Deckhouse
Kubernetes Platform

Istio

Мультикластер
Общие принципы

cluster.local

cluster.local

Есть два кластера под управлением Istio с одинаковым Cluster Domain.

cluster.local

namespace: abc

Pod: foo

Service: foo

cluster.local

namespace: xyz

Pod: bar

Service: bar

В них работают приложения.

cluster.local

Certificates

Root CA

namespace: abc

Pod: foo

Service: foo

cluster.local

Certificates

Root CA

namespace: xyz

Pod: bar

Service: bar

У каждого кластера есть хранилище доверенных сертификатов, которое содержит единственный корневой сертификат кластера.

cluster.local

Certificates

Root CA

namespace: abc

Pod: foo

Service: foo

cluster.local

Certificates

Root CA

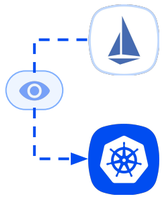
namespace: xyz

Pod: bar

Service: bar

Этими корневыми сертификатами подписаны индивидуальные сертификаты подов для нужд Mutual TLS.

cluster.local



Certificates

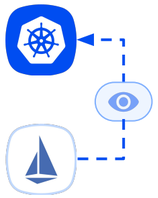
Root CA

namespace: abc

Pod: foo

Service: foo

cluster.local



Certificates

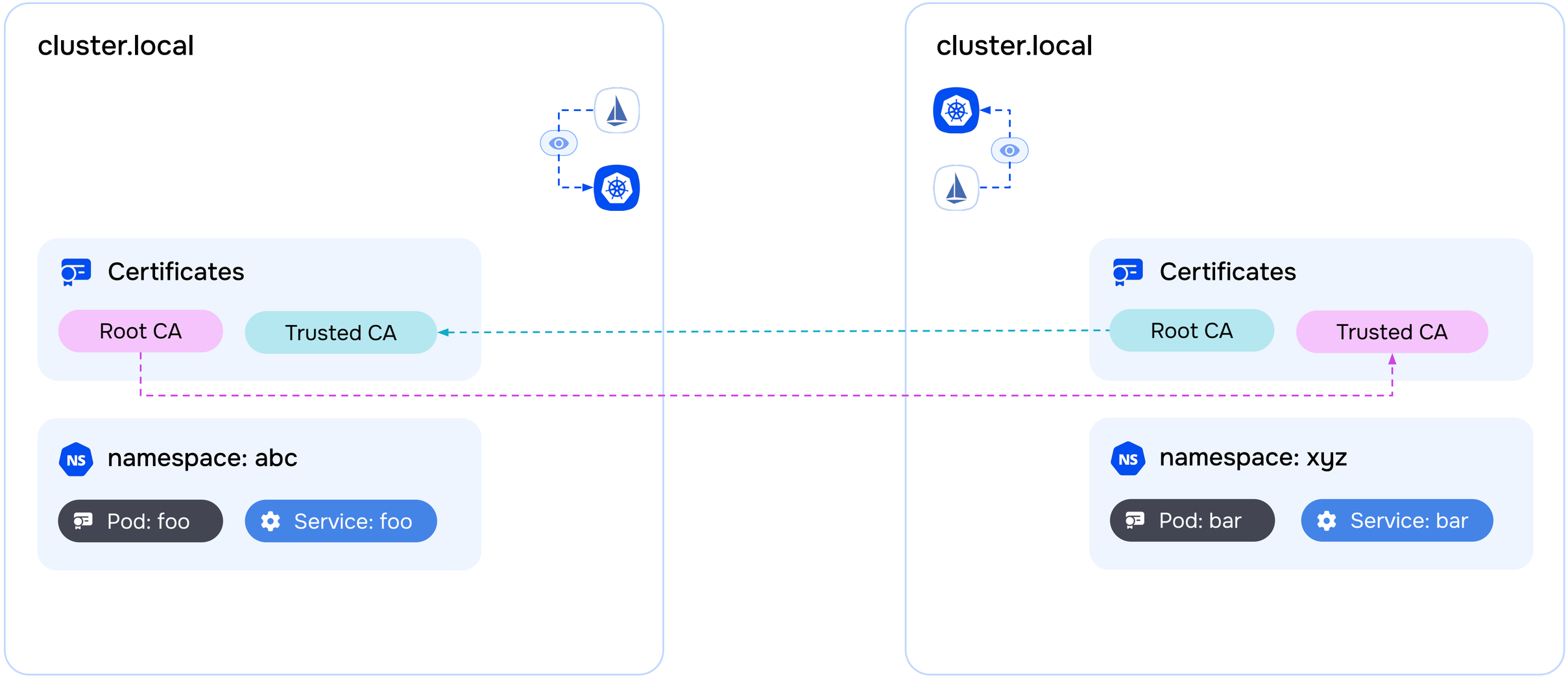
Root CA

namespace: xyz

Pod: bar

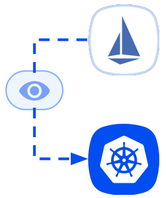
Service: bar

Control plane Istio взаимодействует с локальным kube-apiserver для сбора информации о сервисах,
их адресах и состоянии, собранная информация консолидируется и рассылается по сайдкарам приложений.



