TrilioVault for Kubernetes

Introduction, Architecture, Functionality, Use Cases, Compatibility

Prashanto Kochavara – Director of Product, Kubernetes
OpenShift Commons Briefing

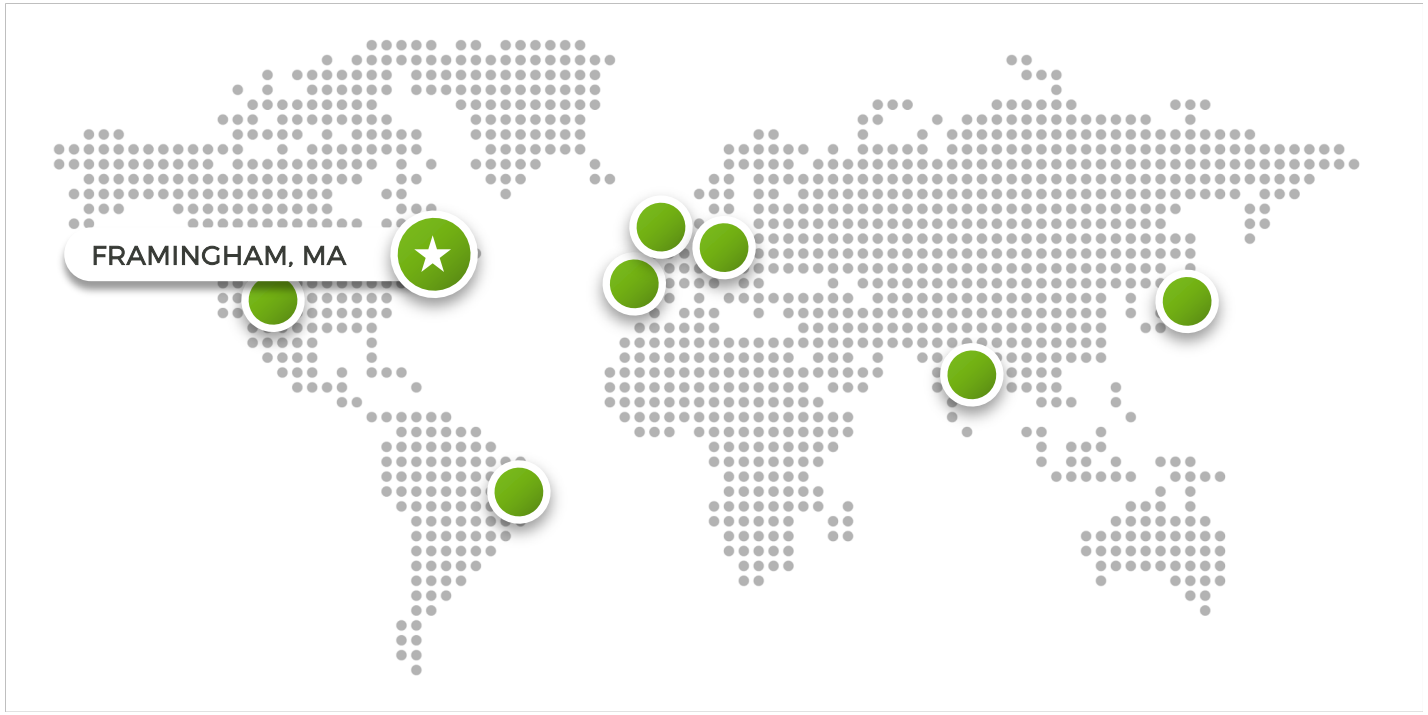
Agenda

- Company Overview
- Customer Problem – Legacy vs. Cloud-native applications
- TrilioVault for Kubernetes Overview
- TrilioVault for Kubernetes Technical Details
- Demo
- Summary

Company Overview

Overview

- Founded in 2013
- Leading platform for cloud-native data protection
- Enterprise customers across defense, automotive, telco, MSPs and more
- Backed by leading VCs and tech luminaries
- Patented technology
- Global Partner Ecosystem
- Red Hat Certified Partner



