



OpenShift Security

The usage of PKI Infrastructures

Stronger Platform Security

Defense in Depth



CONTROL
Application Security



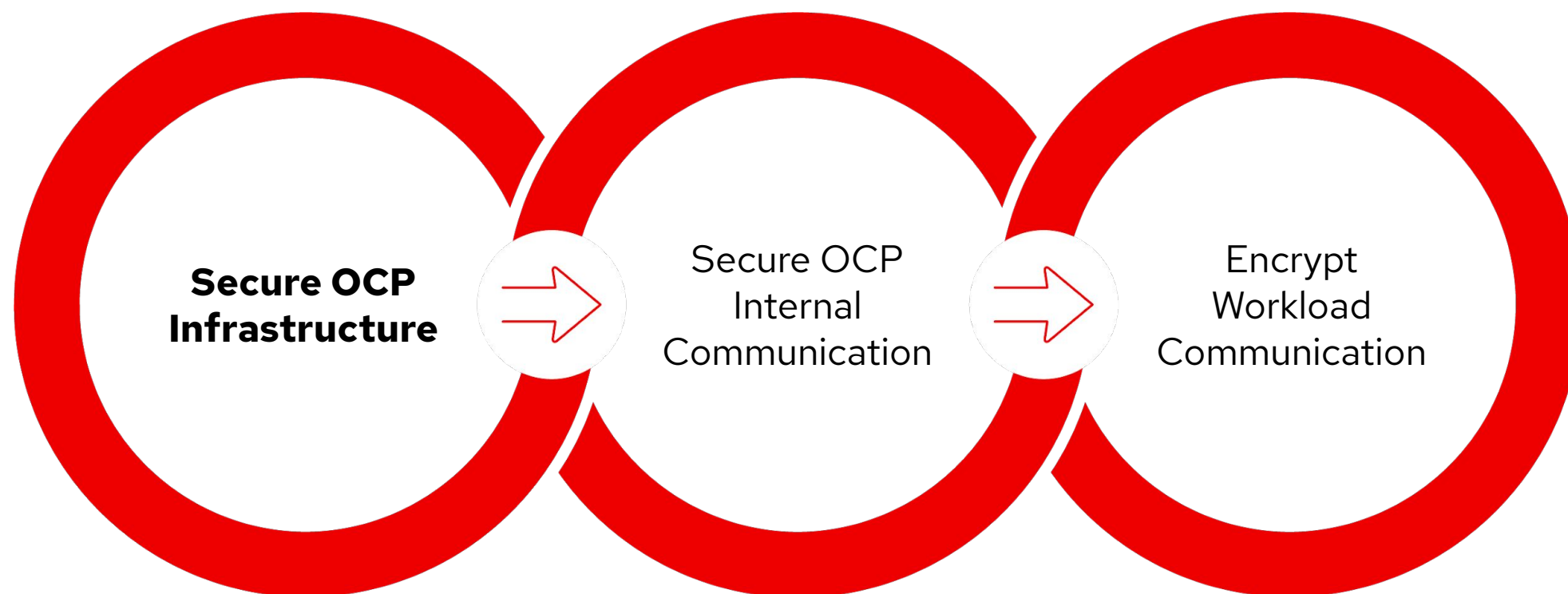
DEFEND
Infrastructure



EXTEND

- [FIPS Compliance](#)
- [Encrypt etcd datastore](#)
- [RHEL CoreOS network bound disk encryption](#)
- [Private clusters with existing VPN / VPC](#)
- [Internal ingress controller](#)
- [Ingress Cipher & TLS Policy Configuration](#)
- [Log forwarding \(tech preview\)](#)

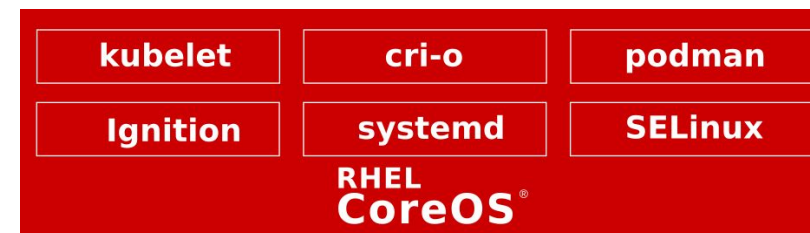
PKI in OpenShift



Red Hat Enterprise Linux CoreOS

4.3 Image Availability: (* = new)

- OpenStack
- GCP
- Azure
- Amazon
- vSphere
- Bare Metal (unified x86_64 image)*
- IBM Z (DASD & FCP via z-stream)*



FIPS mode support:

- Enforces FIPS validated ciphers for node-level cryptography
- Configurable at install/provisioning

Network Bound Disk Encryption:

- Provides encryption for local storage
- Addresses disk/image theft
- Platform/cloud agnostic implementation
- TPM/vTPM (v2) and Tang endpoints for automatic decryption

Kmods via containers:

- A framework to build and load 3rd party kmods
- Viable for drivers unsuitable for the SRO