Для обеспечения взаимного доверия необходимо взаимно обменяться корневыми сертификатами и поместить их в хранилище доверенных сертификатов.

cluster.local



Certificates

Root CA

Trusted CA

namespace: abc

Pod: foo

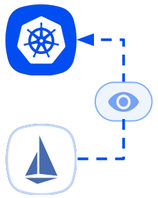
Service: foo

Cloud

Ingress-gateway

Ingress-gateway

cluster.local



Certificates

Root CA

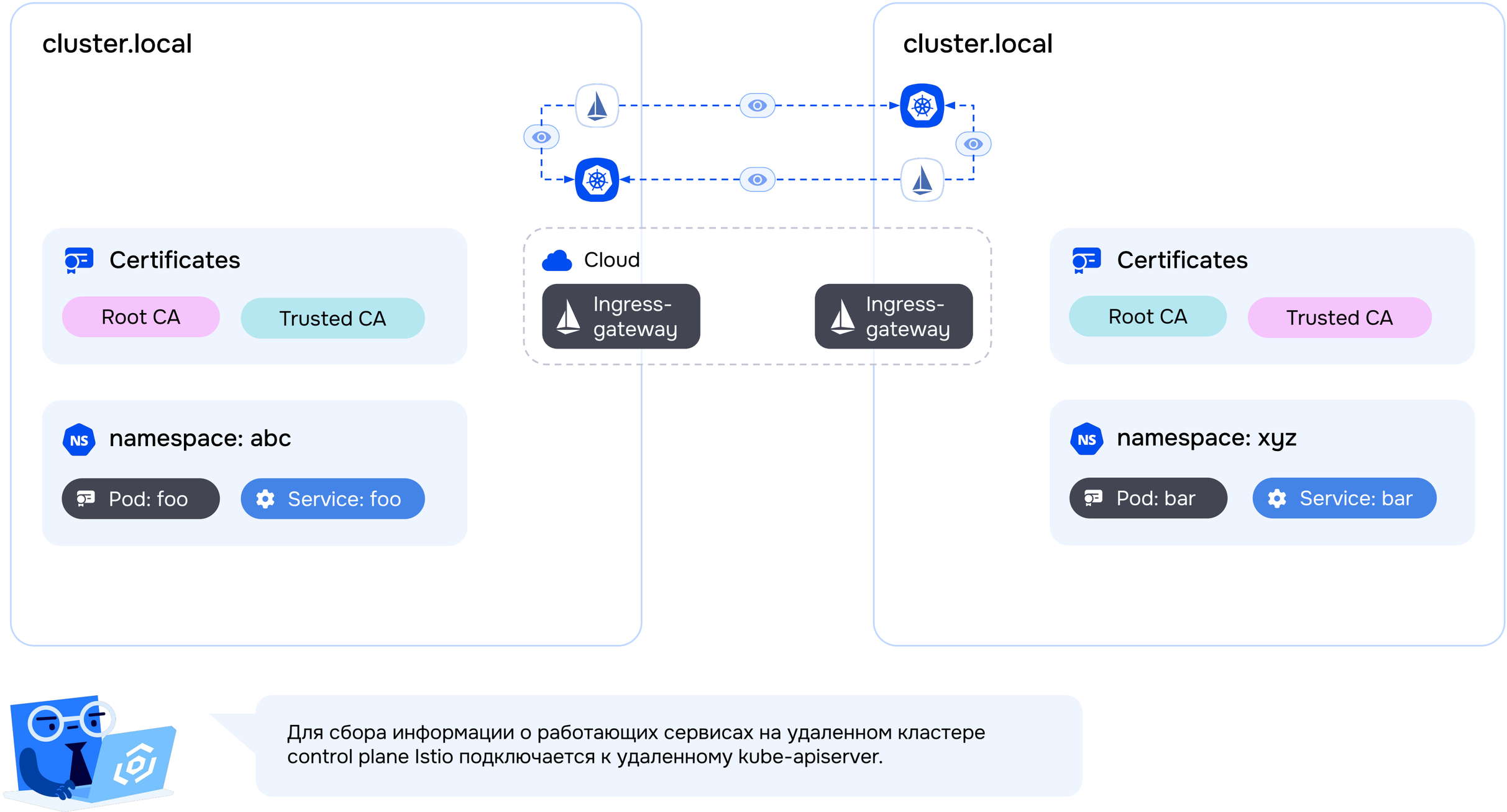
Trusted CA

namespace: xyz