GLOBAL CUSTOMER BASE

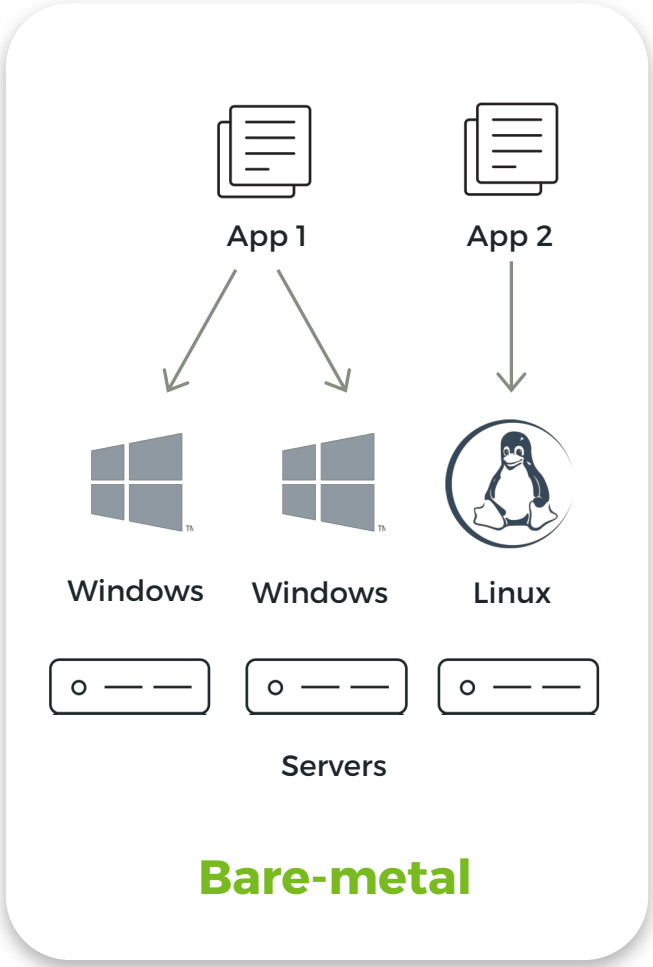
PARTNERSHIPS

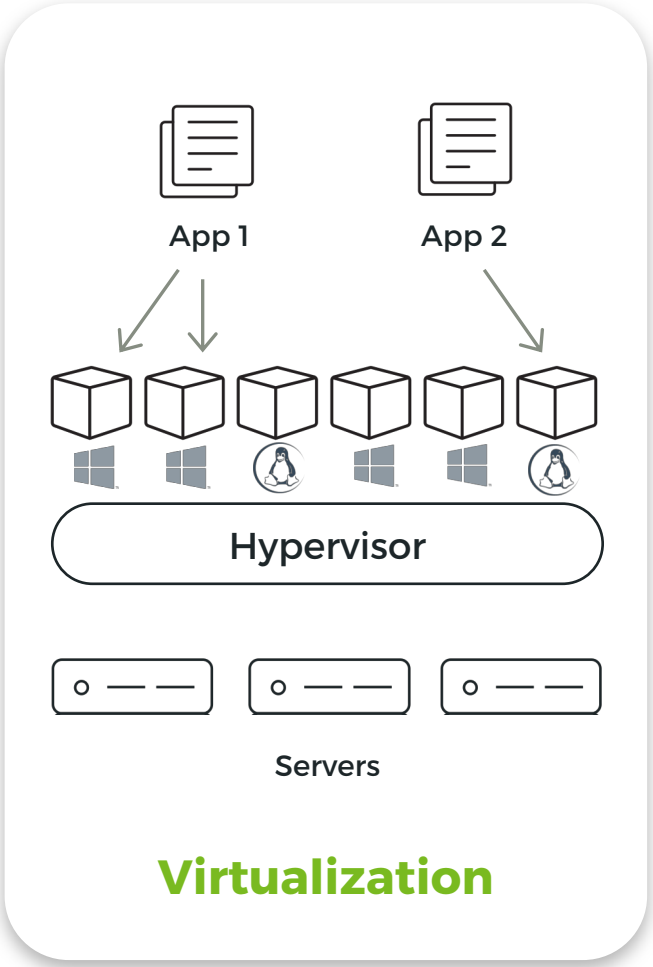
PLATFORM SUPPORT

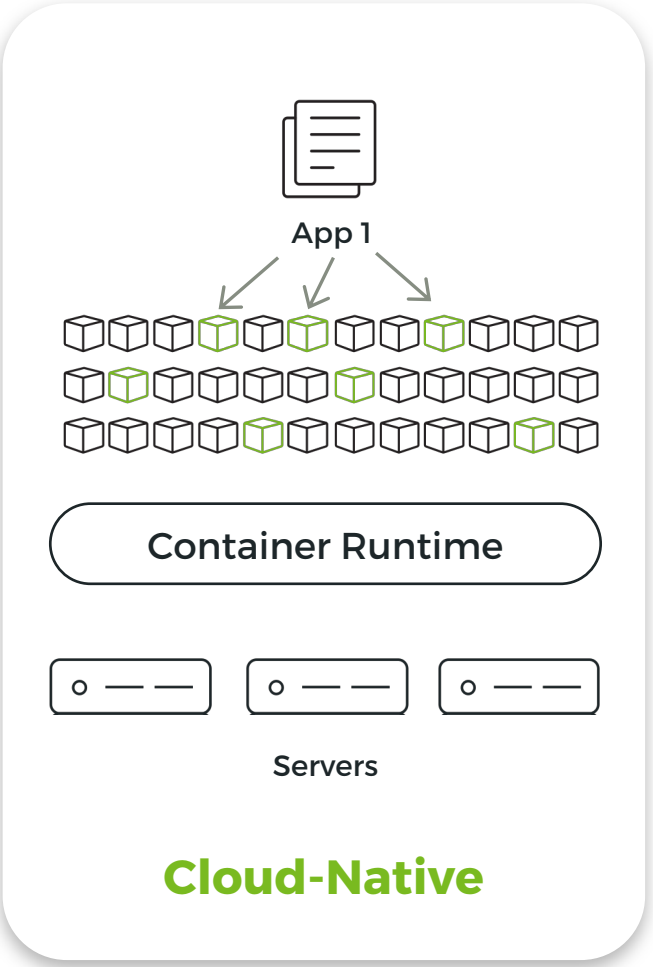
TRILIO

Customer Challenge

Evolution of Applications







Traditional Data Protection Hinders Cloud-Native Apps

Traditional Data Protection Technology

- Disparate technologies
- Application OS dependence
- Silo'd, Monolithic applications
- Rarely changing footprint
- Focused on storage/data volumes not application metadata-layout topology, service, route etc
- Building the app for IT admins versus DevOps