OpenShift 4 Fips 140-2 Compliant Cluster

FIPS ready Services

- When built with RHEL 7 base image

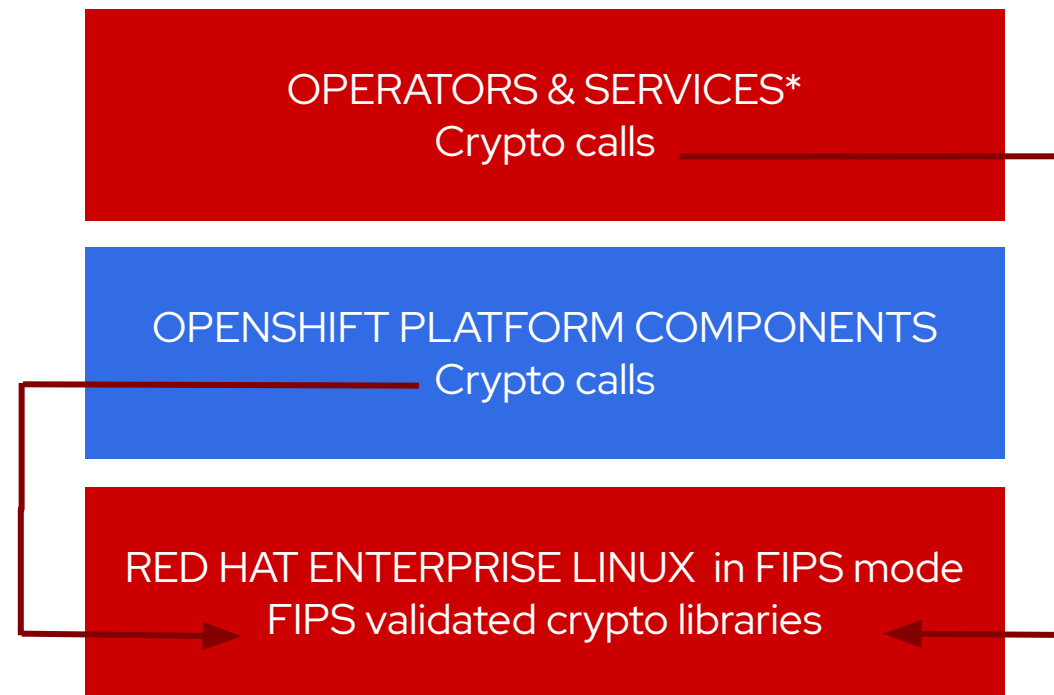
OpenShift calls FIPS validated crypto

- When running on RHEL in FIPS mode, OpenShift components bypass go cryptographic routines and call into a RHEL FIPS 140-2 validated cryptographic library
- This feature is specific to binaries built with the RHEL go compiler and running on RHEL

RHEL CoreOS FIPS mode

- Configure at install to enforce FIPS validated ciphers for node-level cryptography

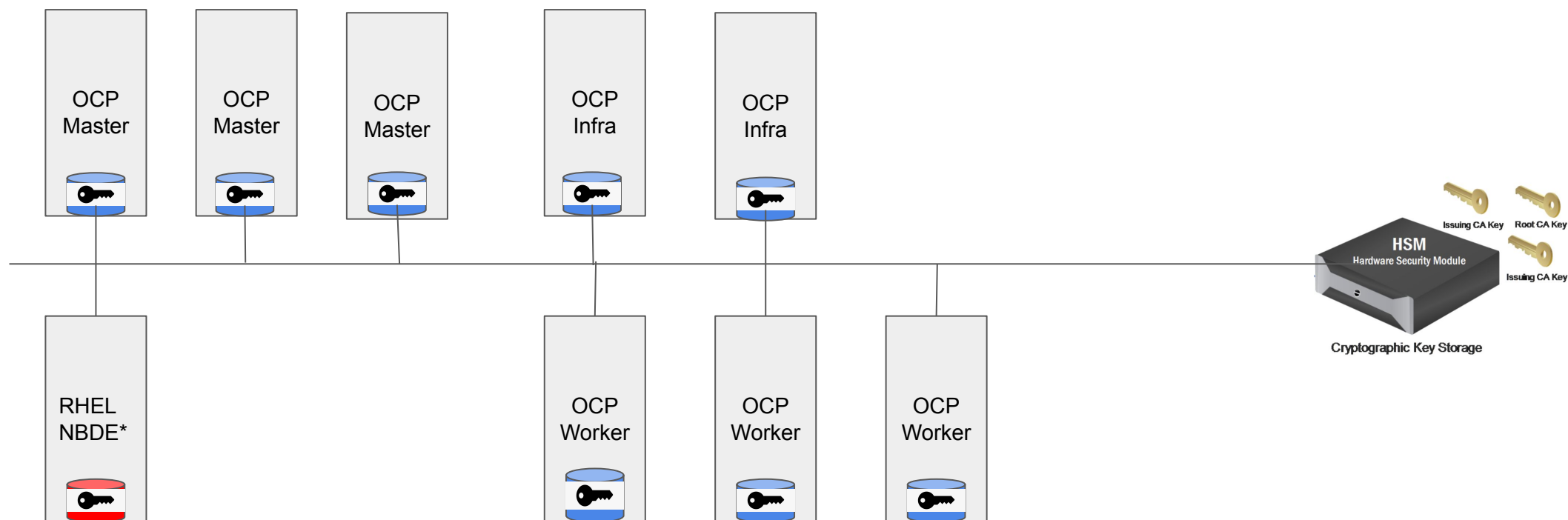
Note: products are not FIPS validated, only libraries.



*When built with RHEL base images

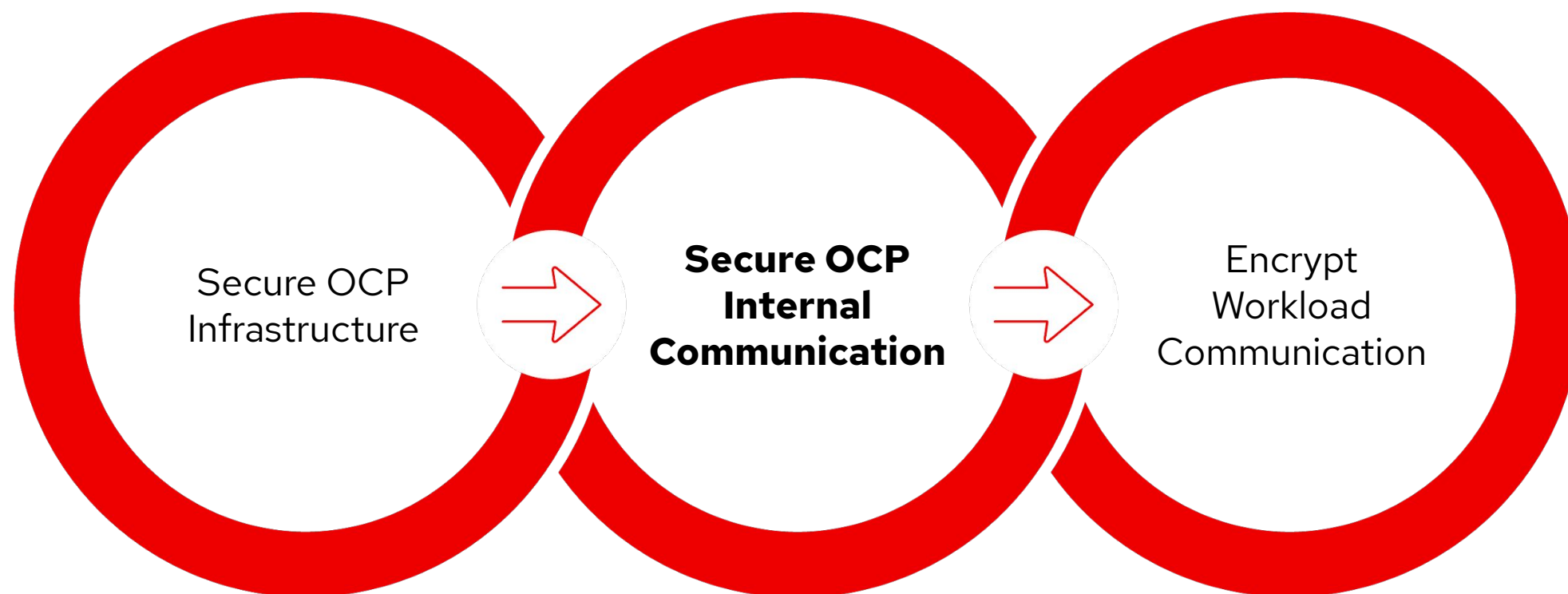
[More about RHEL go and FIPS 140-2](#)

