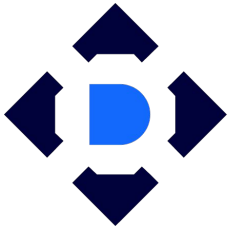
Istio

Multicluster

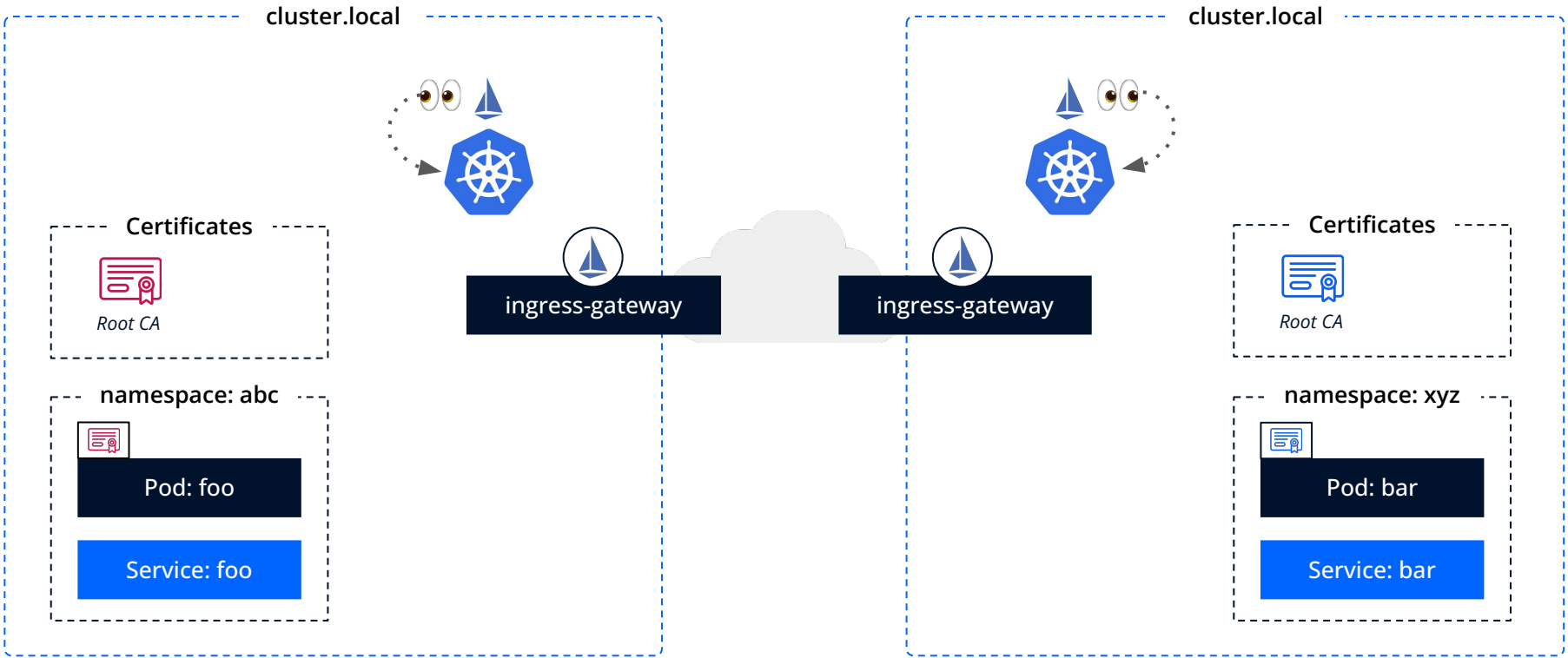
IstioMulticluster



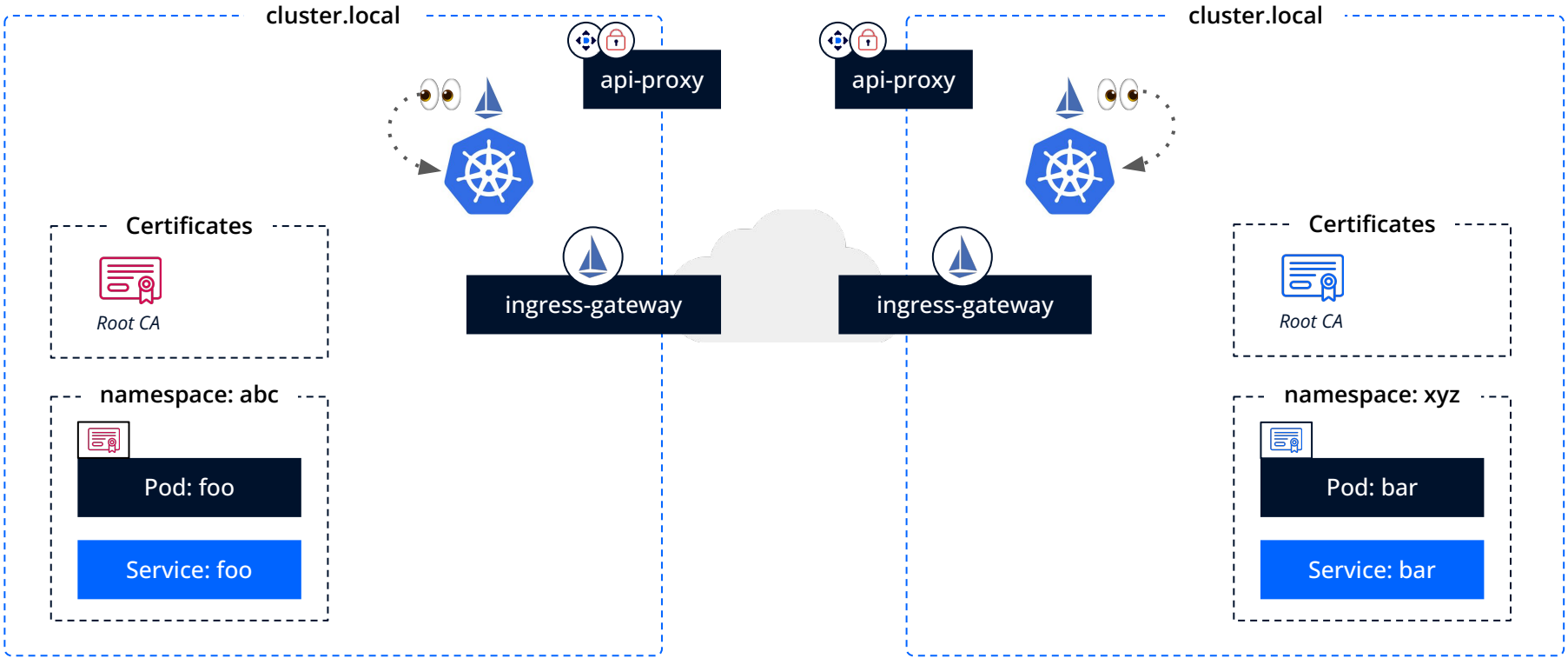
FLANT

Deckhouse

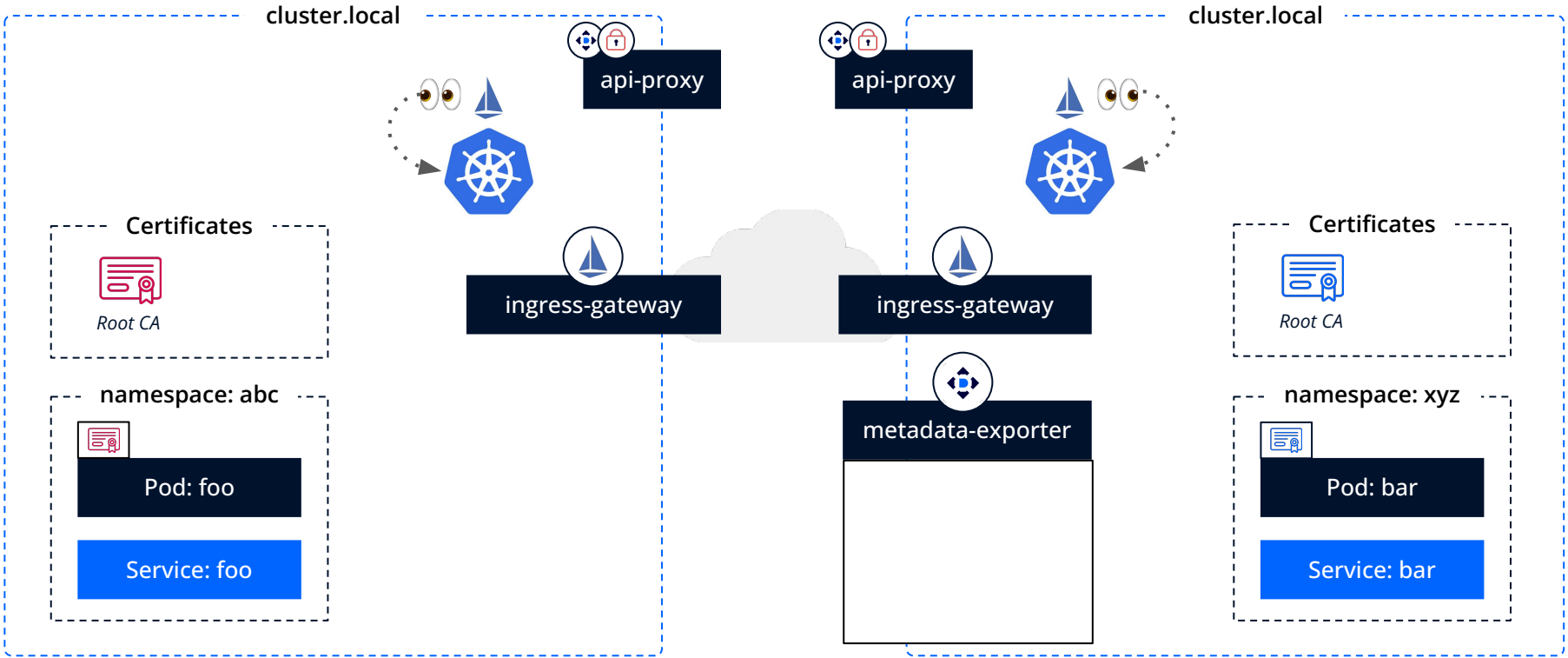
Kubernetes Platform



Suppose, there are two clusters managed by Istio with the identical Cluster Domain. Each cluster has its own Istio root certificate. You need to combine the two clusters into a single Service Mesh.