≠

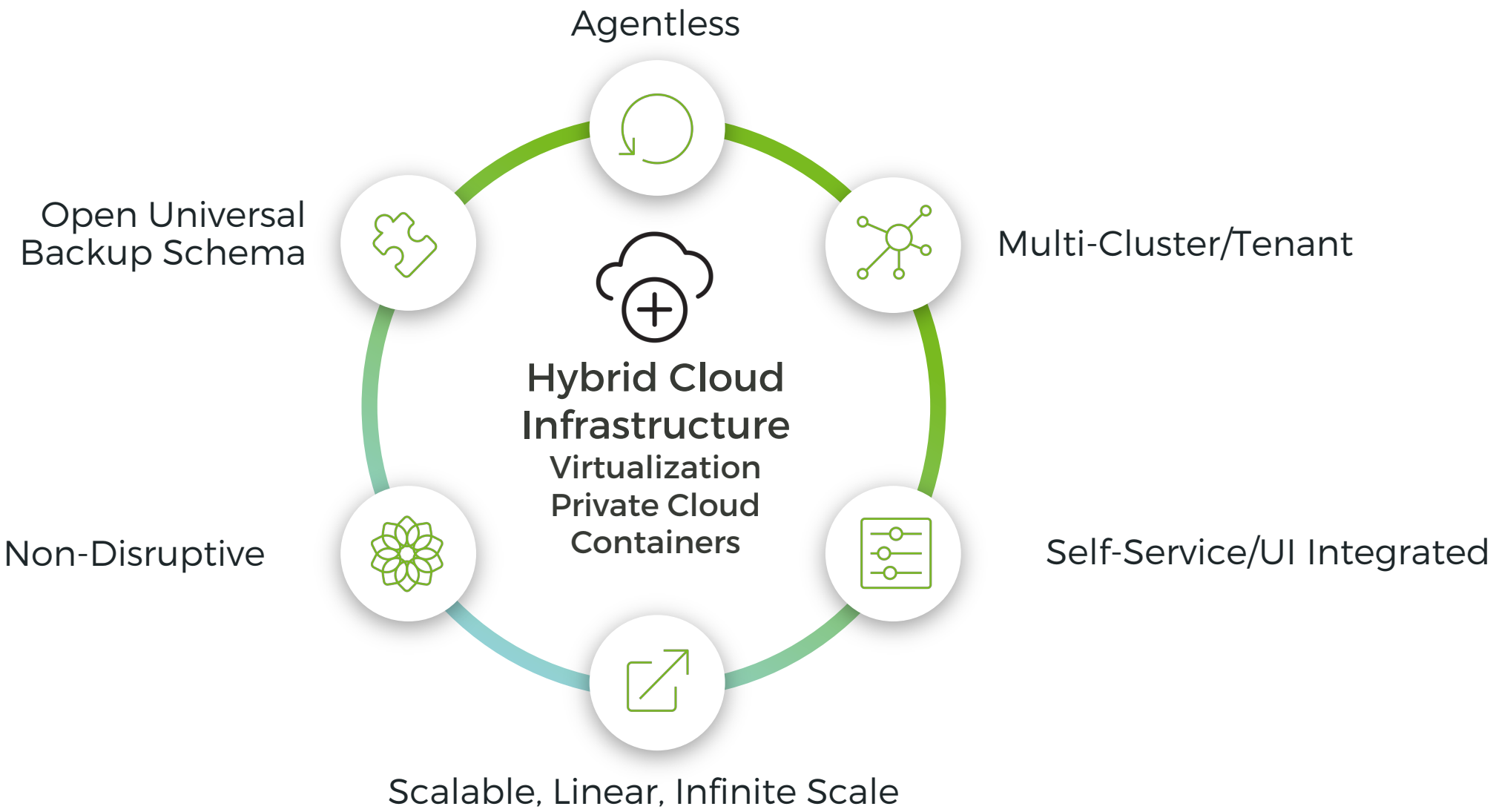
Cloud-Native Applications

- New approach to building and deploying applications
- Modular and Micro-services oriented
- Highly available
- Highly scalable
- New languages and frameworks
- Highly automated and API driven
- Policy-driven and multi-tenant

TRILIO

Introducing TrilioVault for Kubernetes

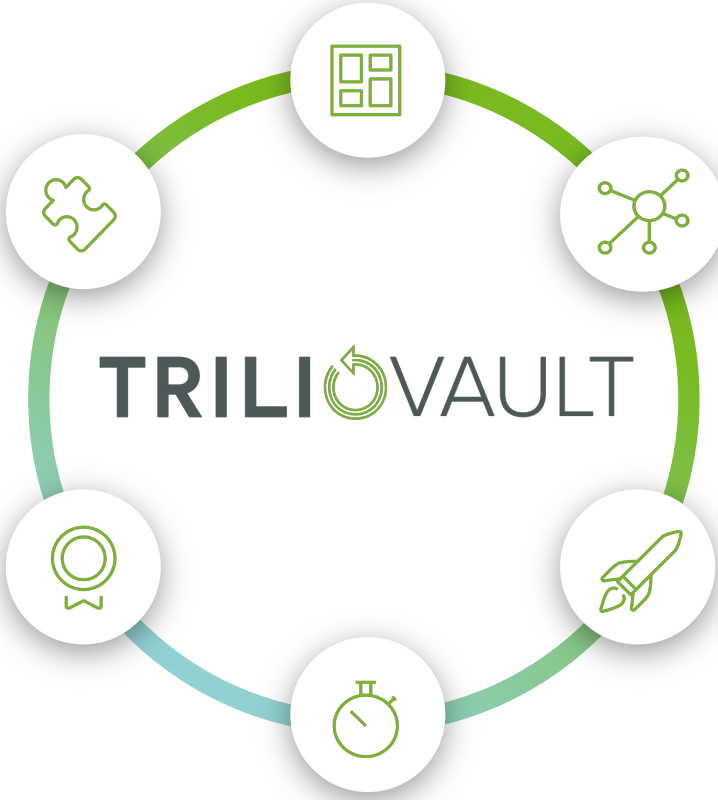
TrilioVault Data Protection: Designed for Cloud-Native Applications



TrilioVault for Kubernetes: Key Attributes

Application-Centric
Protects data, metadata and Kubernetes objects

Infrastructure Compatibility
Compatible with any Storage (CSI, NFS, S3), or any Cloud (Private, AWS, GCP, Azure, IBM)



Native to Kubernetes & OpenShift
Packaged as an Operator, Deployed as a CRD, and API with all facilities (CSI)

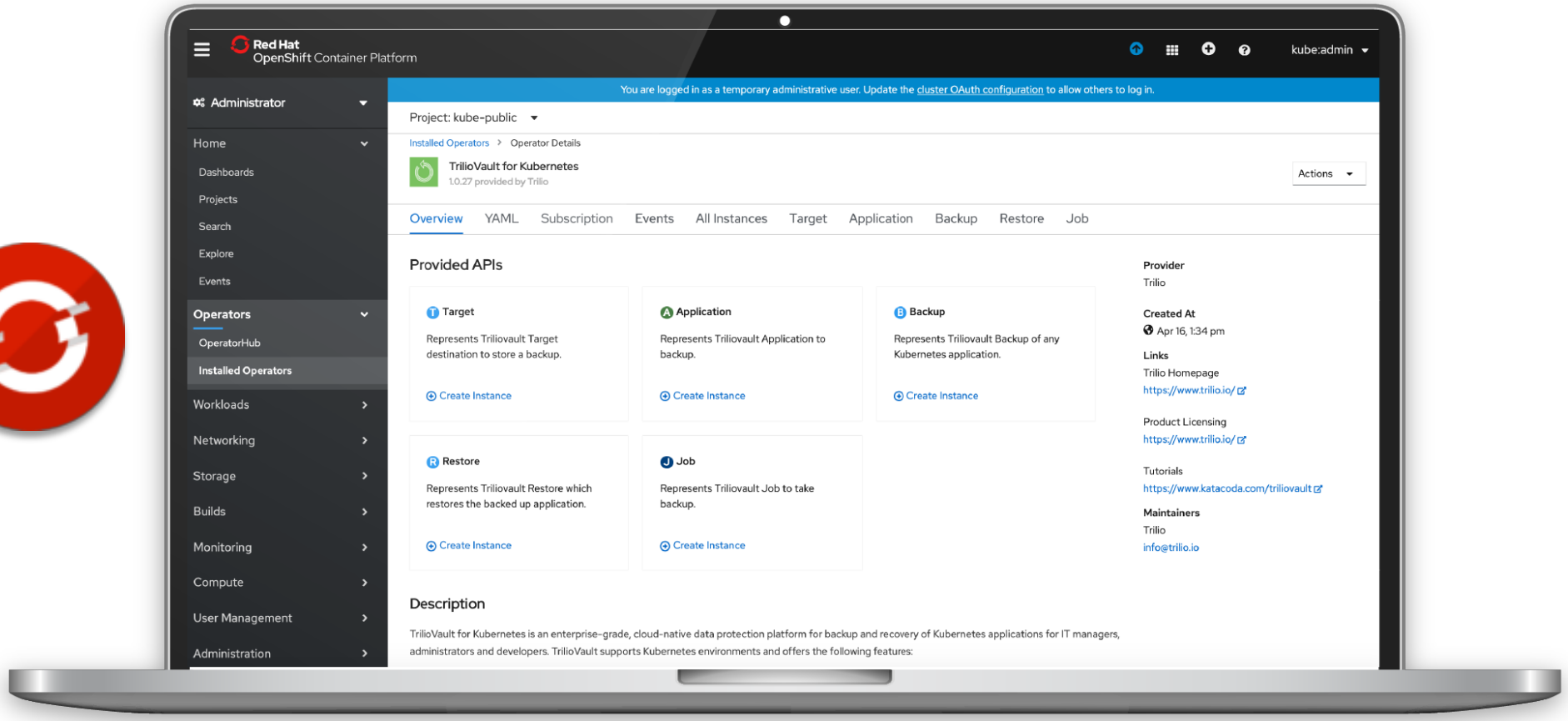
Certified Technology
Red Hat OpenShift Operator
IBM Cloud Pak for Data & Multicloud Management

