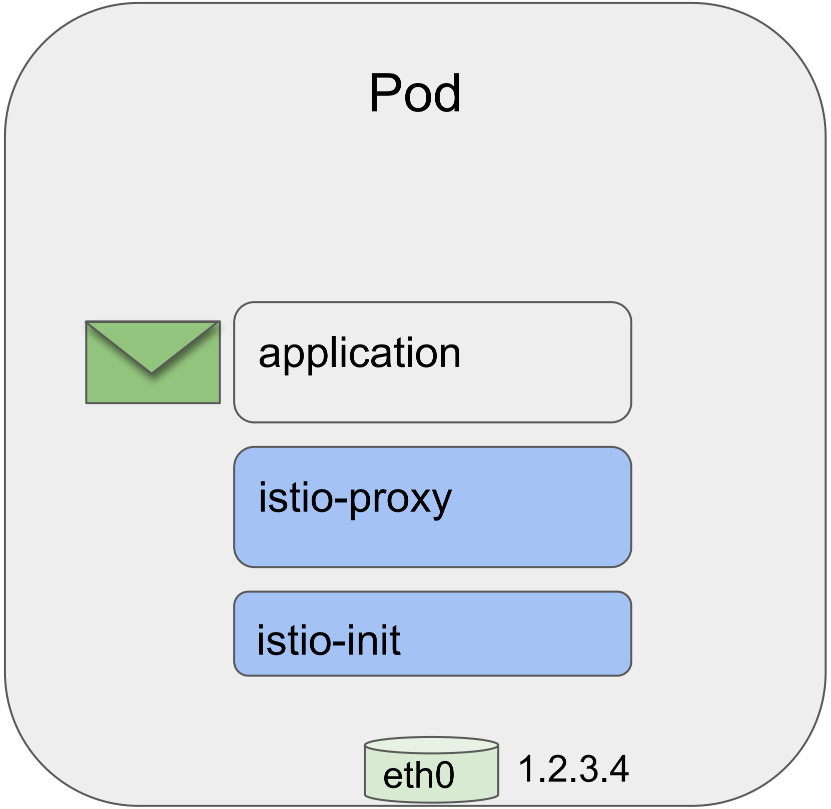
Istio - Envoy proxying

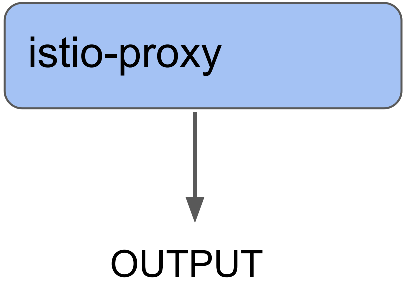




```
chain OUTPUT {
    . . .
    skuid 1337 return
    . . .
    jump ISTIO_REDIRECT
}
chain ISTIO_REDIRECT {
    meta l4proto tcp redirect to :15001
}
```