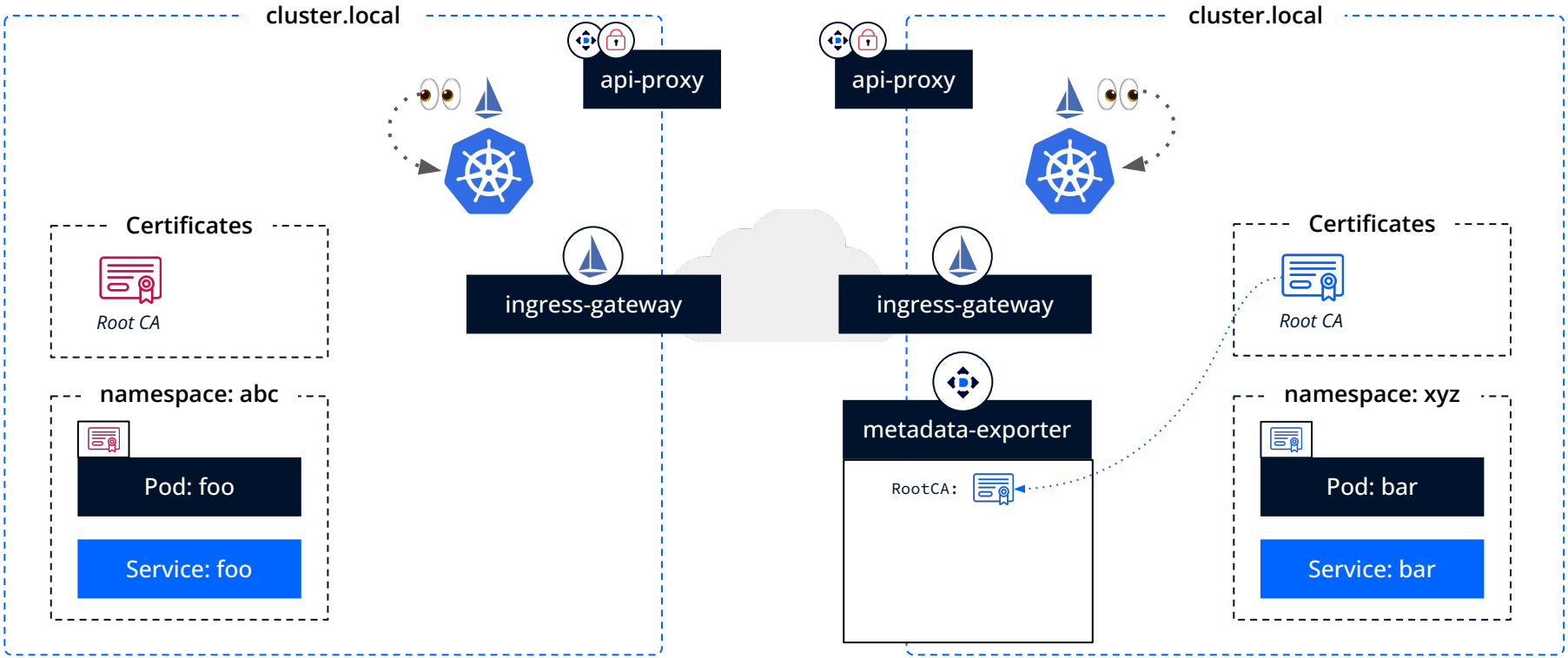


To do this, enable the istio.multicluster.enabled module parameter by setting it to true (istio.multicluster.enabled = true). The api-proxy component will start; it will provide the remote clusters with limited access to the apiserver.