Application Deployment & Ecosystem Tooling
Helm, Operators, Labels, Prometheus, Fluentd, Grafana

Easy to Try and Install
Demos and Test Drive scenarios | 30-Day Free Trial
Basic Edition Up to 10 Nodes | Enterprise Edition with Premium Support

TRILIO

TrilioVault for Kubernetes within Red Hat OpenShift



Packaging & Accessibility

Upstream Operator

- Helm v2 and v3 Support
- Deploys TrilioVault for Kubernetes CRD
- Helm Repository / GitHub

Operator

- OLM - UBI based operator
- Deploys TrilioVault for Kubernetes CRD
- Embedded OperatorHub, OperatorHub.io

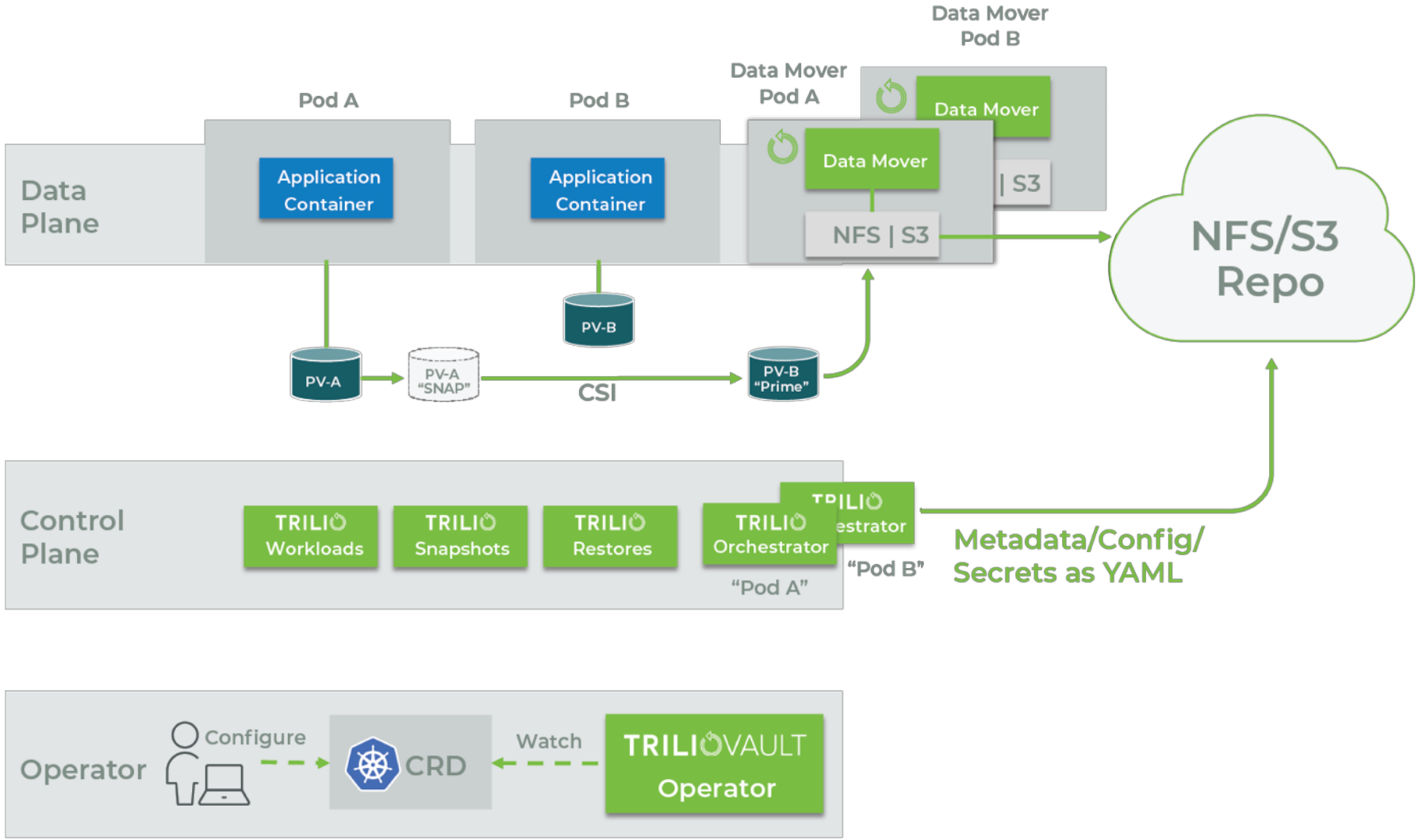
Architecture

Data Plane

- Mounts PV's
- Calculates Incremental changes
- Writes QCOW2 to backup target

- Manages all CRD's and Trilio controllers
- Captures application metadata – deployments/secrets/ configmaps/services/routes