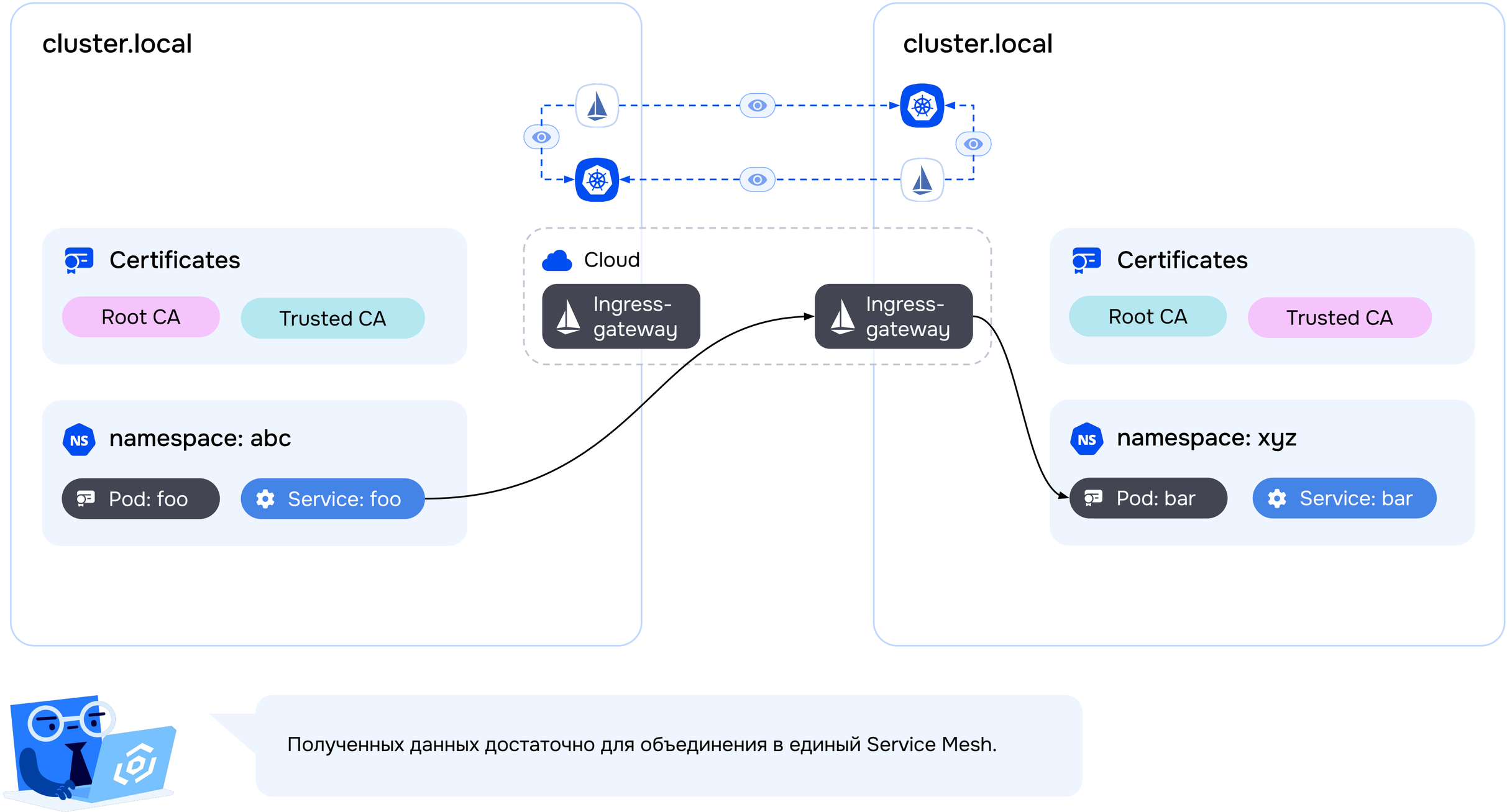
Pod: bar

Service: bar

Для обмена трафиком между подами из разных кластеров используются ingress-gateway, которые позволяют принимать Mutual TLS запросы из соседних доверенных кластеров.



Для сбора информации о работающих сервисах на удаленном кластере control plane Istio подключается к удаленному kube-apiserver.



Полученных данных достаточно для объединения в единый Service Mesh.