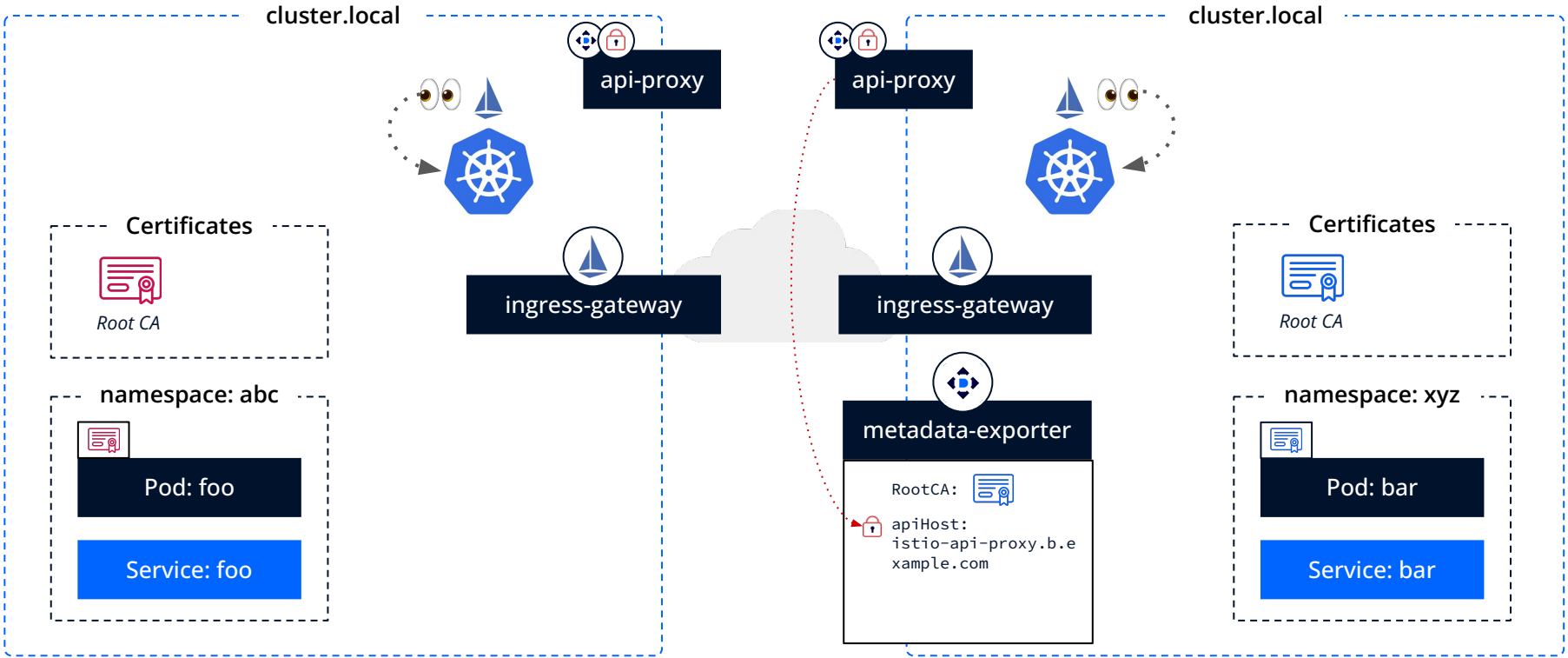


Also, a metadata-exporter component will run to collect and publish meta-information about the cluster, such as:

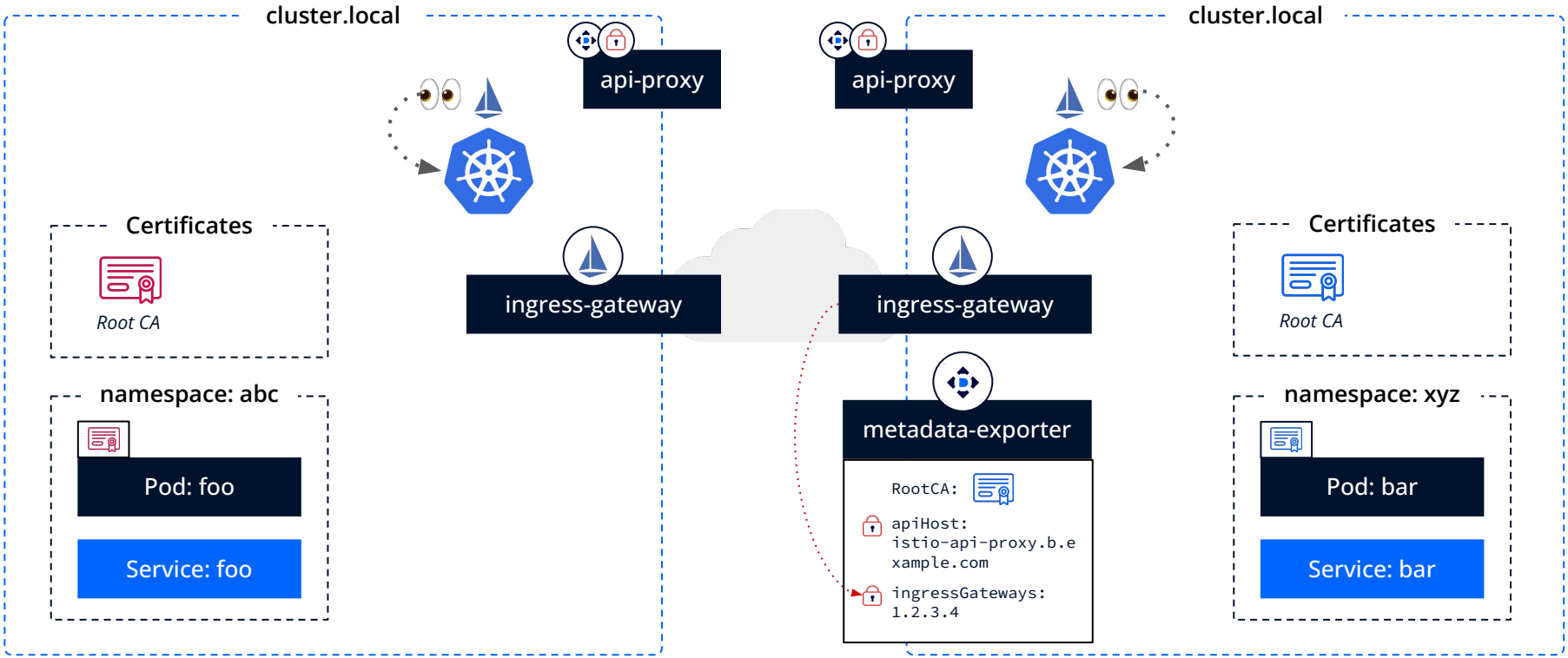
(Note that we will illustrate the process for the right cluster. However, it is the same for both of them.)