- BackupPlans, Target, BU, Restore

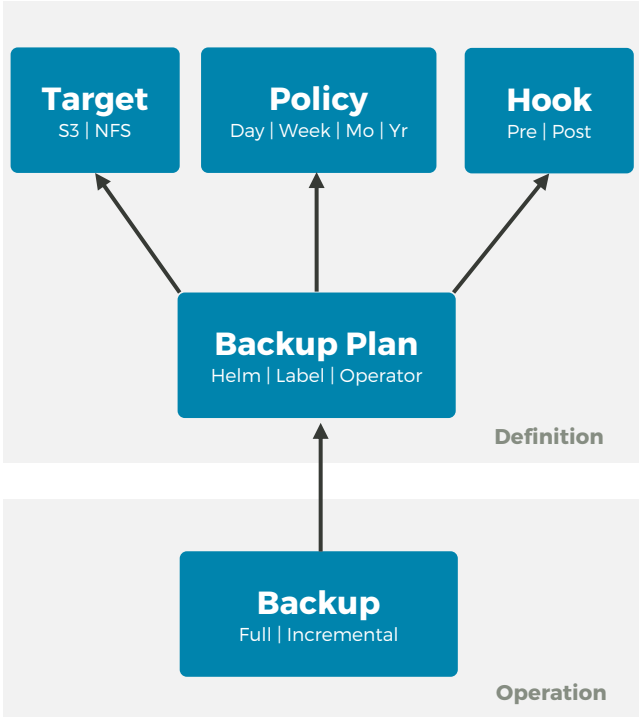


Custom Resource Definitions

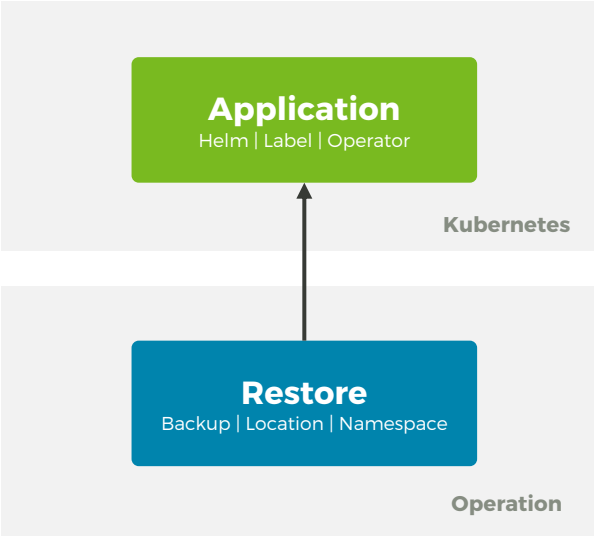
TrilioVault for Kubernetes provides the following CRDs:

- **Target** – Designated NFS | S3 location to store backups
- **Policy** - Backup Frequency and Backup Retention
- **Hook** – Pre and Post triggers to execute commands inside containers
- **BackupPlan** – Kube Application, Target, Policy, Hook
- **Backup** - Performs the TrilioVault backup operation
- **Restore** - Performs the TrilioVault restore operation

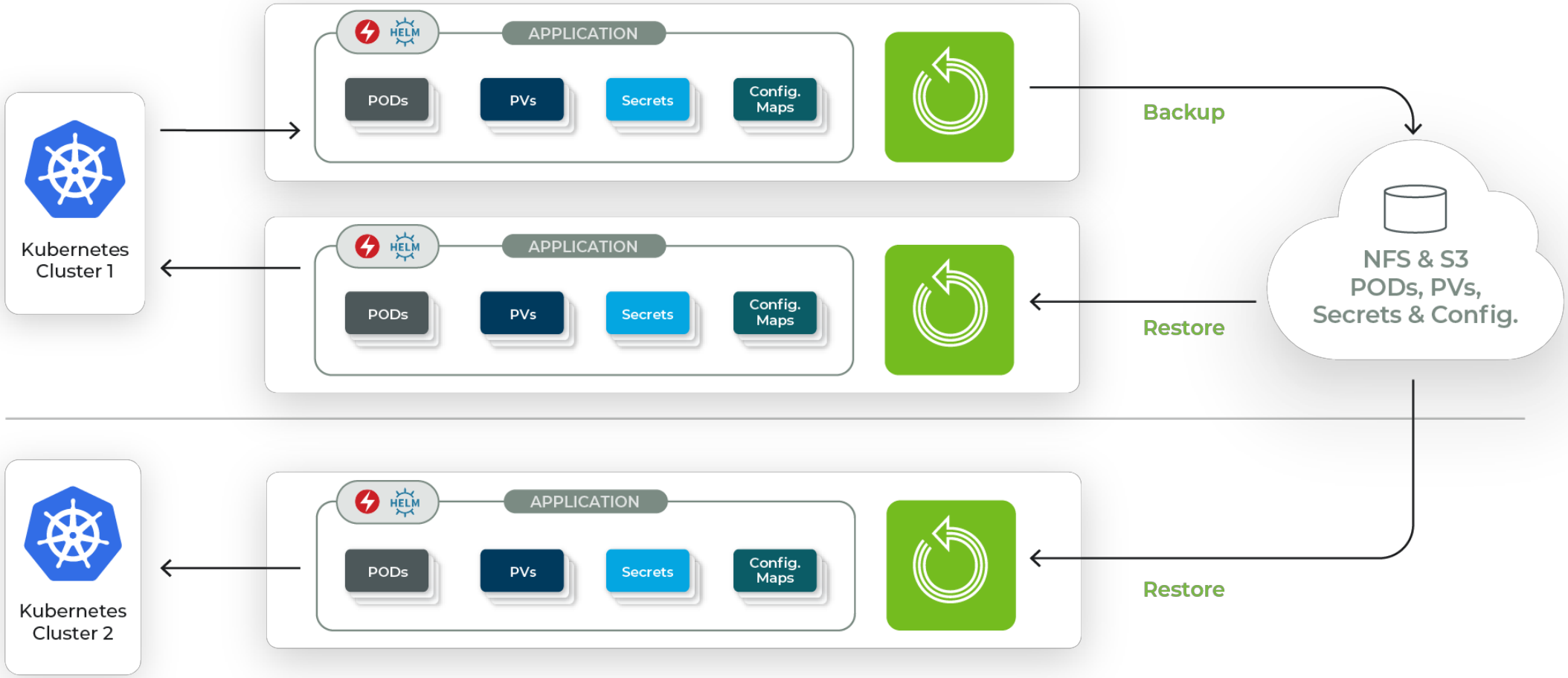
Backup Process



Restore Process



TrilioVault Protection and Recovery



What does TrilioVault for Kubernetes Backup?