Encrypting the Disk of the OCP Nodes



*Network bound disk encryption ([Clavis & Tang](#))

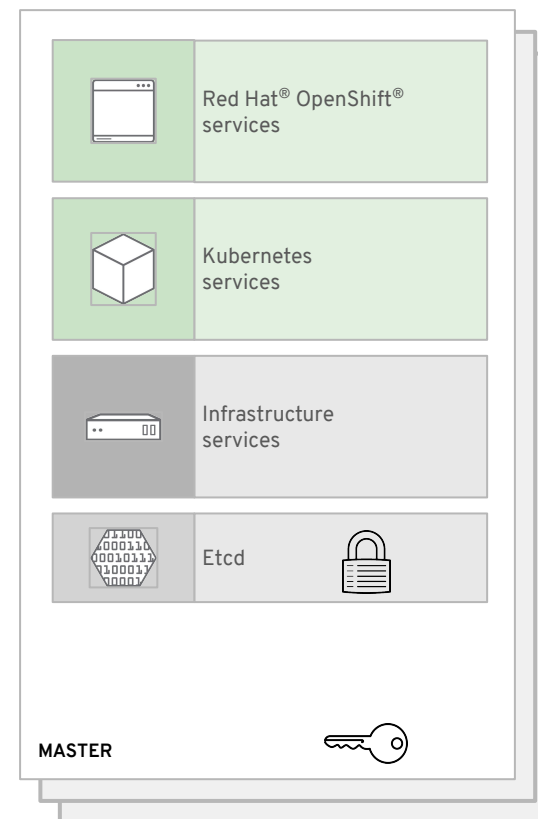
PKI in OpenShift



OpenShift 4 etcd Encryption

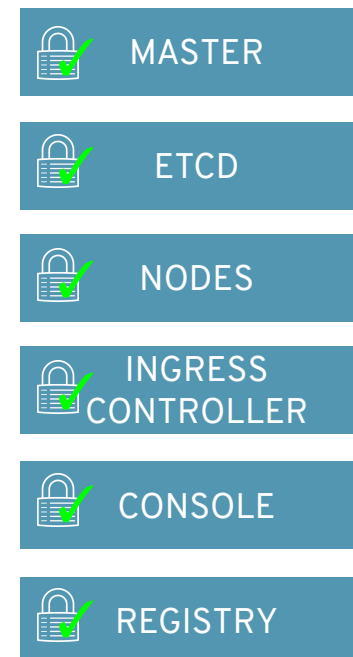
Encrypt secrets, config maps...

- Encryption of the etcd datastore is optional. Once enabled, encryption cannot be disabled.
- The aes-cbc cipher is used.
- Keys are created and automatically rotated by an operator and stored on the master node's file system.
- Keys are available as a secret via the kube API to a cluster admin.
- Assuming a healthy cluster: after enabling encryption, within a day, all relevant items in etcd are encrypted
- Backup: The etcd data store should be backed up separately from the file system with the key.
- Disaster recovery: a backup of both the encrypted etcd data and encryption keys must be available.

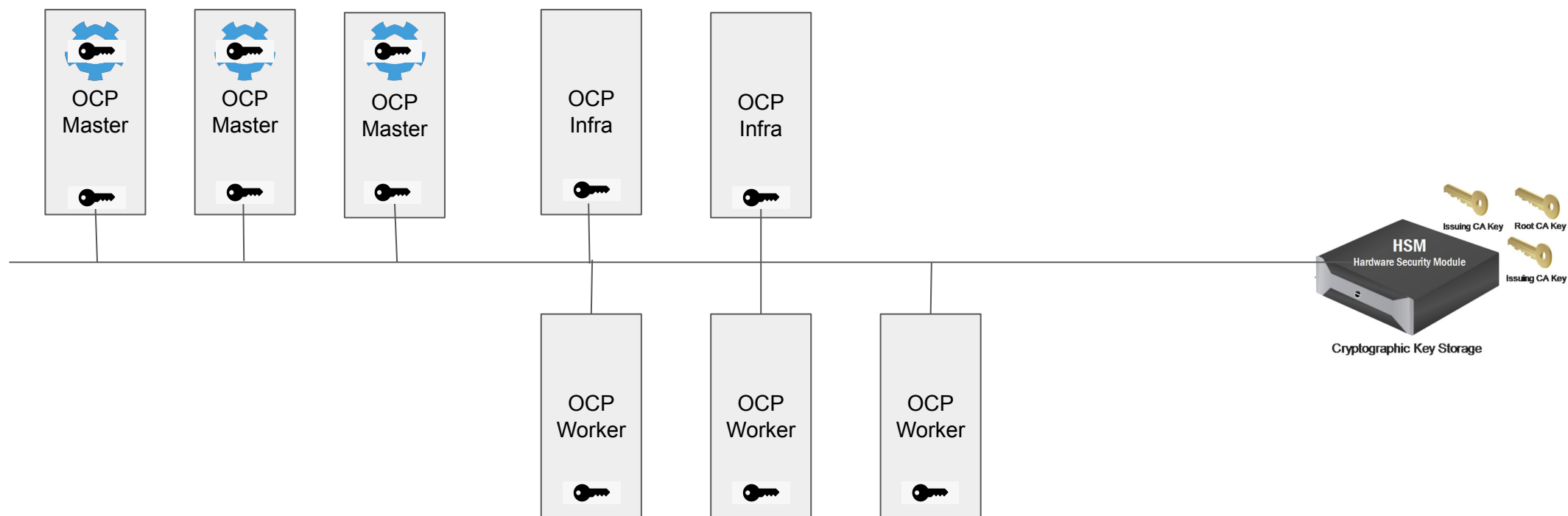


Certificates and Certificate Management

- OpenShift provides its own internal CA
- Certificates are used to provide secure connections to
 - master (APIs) and nodes
 - Ingress controller and registry
 - etcd
- Certificate rotation is automated
- Optionally configure external endpoints to use custom certificates