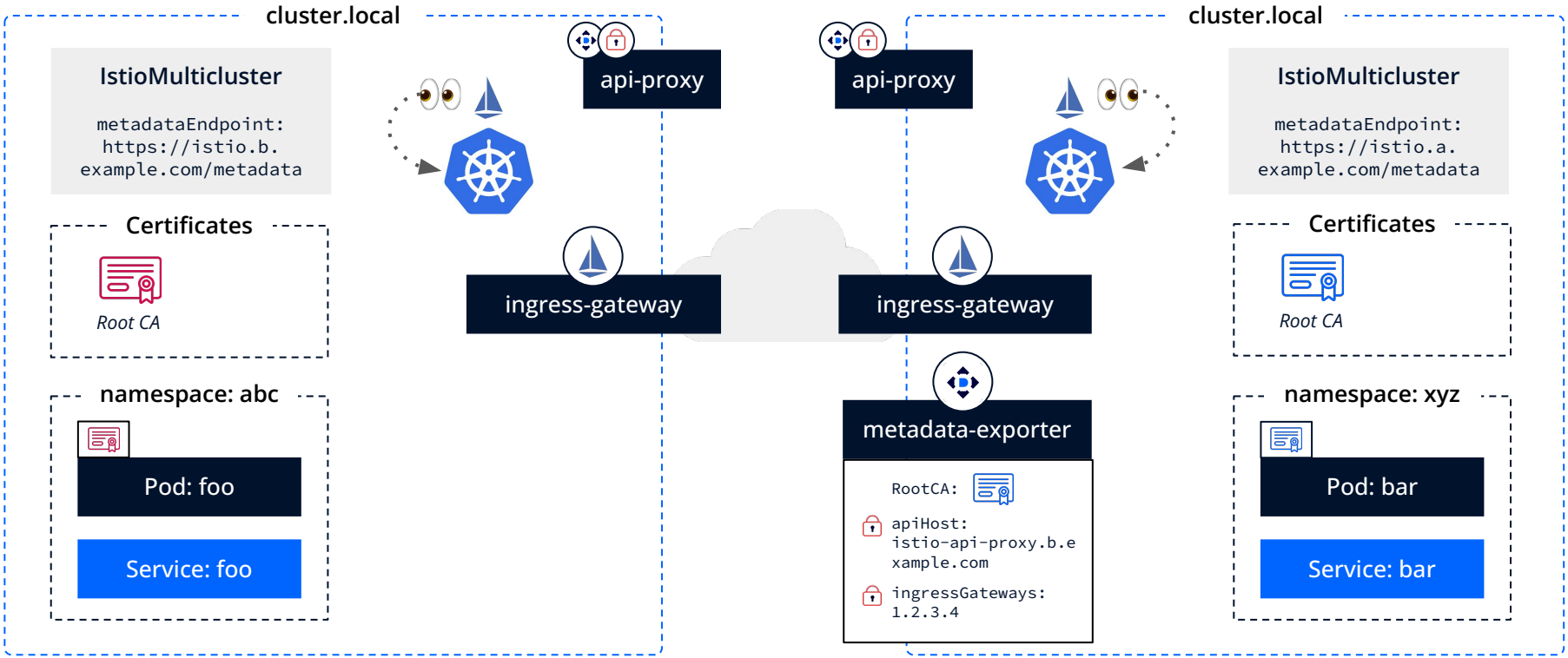
- the public part of the Istio root certificate;



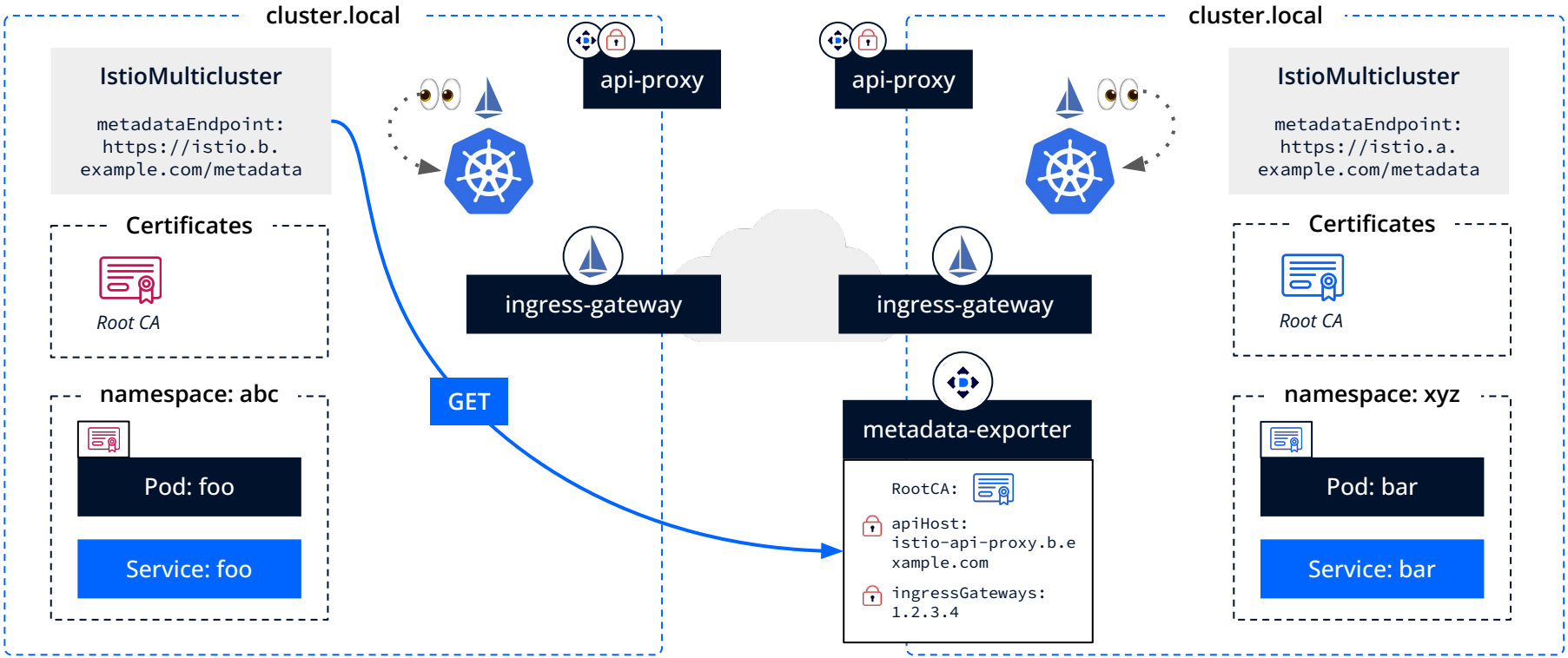
- proxy address to access the apiserver;