Custom Labels

- Backup spec of all the resources
- All resources that comprise the application (PODs, PVs, Config Maps, Secrets)
- PV's used by the application/resources and take backup of the PV’s

Helm

- Backup all the revisions including the deployed revision of the release given by user.
- Parse the chart and identify all the PV's and backup data for those PV’s

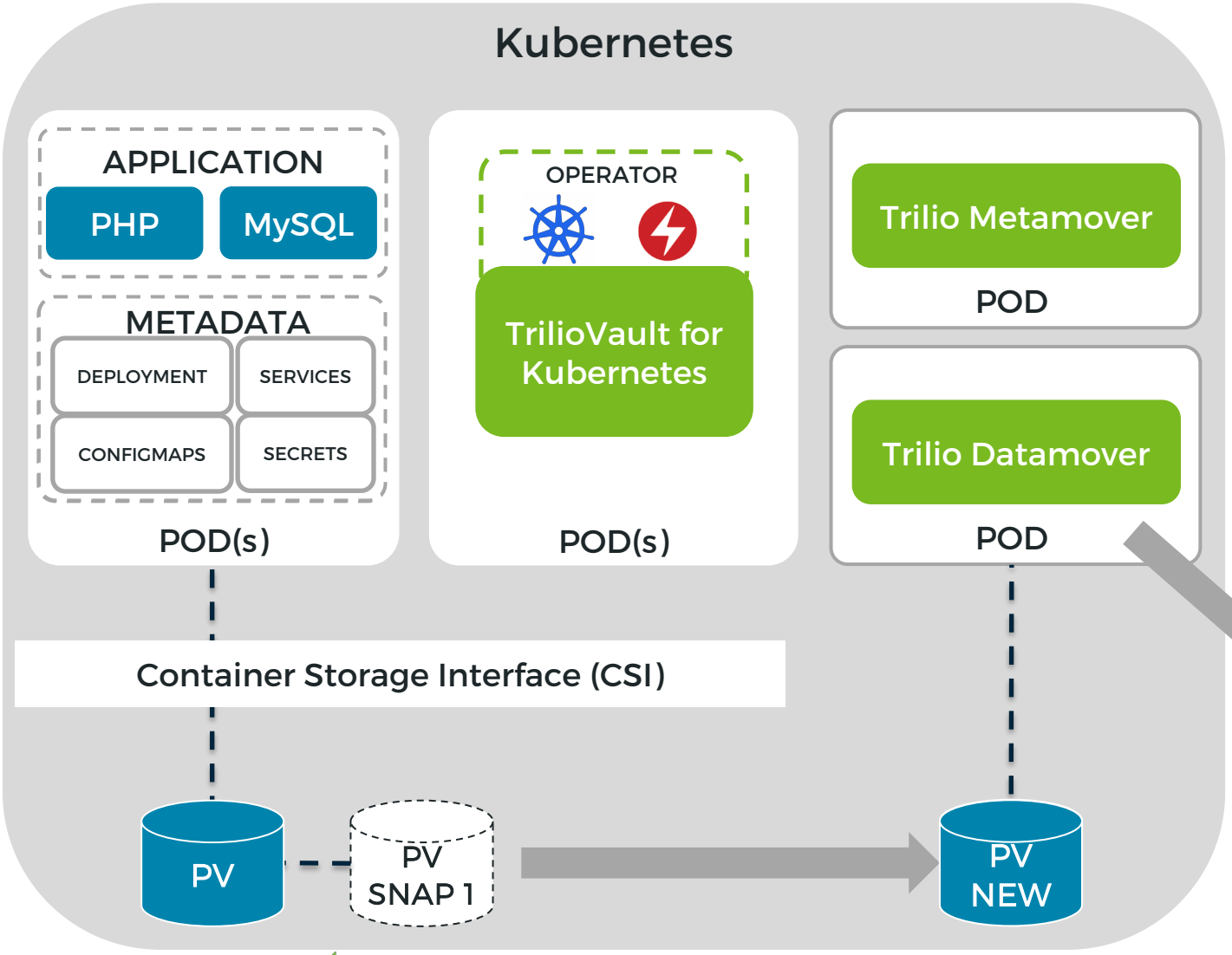
Operator

- Backup the resources of the Operator
- Backup Operator CR created by user
- Parse resources and backup PVs used by the Application (managed by Operator)

16

TrilioVault Backup

Full Backup

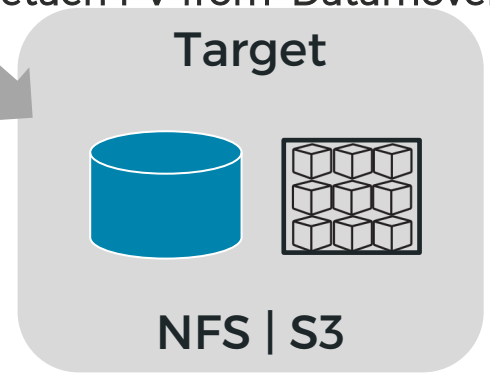


Metadata Backup

1. '*Metamover Pod*' spawned for metadata backup
2. Kube Application metadata backup to target (JSON)