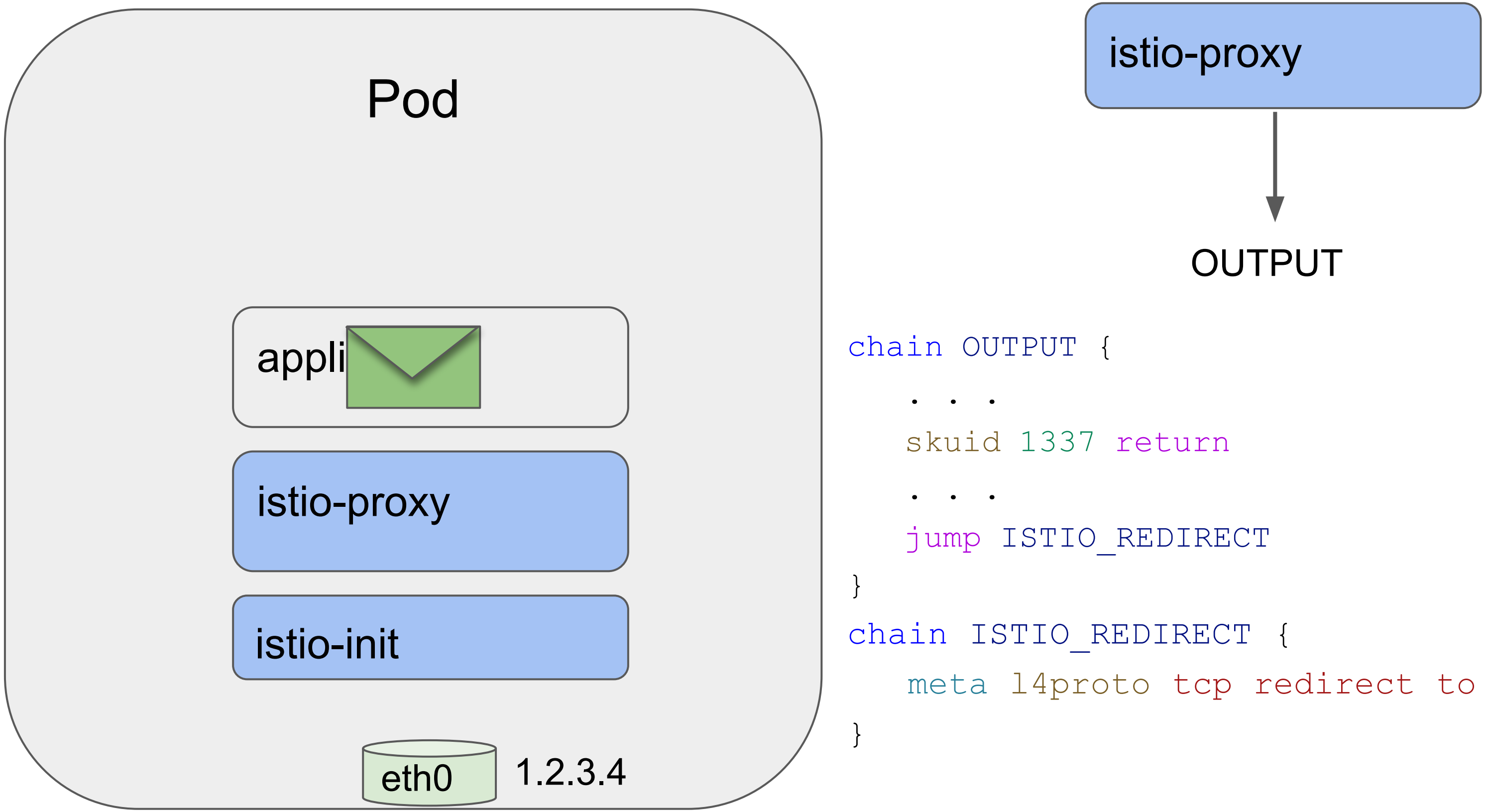
Istio - Envoy proxying





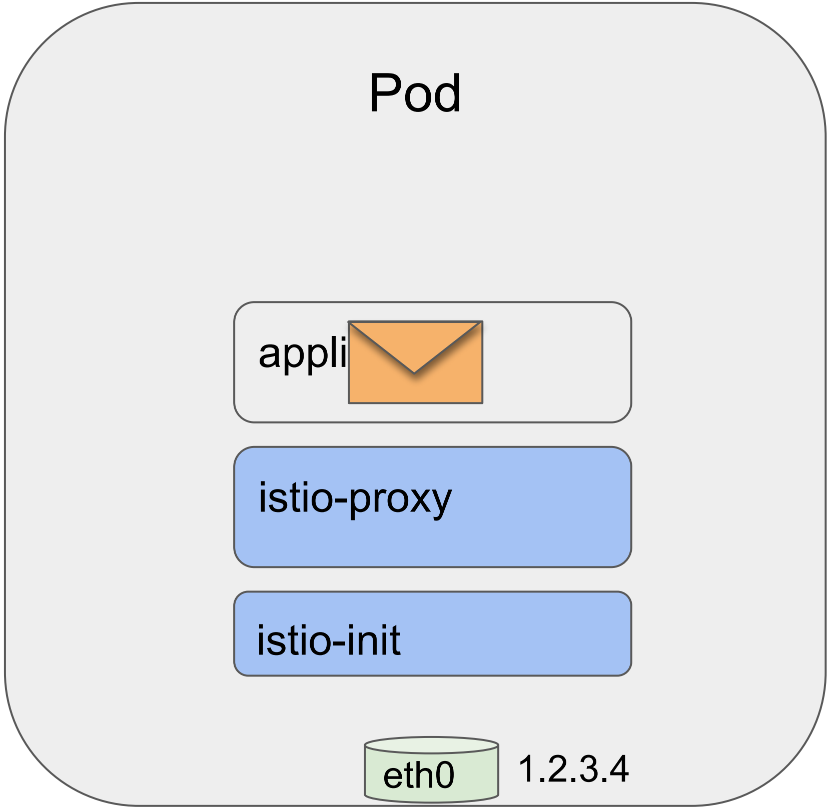
```
chain OUTPUT {
    . . .
    skuid 1337 return
    . . .
    jump ISTIO_REDIRECT
}
chain ISTIO_REDIRECT {
    meta l4proto tcp redirect to :15001
}
```