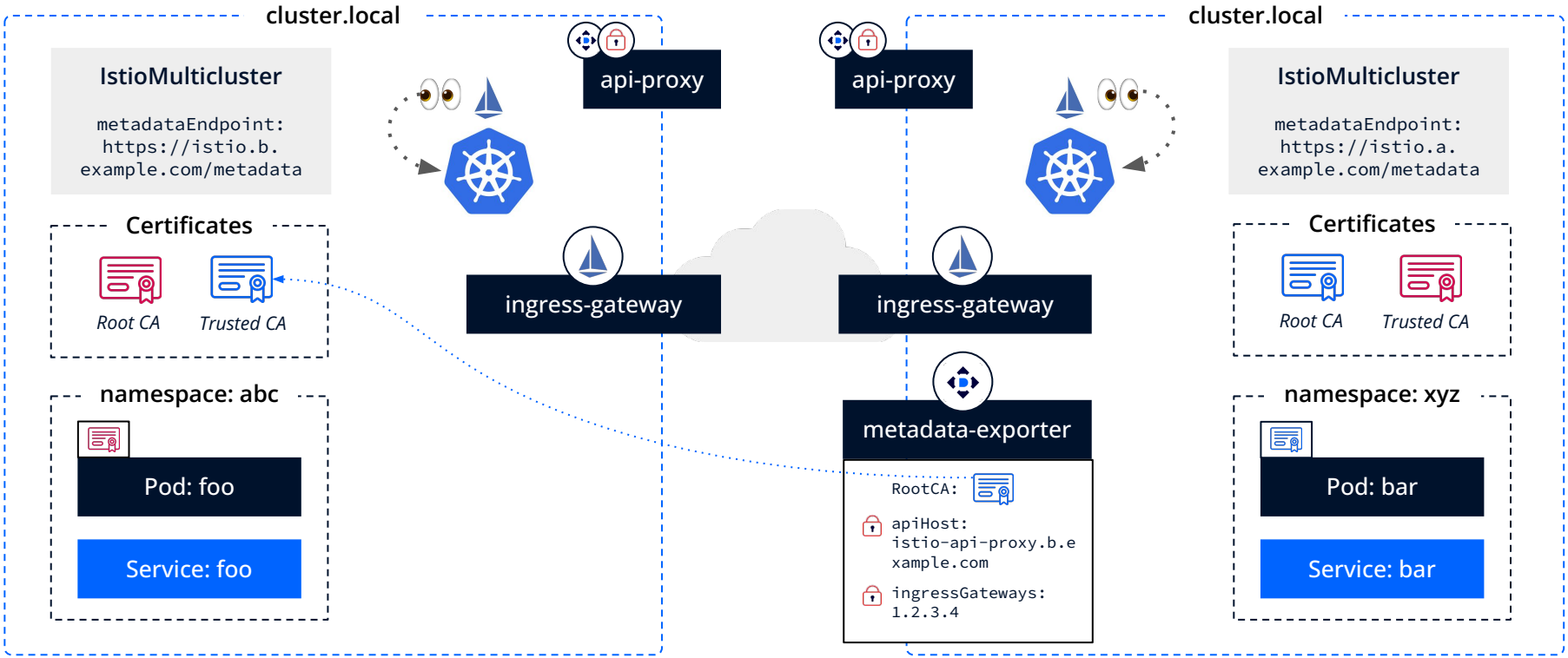
- public addresses of ingress-gateway components (accessible only from the federated clusters), and other meta-information.