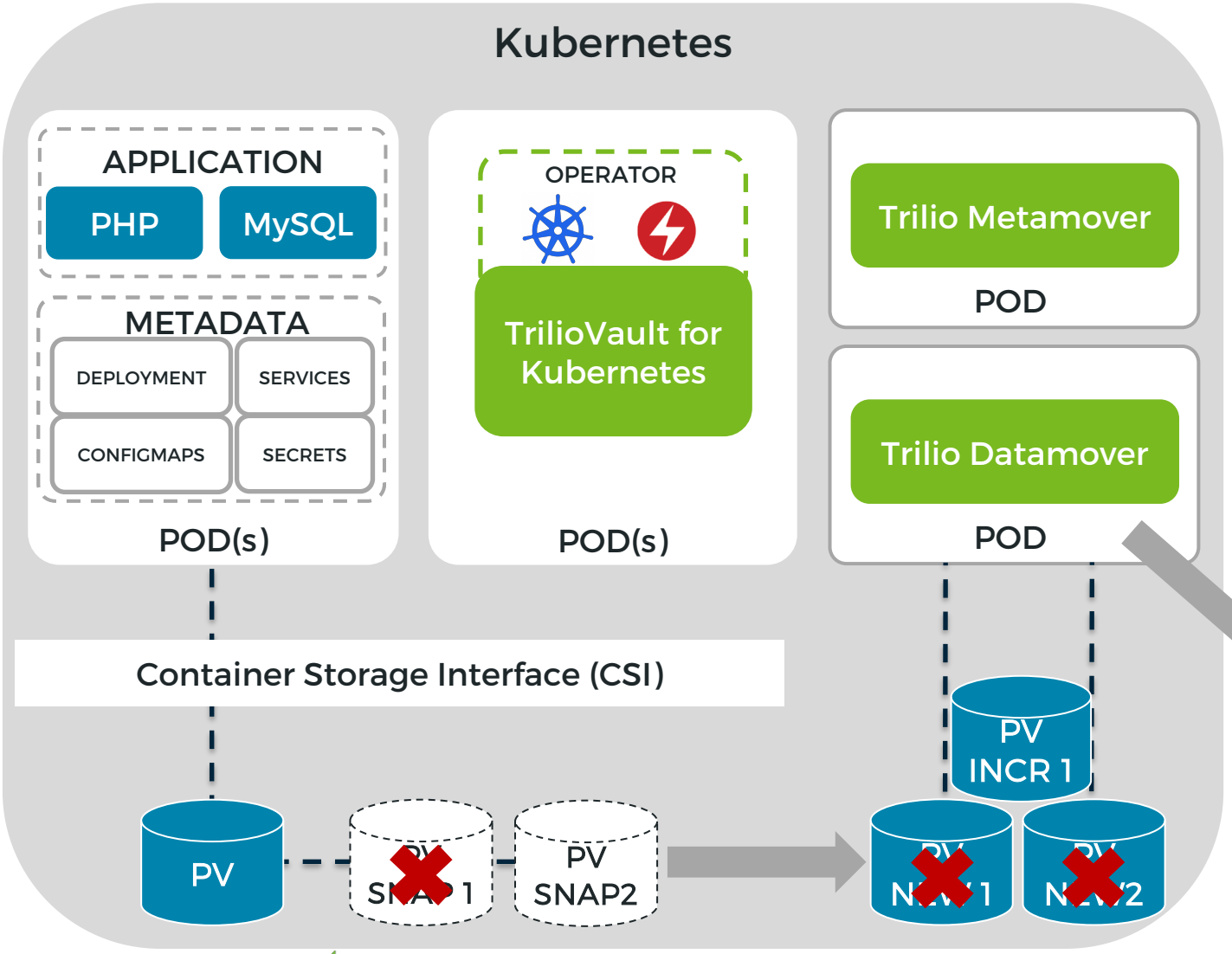
Data Backup

3. '*Datamover Pod*' spawned for "<u>each</u>" Persistent Volume
4. Create CSI PV Snapshot(s) – '*PV Snap 1*'
5. Create CSI Volume from snapshot
6. Attach PV to '*Datamover Pod*'
7. '*qemu-img convert*' to QCOW2 format on target
8. Detach PV from 'Datamover Pod' and Delete PV



TRILIO

Incremental Backup

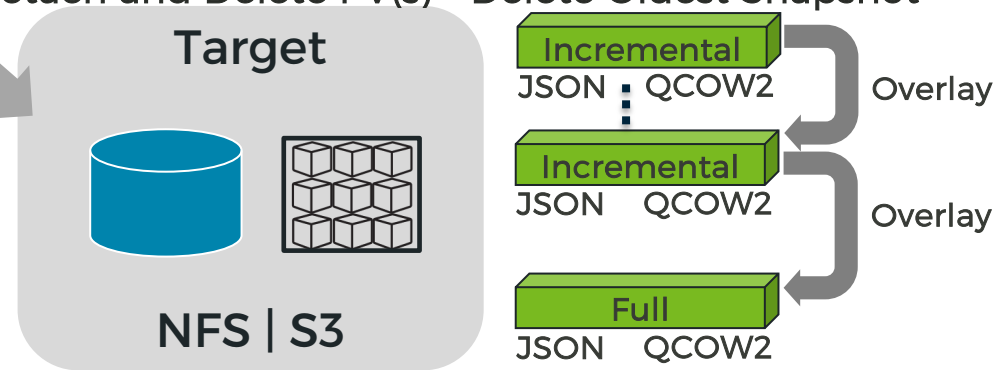


Metadata Backup

1. '*Metamover Pod*' spawned for metadata backup
2. Kube Application metadata backup to target