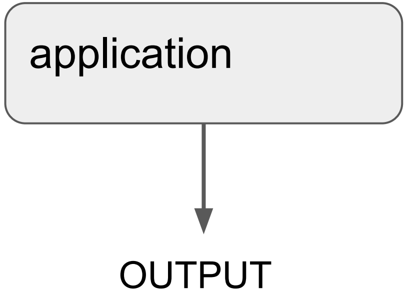
Istio - Envoy proxying



```
chain OUTPUT {
    . . .
    skuid 1337 return
    . . .
    jump ISTIO_REDIRECT
}
chain ISTIO_REDIRECT {
    meta l4proto tcp redirect to :15001
}
```

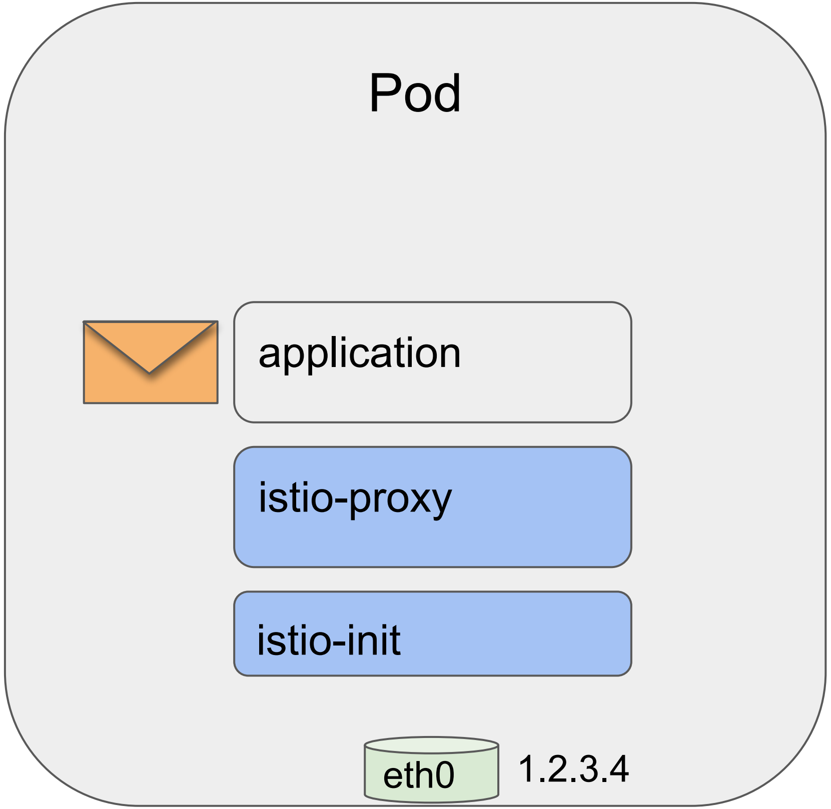
Istio - Envoy proxying

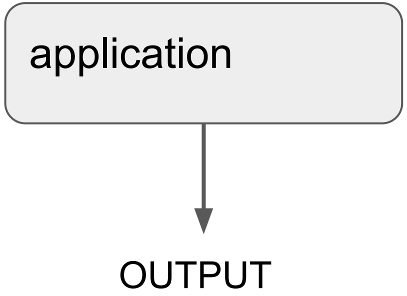




```
chain OUTPUT {
    . . .
    skuid 1337 return
    . . .
    jump ISTIO_REDIRECT
}
chain ISTIO_REDIRECT {
    meta l4proto tcp redirect to :15001
}
```