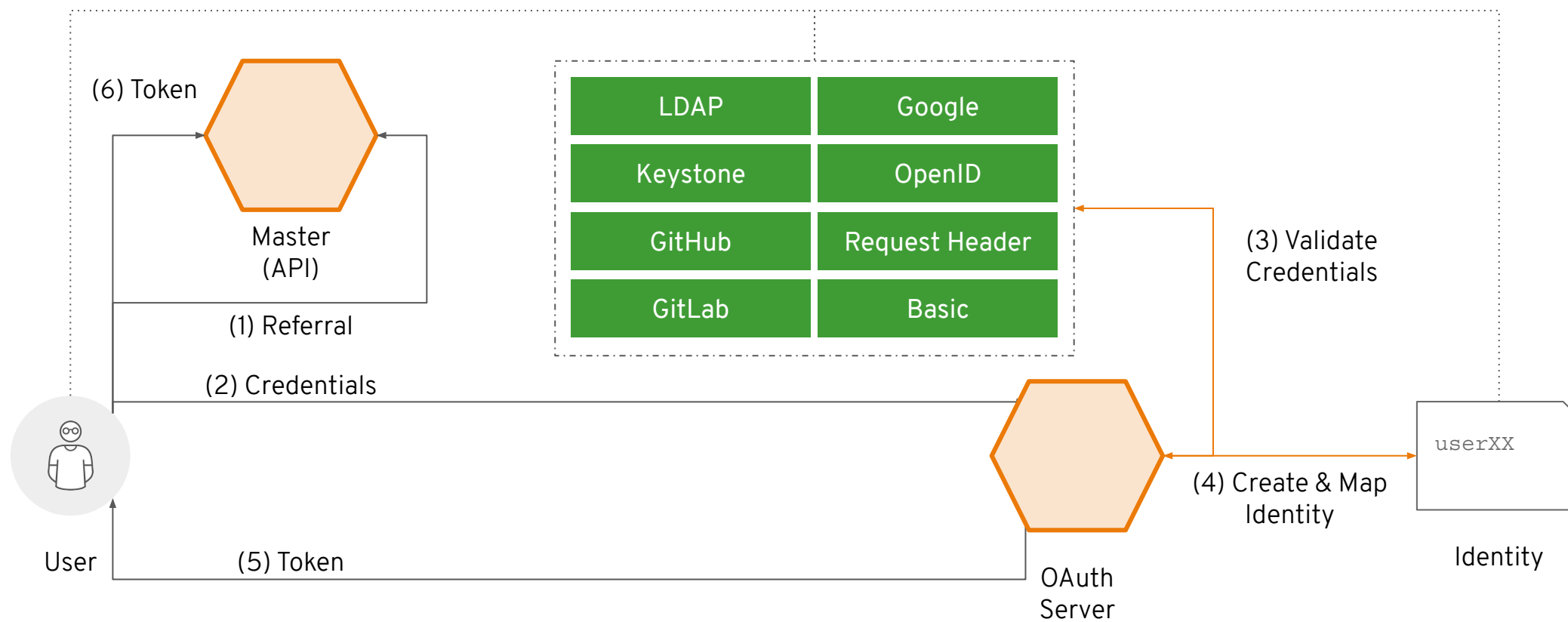


Encrypting the OCP Network traffic and the etcd Database



Identity and Access Management

Identity and Access Management



Fine-Grained RBAC

- Project scope & cluster scope available
- Matches request attributes (verb,object,etc)
- If no roles match, request is denied (deny by default)
- Operator- and user-level roles are defined by default
- Custom roles are supported

