OpenShift 4.16 – Pod-to-Pod Encryption with Istio Ambient Mode (ztunnel) Implementation Runbook

This runbook provides a detailed, step-by-step guide to implement pod-to-pod encryption using Istio Ambient mode with ztunnel on OpenShift 4.16. The approach leverages Red Hat OpenShift Service Mesh (based on Istio 1.21+) to enable zero-trust mTLS encryption without sidecar proxies.

1. Prerequisites

* OpenShift 4.16 cluster with cluster-admin privileges.
* oc CLI 4.16 installed and logged in.
* Operators:
- Red Hat OpenShift Service Mesh v2.5 (or later)
- Optional: Jaeger, Kiali for observability.
* CNI: Default OVN-Kubernetes or OpenShift SDN works. If using Cilium, enable CNI chaining.

2. Install Red Hat OpenShift Service Mesh in Ambient Mode

a. Install Operators:
- In OpenShift Web Console → Operators → OperatorHub, install:
• Red Hat OpenShift Service Mesh
• Optional: Jaeger and Kiali

b. Create the Service Mesh Control Plane (SMCP):
Create namespace:
oc new-project istio-system

Create a YAML file named smcp-ambient.yaml:
---
apiVersion: maistra.io/v2
kind: ServiceMeshControlPlane
metadata:
name: ambient-mesh
namespace: istio-system
spec:
version: v2.5
mode: Ambient
gateways:
enabled: true

Apply:
oc apply -f smcp-ambient.yaml

Wait for pods to be Ready:
oc get pods -n istio-system

3. Enable Ambient Mode for Application Namespace

Label the application namespace to join the ambient mesh:
oc label namespace my-app istio.io/dataplane-mode=ambient

4. Verify ztunnel Deployment

Check that ztunnel DaemonSet is running on each node:
oc get daemonset ztunnel -n istio-system

5. Enforce Pod-to-Pod mTLS

Create PeerAuthentication policy in the application namespace:
---
apiVersion: security.istio.io/v1beta1
kind: PeerAuthentication
metadata:
name: default
namespace: my-app
spec:
mtls:
mode: STRICT

Apply:
oc apply -f peerauth.yaml

6. Testing the Setup

Deploy two test pods:
oc run pod-a --image=quay.io/centos/centos:stream9 -- sleep infinity
oc run pod-b --image=quay.io/centos/centos:stream9 -- sleep infinity

From pod-a:
oc exec -it pod-a -- curl http://pod-b.my-app.svc.cluster.local:80

Use Kiali or ztunnel logs to confirm mTLS (look for tls: true).

7. Observability (Optional)

Use Kiali dashboard to view the service mesh topology with mTLS locks.
Optionally, use istioctl for deeper inspection:
istioctl x ztunnel-config

Key Notes

* Performance: ztunnel runs once per node, reducing sidecar overhead.
* Combine PeerAuthentication and AuthorizationPolicy for Zero Trust Network Access.
* Service Mesh 2.5 in OpenShift 4.16 supports Ambient mode as GA, no tech preview flag needed.

Quick Checklist

Step	Command/Action
Install Operators	Web Console → OperatorHub
Create SMCP (ambient)	oc apply -f smcp-ambient.yaml

Label Namespace	oc label namespace my-app istio.io/dataplane-mode=ambient
Enforce mTLS	oc apply -f peerauth.yaml
Verify ztunnel	oc get daemonset -n istio-system ztunnel