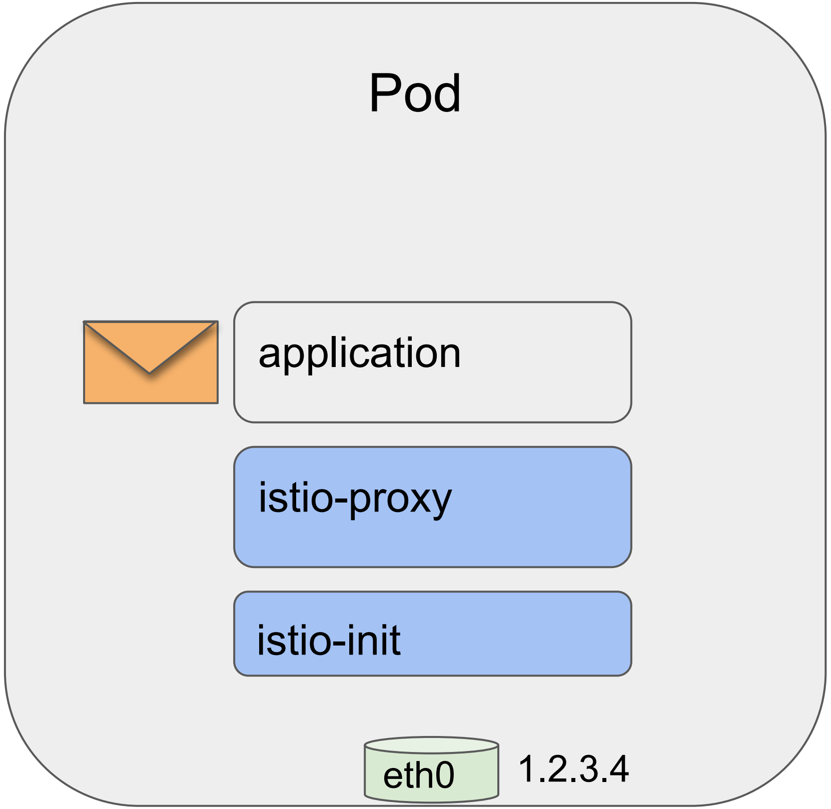
Istio - Envoy proxying

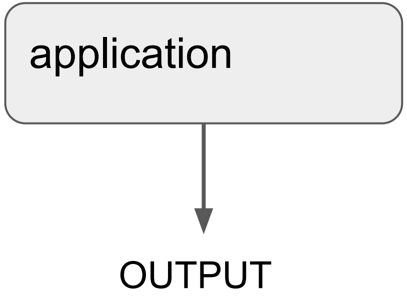




```
chain OUTPUT {
    . . .
    skuid 1337 return
    . . .
    jump ISTIO_REDIRECT
}
chain ISTIO_REDIRECT {
    meta l4proto tcp redirect to :15001
}
```