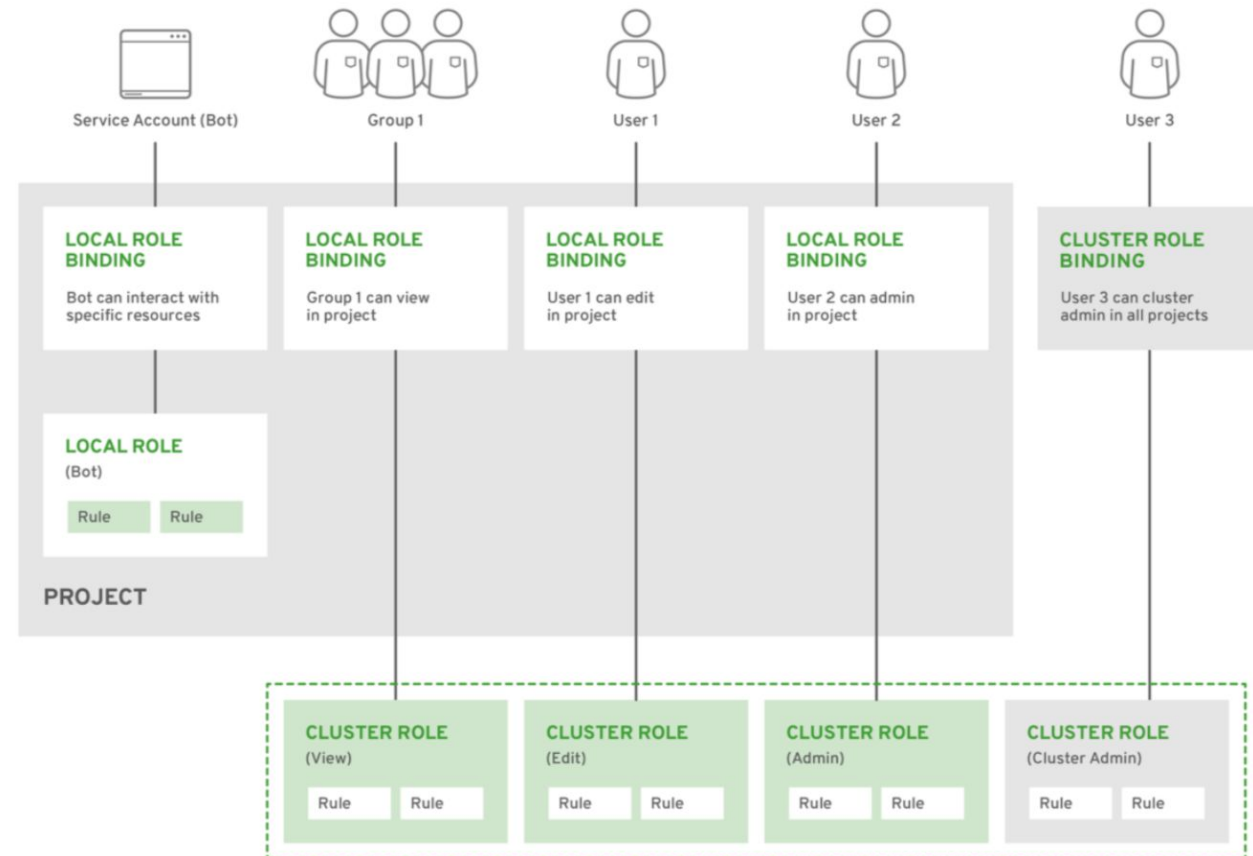
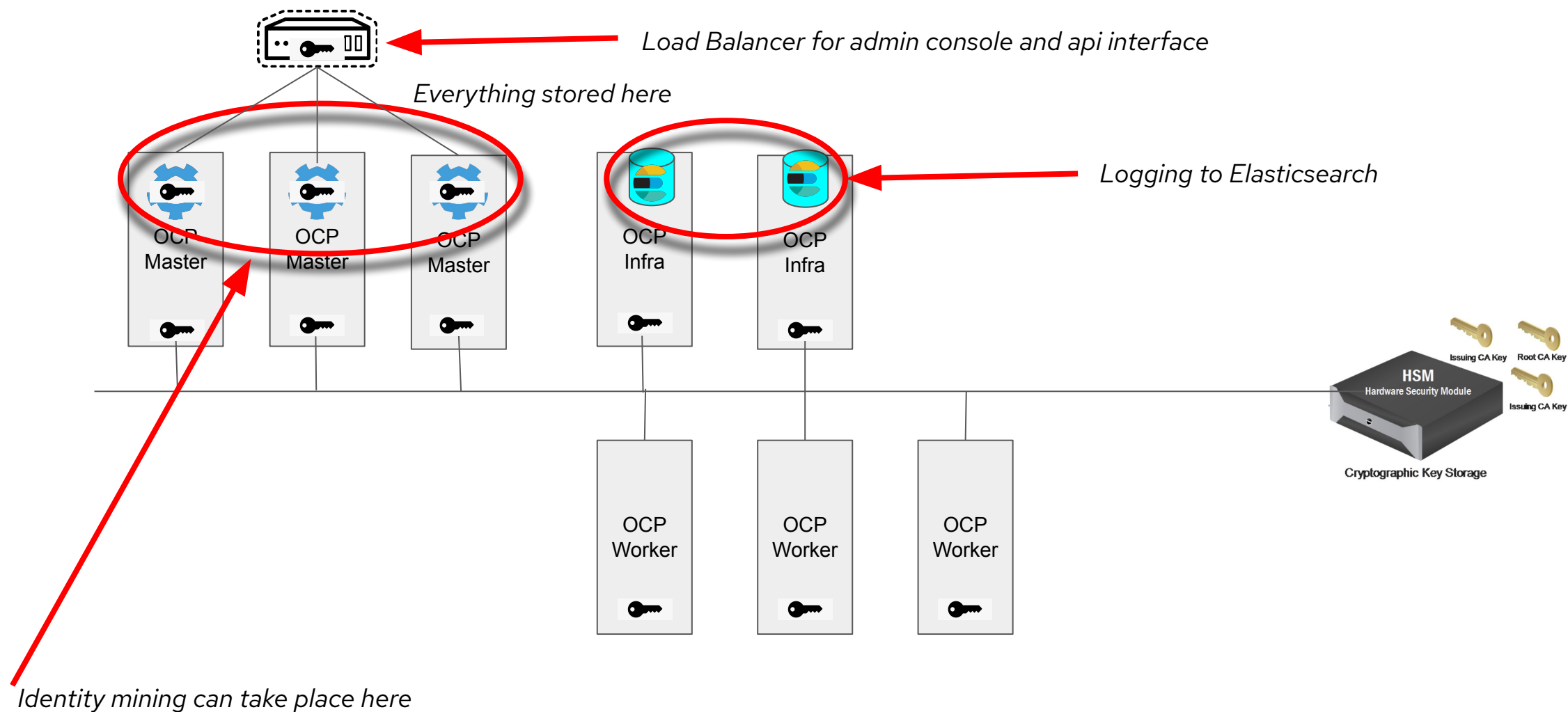
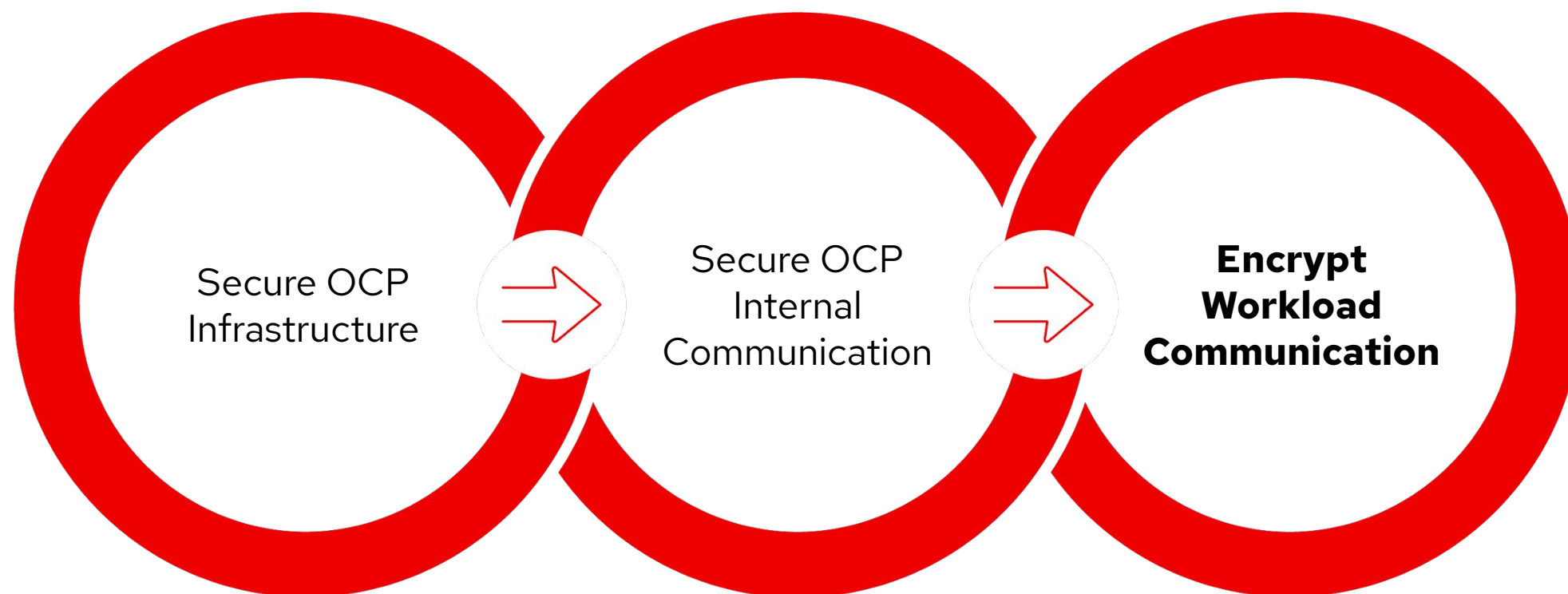