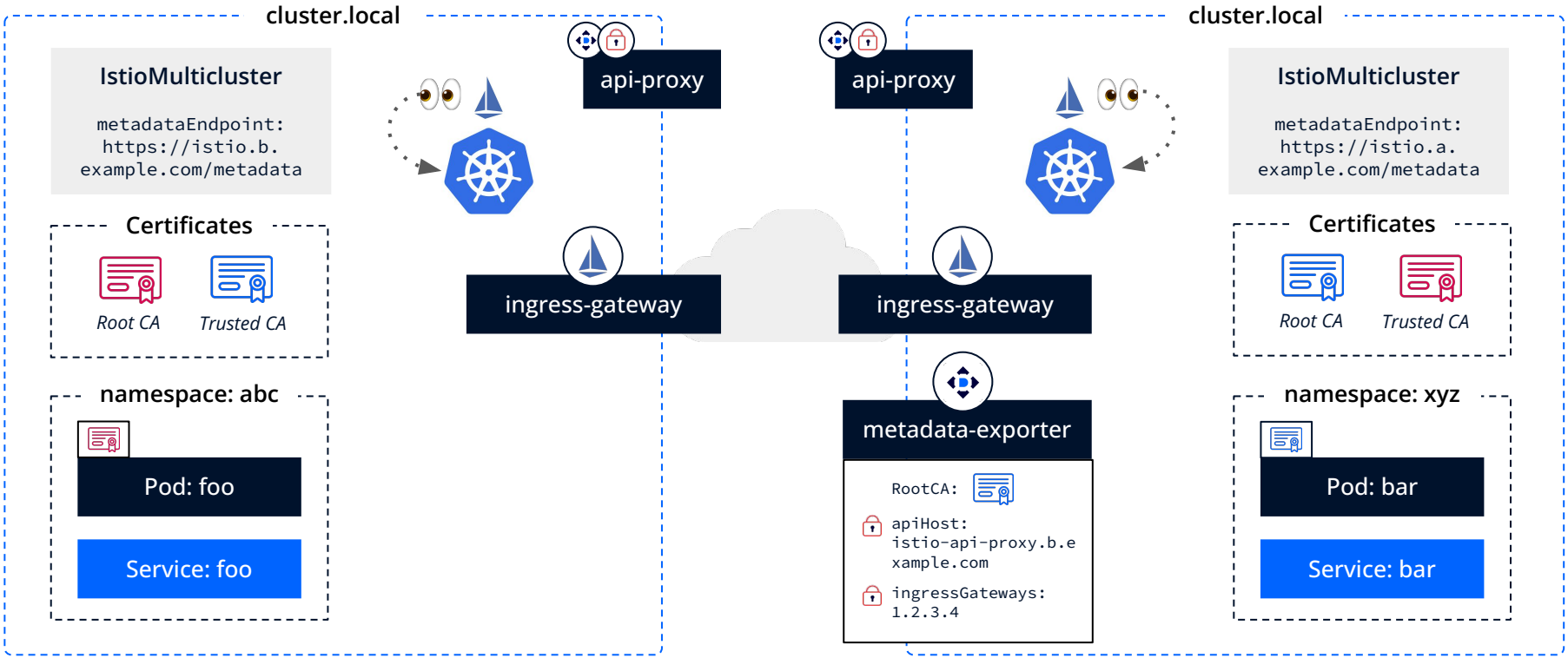
IstioMulticluster resources are simultaneously created on the clusters. They point to a location with the remote cluster's metadata. The multicluster is then automatically established.



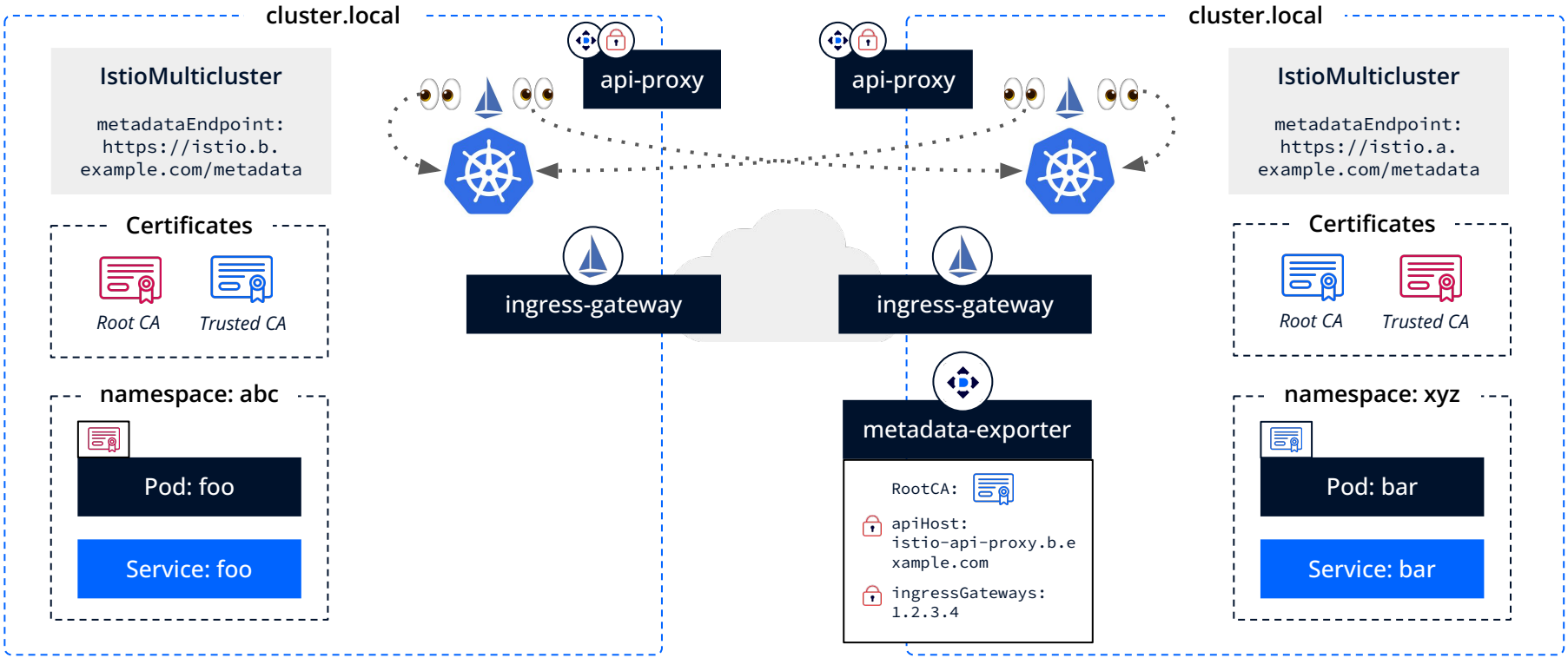
Deckhouse collects the remote metadata,..