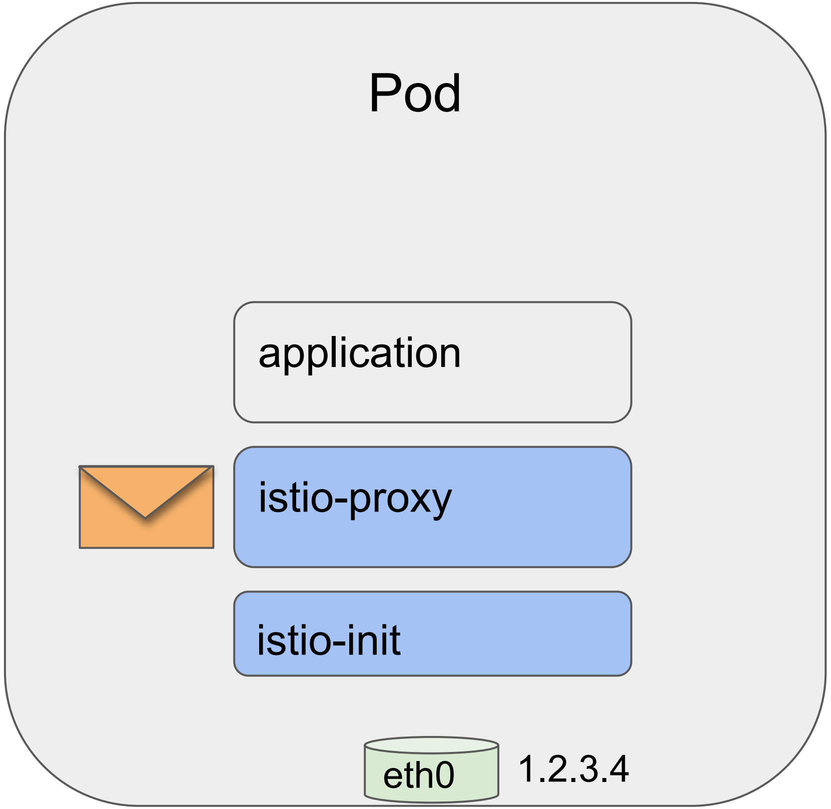
Istio - Envoy proxying

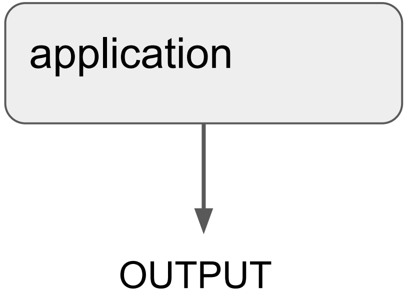




```
chain OUTPUT {
    . . .
    skuid 1337 return
    . . .
    jump ISTIO_REDIRECT
}
chain ISTIO_REDIRECT {
    meta l4proto tcp redirect to :15001
}
```