Figure 12 - Authorization Relationships

Identity Mining and SIAM Mining



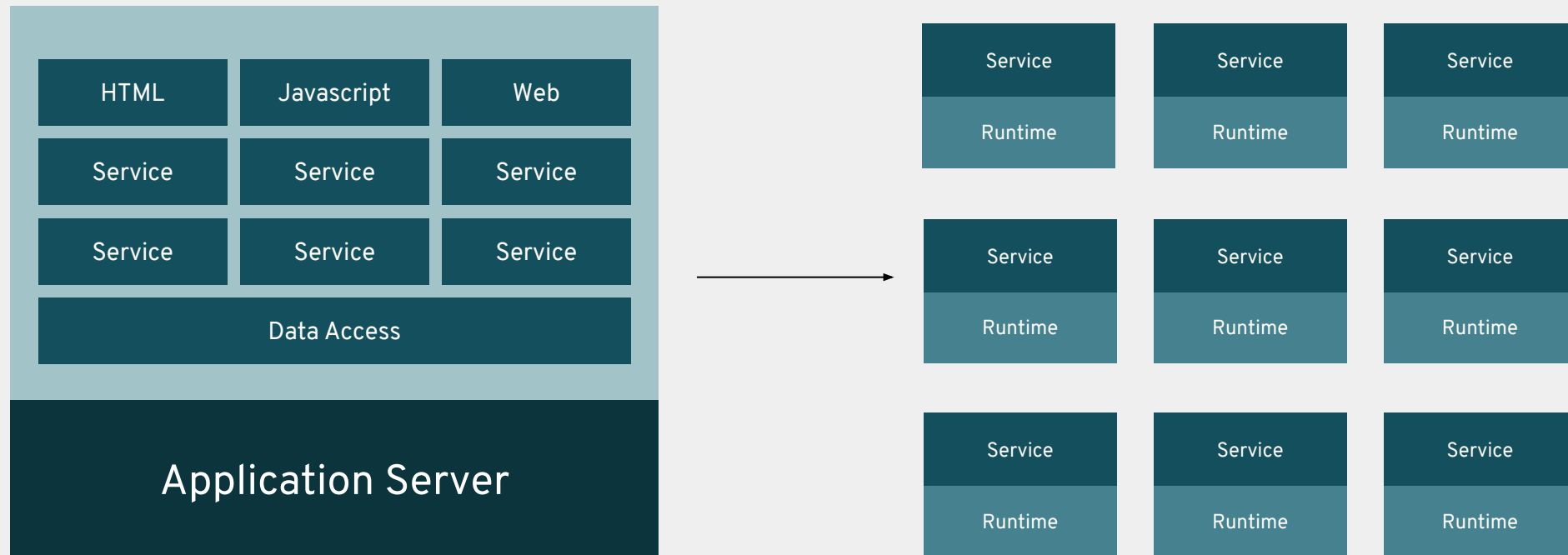
PKI in OpenShift



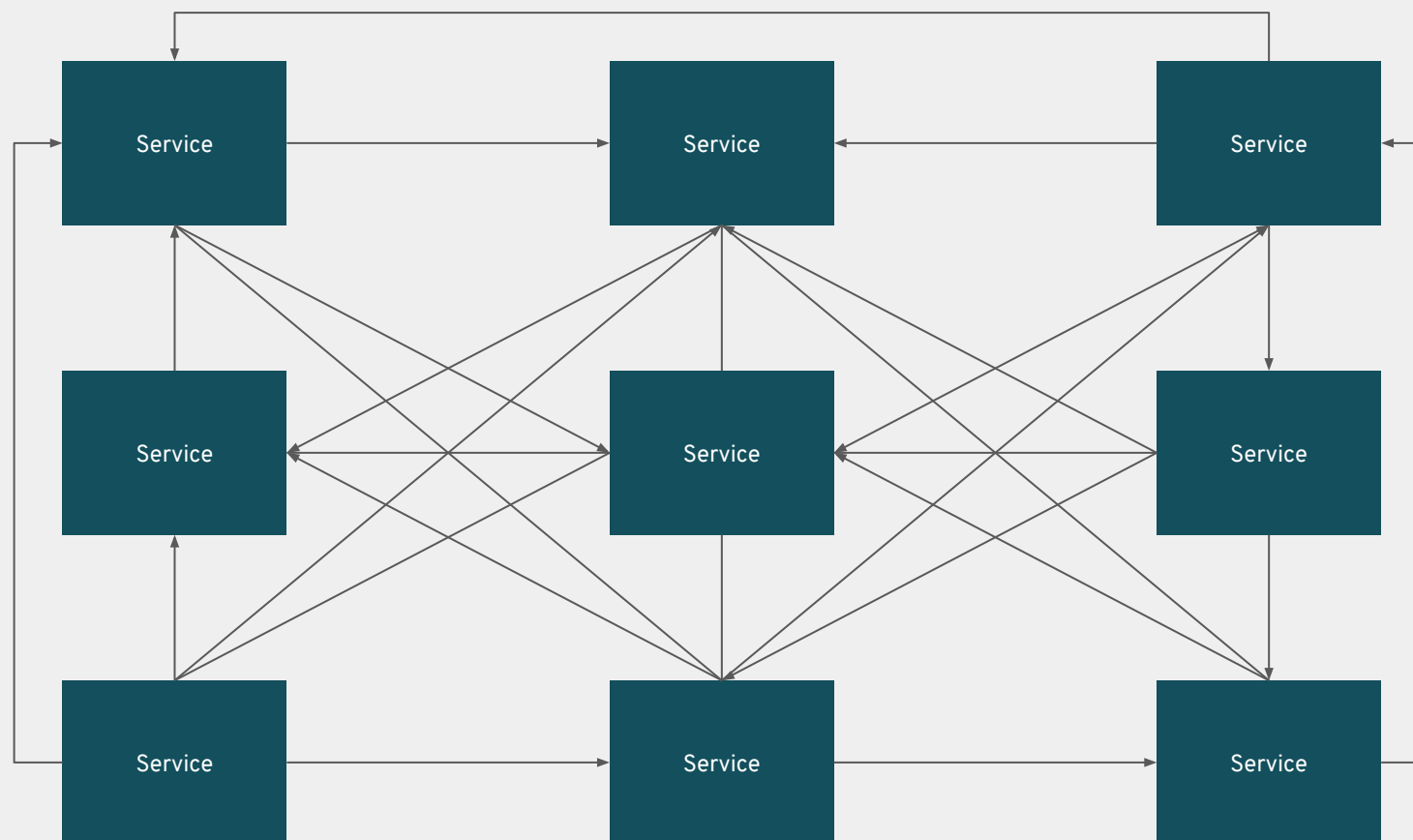
~~MICROSERVICES~~ ARCHITECTURE

CONFIDENTIAL designator

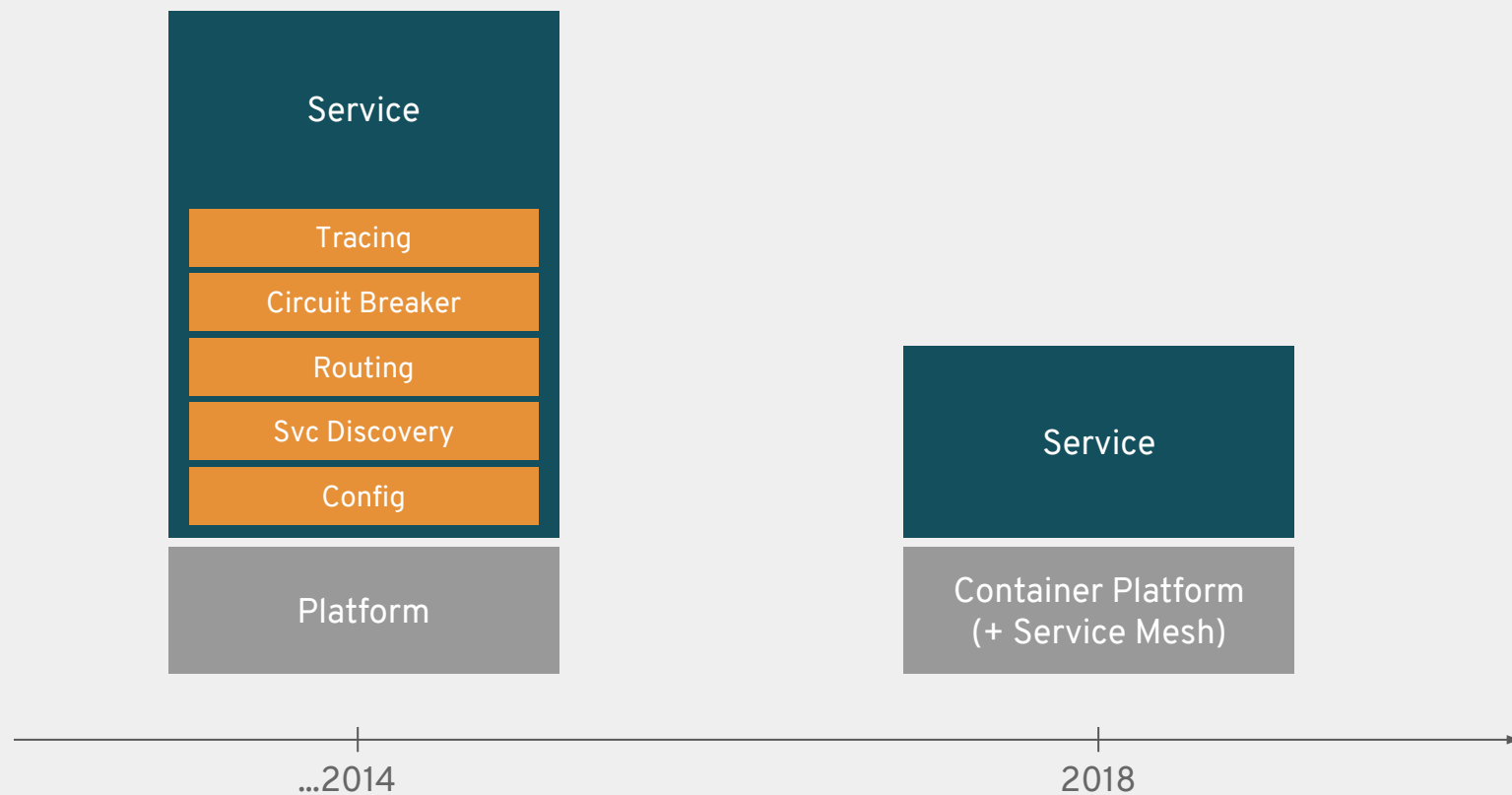
DISTRIBUTED



DISTRIBUTED ARCHITECTURE



A better way with a service mesh

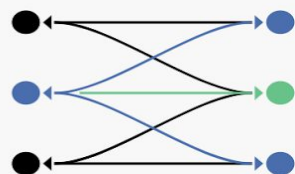


A service mesh provides a **transparent** and **language-independent** network for connecting, observing, securing and controlling the connectivity between services.



Istio

Connect, secure, control, and observe services.



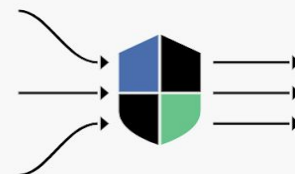
Connect

Intelligently control the flow of traffic and API calls between services, conduct a range of tests, and upgrade gradually with red/black deployments.



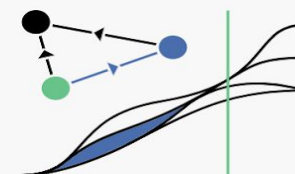
Secure

Automatically secure your services through managed authentication, authorization, and encryption of communication between services.



Control

Apply policies and ensure that they're enforced, and that resources are fairly distributed among consumers.



Observe

See what's happening with rich automatic tracing, monitoring, and logging of all your services.

ISTIO'S CAPABILITIES AT 10,000 FEET

Traffic Management.

Rules and traffic routing lets you control the flow of traffic and API calls between services.

Service Identity and Security.

Enforce consistently across diverse protocols and runtimes with little or no application changes.

Policy Enforcement.