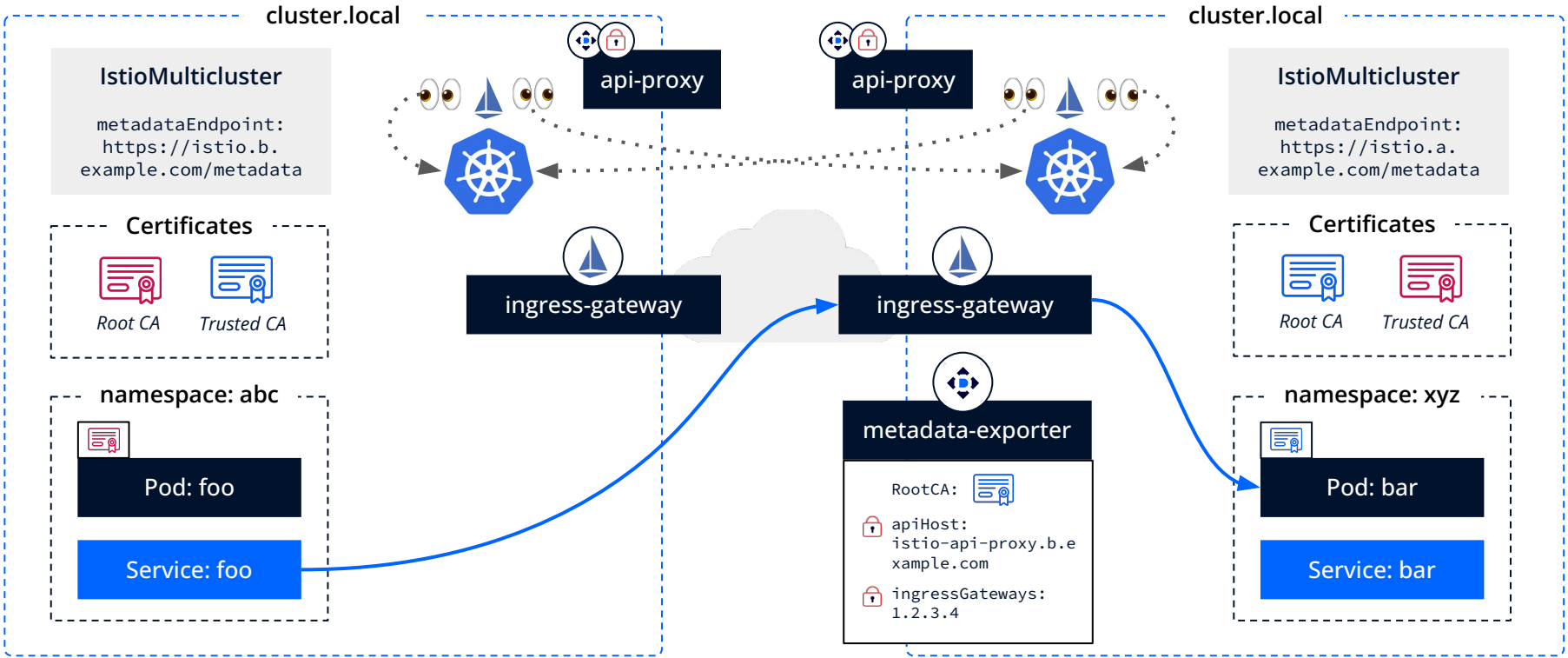
...pulls the public root certificate, and exchanges keys to access the private metadata.



Next, it fetches information about the API proxy addresses for accessing the remote apiserver and the ingress gateway addresses through which applications on the remote cluster are accessible.



Then the Istio control plane connects to the remote apiserver,..



...and cluster applications become mutually accessible.