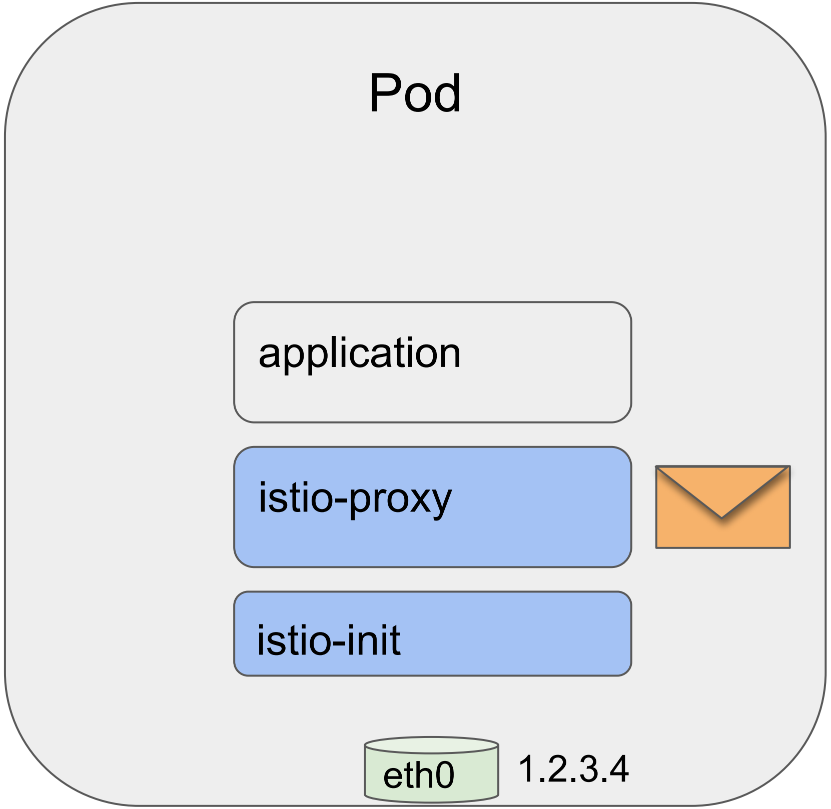
Istio - Envoy proxying

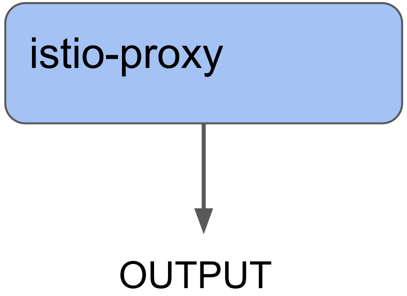




```
chain OUTPUT {
    . . .
    skuid 1337 return
    . . .
    jump ISTIO_REDIRECT
}
chain ISTIO_REDIRECT {
    meta l4proto tcp redirect to :15001
}
```