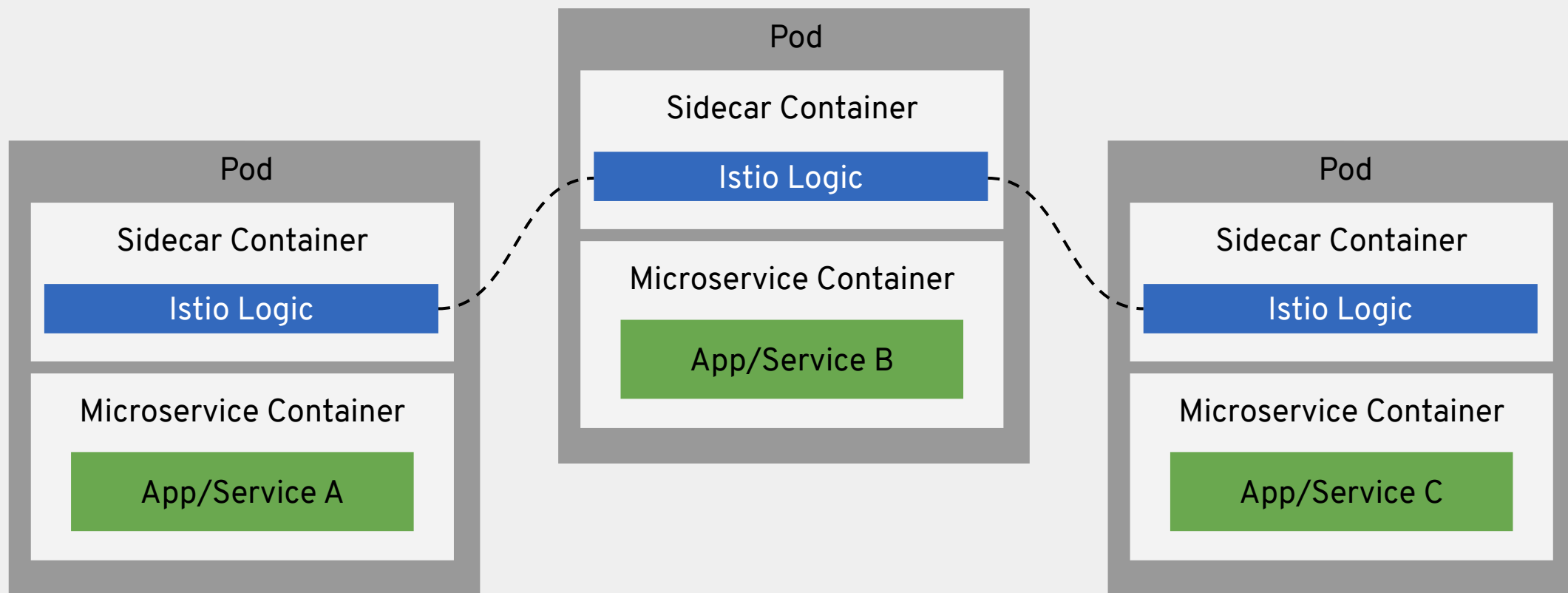
Apply to the interaction between services and ensure they are enforced. Changes are made by configuring the mesh, not by changing application code.

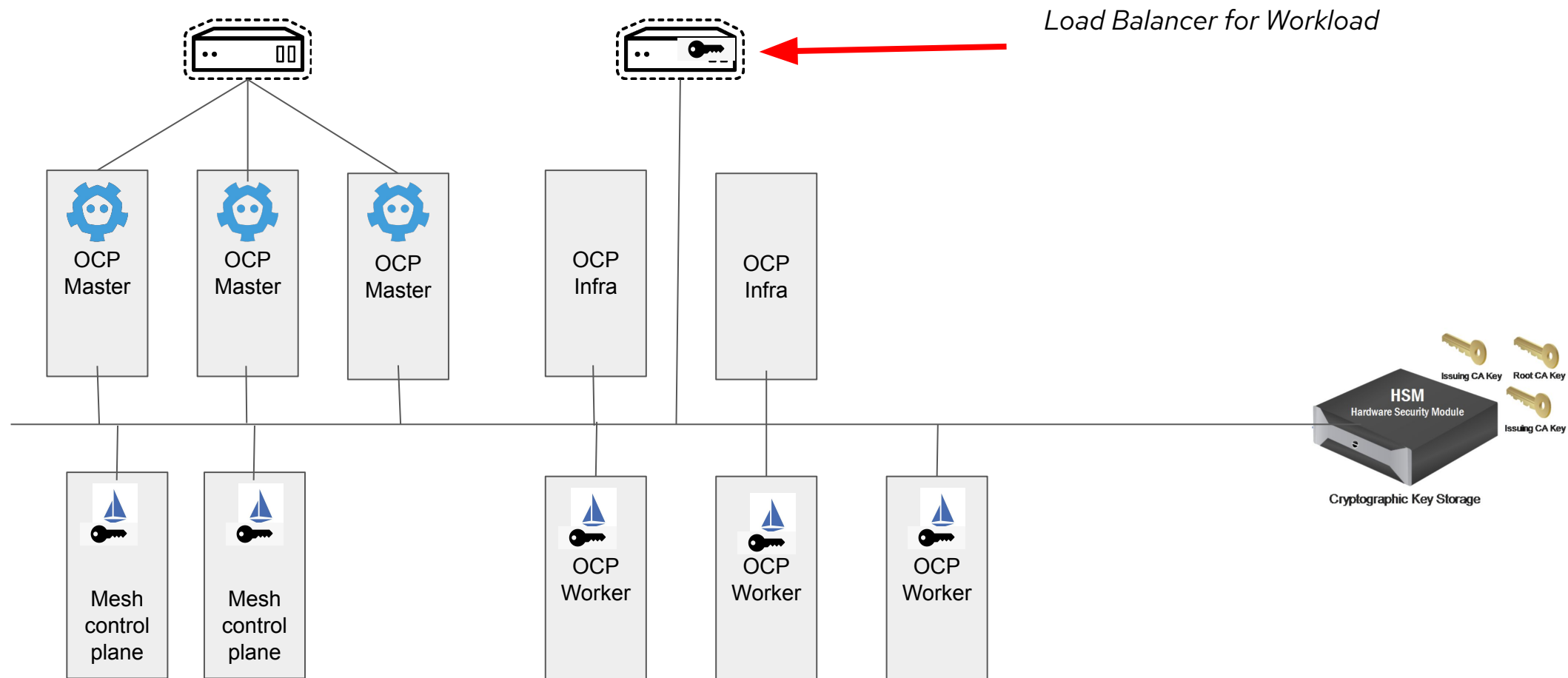
Observability.

Gain understanding of the dependencies between services and the nature and flow of traffic between them, providing the ability to quickly identify and fix issues.

MICROSERVICES WITH ISTIO

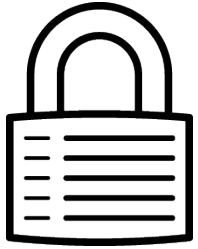
connect, manage, and secure microservices transparently





Stronger Platform Security

Defense in Depth



CONTROL
Application Security



DEFEND
Infrastructure



EXTEND

- [FIPS Compliance](#)
- [Encrypt etcd datastore](#)
- [RHEL CoreOS network bound disk encryption](#)
- [Private clusters with existing VPN / VPC](#)
- [Internal ingress controller](#)
- [Ingress Cipher & TLS Policy Configuration](#)
- [Log forwarding \(tech preview\)](#)

Thank you

Red Hat is the world's leading provider of enterprise open source software solutions. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500.

 linkedin.com/company/red-hat

 youtube.com/user/RedHatVideos

 facebook.com/redhatinc

 twitter.com/RedHat