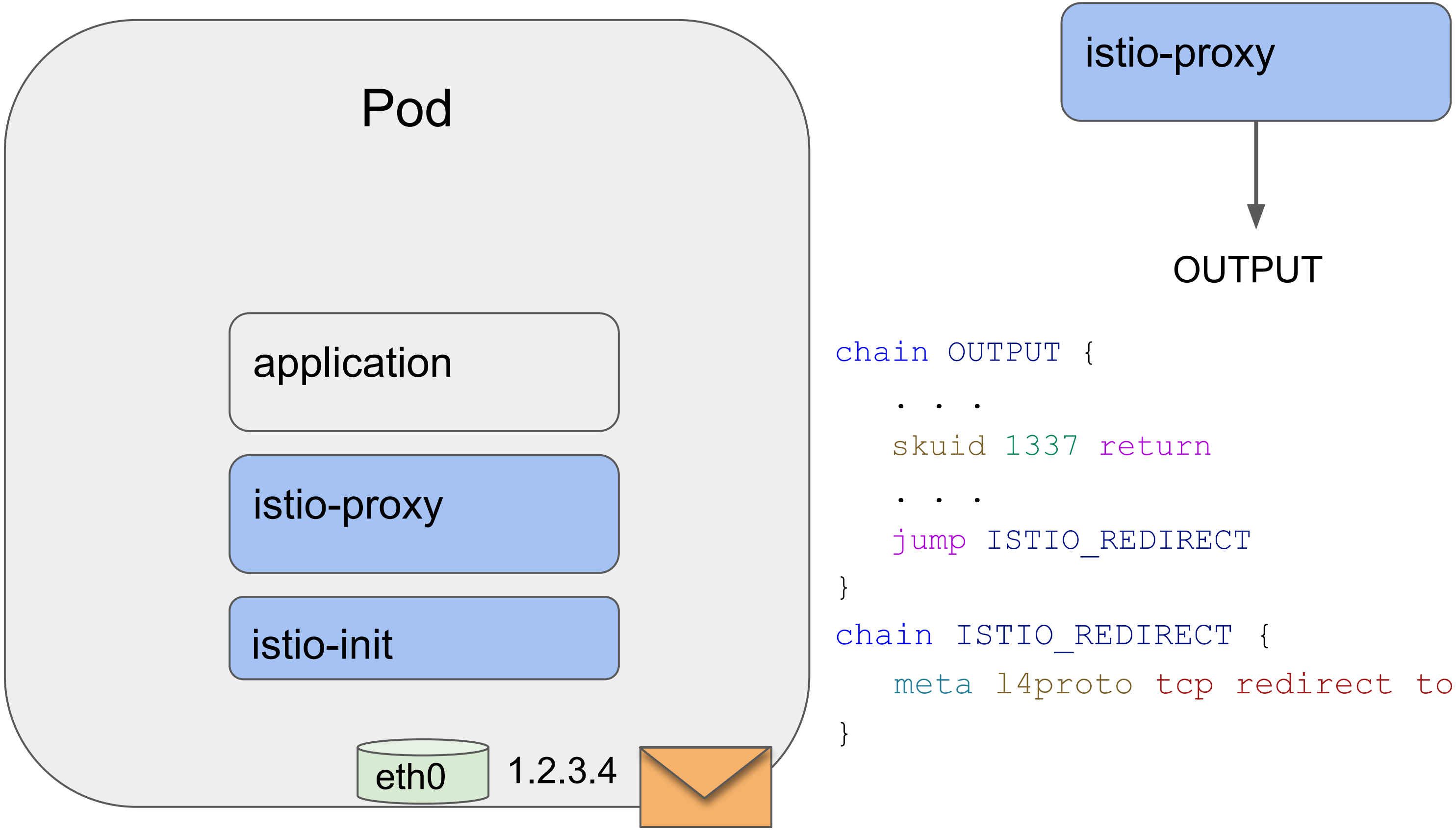
Istio - Envoy proxying





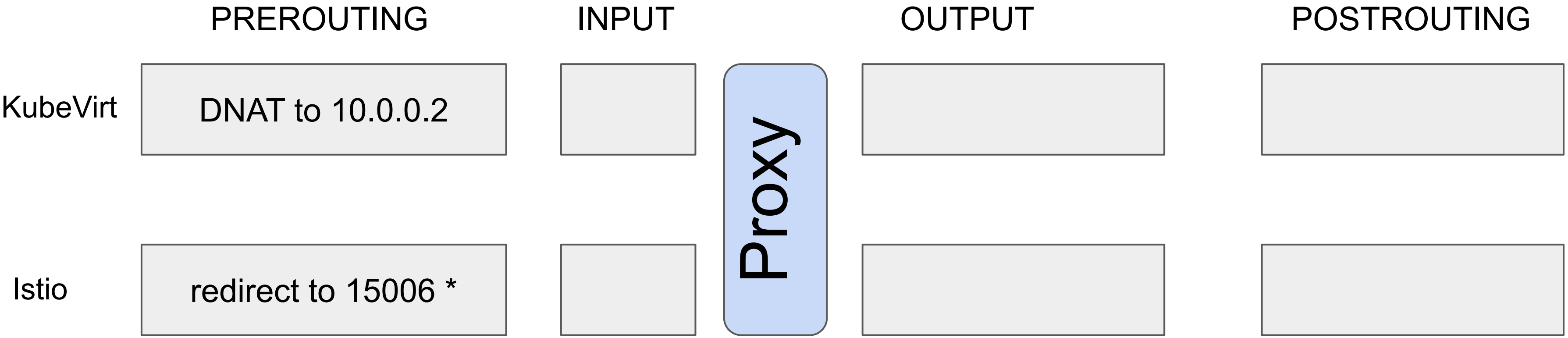
```
chain OUTPUT {
    . . .
    skuid 1337 return
    . . .
    jump ISTIO_REDIRECT
}
chain ISTIO_REDIRECT {
    meta l4proto tcp redirect to :15001
}
```

Istio - Envoy proxying



```
chain OUTPUT {
    . . .
    skuid 1337 return
    . . .
    jump ISTIO_REDIRECT
}
chain ISTIO_REDIRECT {
    meta l4proto tcp redirect to :15001
}
```

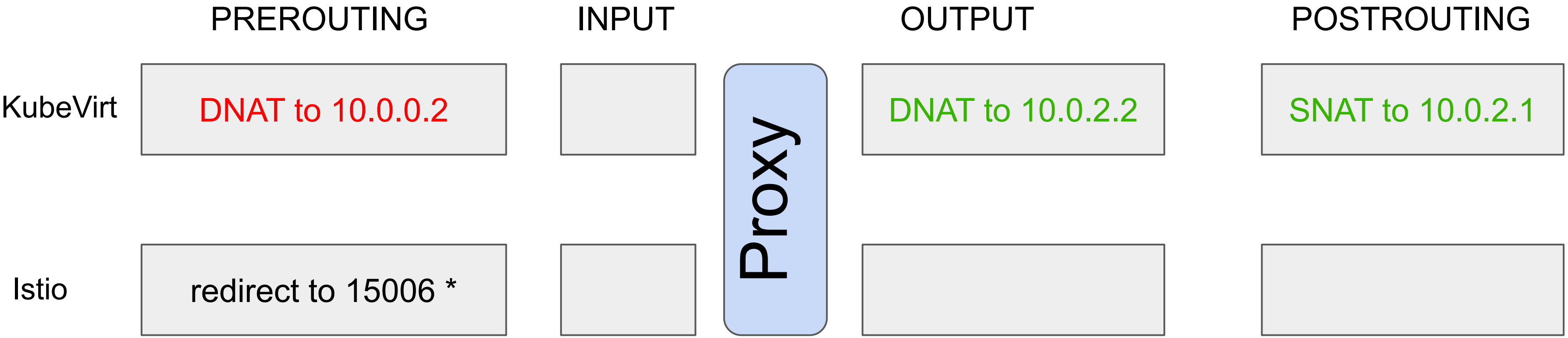
Adjusting NAT rules



	PREROUTING	INPUT		OUTPUT	POSTROUTING
KubeVirt	DNAT to 10.0.0.2		Proxy		
Istio	redirect to 15006 *				

* 15006 is istio-proxy inbound traffic listener port

Adjusting NAT rules

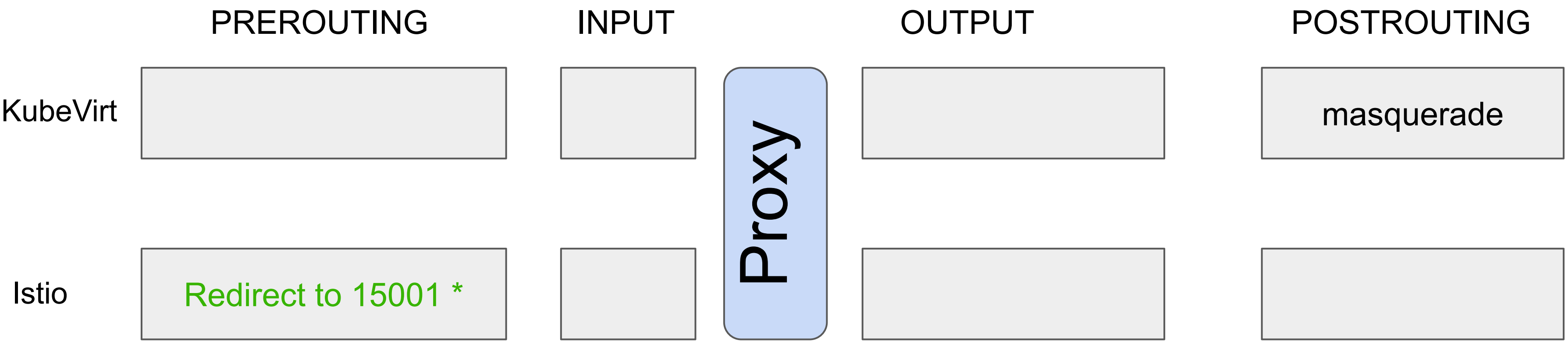


	PREROUTING	INPUT		OUTPUT	POSTROUTING
KubeVirt	DNAT to 10.0.0.2		Proxy	DNAT to 10.0.2.2	SNAT to 10.0.2.1
Istio	redirect to 15006 *				

* 15006 is istio-proxy inbound traffic listener port

```
kind: VirtualMachineInstance
metadata:
 annotations:
   "sidecar.istio.io/inject": "true"
```

Adjusting NAT rules



	PREROUTING	INPUT		OUTPUT	POSTROUTING
KubeVirt			Proxy		masquerade
Istio	Redirect to 15001 *				

* 15001 is istio-proxy outbound listener port

```
traffic.sidecar.istio.io/kubev
irtInterfaces: k6t-eth0
```

```
chain prerouting {
        iifname "k6t-eth0" jump ISTIO_REDIRECT
        iifname "k6t-eth0" return
        meta l4proto tcp jump ISTIO_INBOUND
}
```

Credits to SchSeba
Istio PR

Demo

Thank you

References

[0] Istio

[1] KubeVirt

[2] kubevirt.io: Running VMs in Istio Service Mesh

[3] jimmisong.io: Understanding Envoy sidecar traffic routing

[4] linuxtopia.org: Traversing of tables and chains

[5] nftables.org: Netfilter hooks