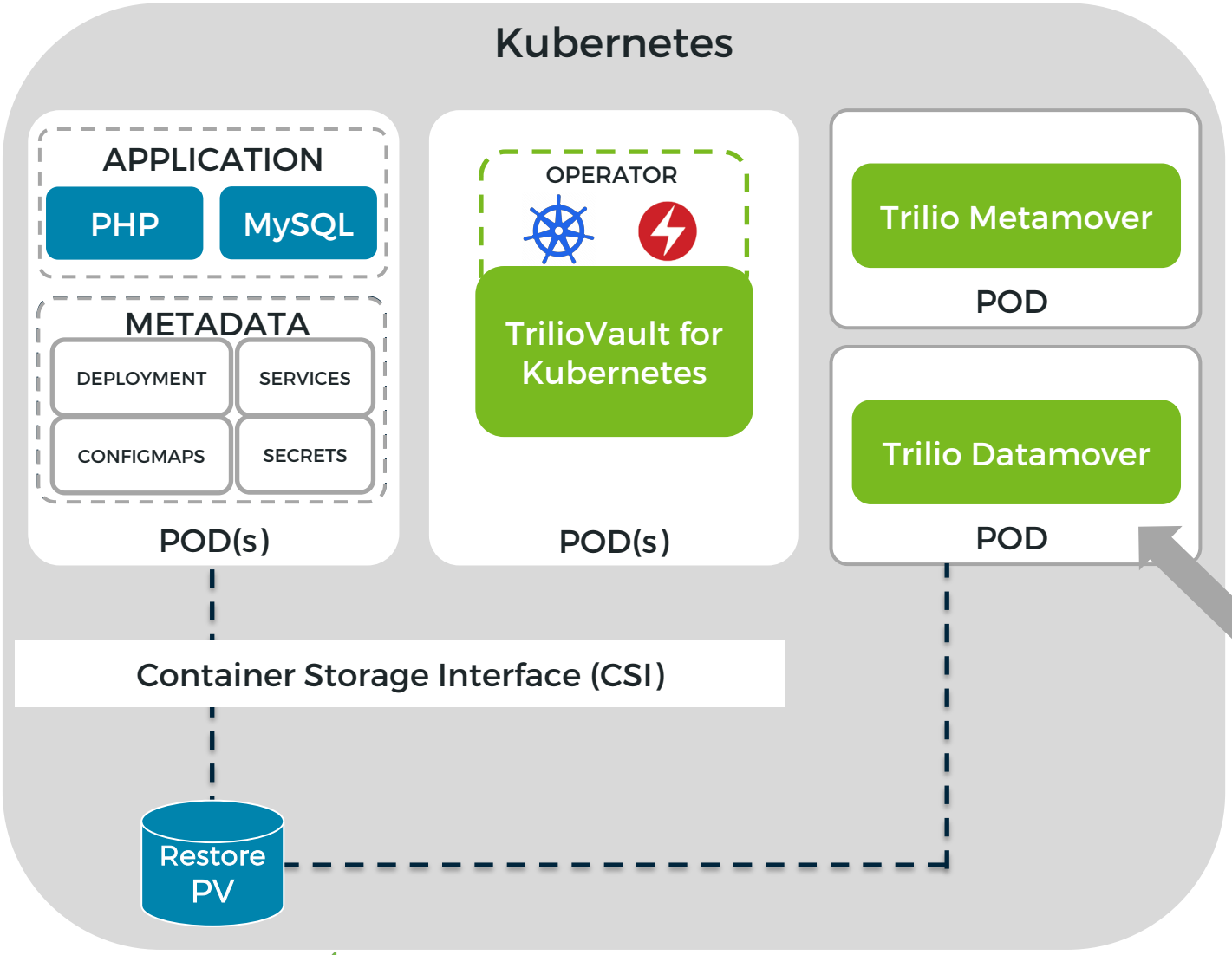
Data Backup

3. '*Datamover Pod*' spawned for "<u>each</u>" Persistent Volume
4. Create CSI PV Snapshot(s) – *'PV Snap 2'*
5. Create CSI Volume(s) - *'PV New 1' & 'PV New2'*
6. Attach PV(s) to '*Datamover Pod*'
7. *'qemu-img convert'* to QCOW2 format on target
8. Detach and Delete PV(s) – Delete Oldest Snapshot



TrilioVault Restore

Restore

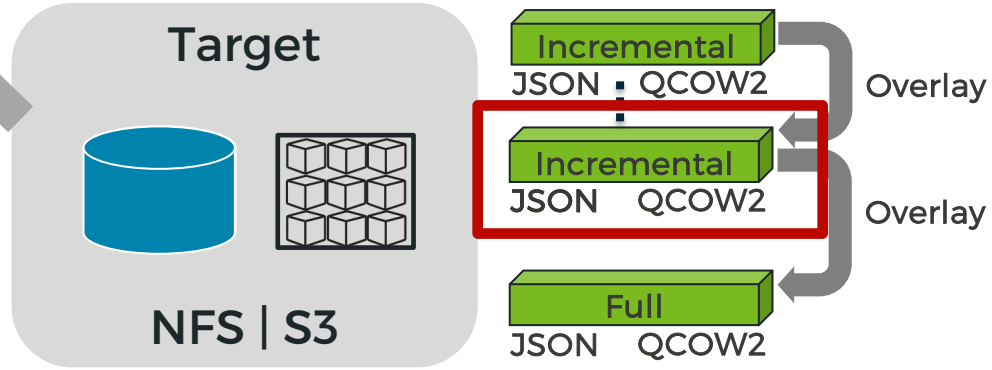


Data Restore

1. *'Datamover Pod'* spawned for "each" Persistent Volume
2. Create Restore CSI Volume(s) - *'Restore PV'*
3. Attach Restore PV(s) to *'Datamover Pod'*
4. *'qemu-img convert'* to *'Restore PV'*
5. Detach PV(s) from 'Datamover Pod'

Metadata Restore

6. *'Metamover Pod'* spawned for metadata backup
7. Application metadata restored to Kubernetes
8. Application Specs - PVC to Restored PV(s)



Use Cases

Backup and Recovery

- Schedule and on-demand jobs
- Full and Incremental backups
- Cluster or Namespace Level Restore
- Full or Selective App Restores

Disaster Recovery

- Build Disaster Recovery strategies leveraging Trilio
- Recover applications to any Kubernetes cluster
- On-prem or Public Cloud

Application Mobility

- Accelerate CI/CD by building multiple Test/Dev environments leveraging Trilio
- Move to a new Kubernetes cluster

Migration

- Kubernetes Distributions
- Kubernetes Versions
- On-prem, Public Cloud, or Hybrid Cloud

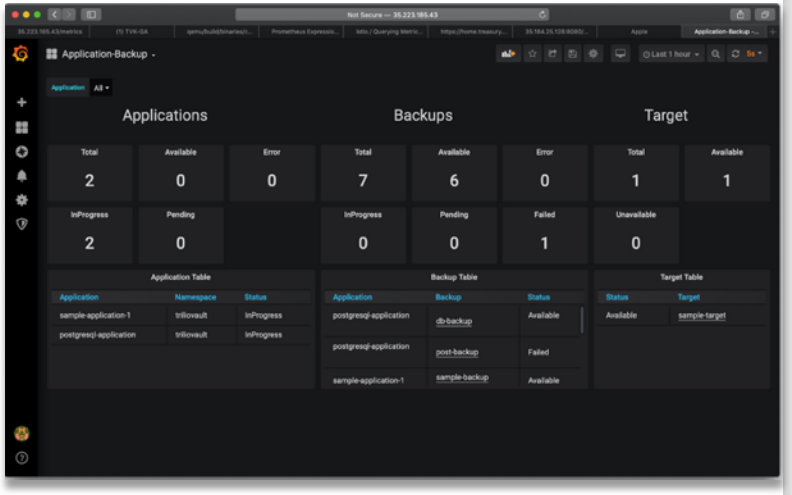
TRILIO

Monitoring & Logging

TrilioVault Monitoring & Logging

Prometheus & Grafana

- All metrics sent to Kubernetes Prometheus
- Visualization through Grafana
- Pre-canned and custom dashboards available in Grafana



FluentD

- TrilioVault integrated to send all logs to FluentD for intelligent log management

Prometheus Metrics Exporter - Integration

TrilioVault Objects	Description
trilio_backupplan_info	Details on BackupPlan CRDs
trilio_target_info	Details on Target CRDs
trilio_backup_info	Details on Backup CRDs
trilio_backup_status_percentage	Details on Backups and Percentage Complete
trilio_backup_completed_duration	Details on Backup Durations (Status – Available)
trilio_backup_storage	Details on Backup Storage
trilio_restore_info	Details on Restore CRDs
trilio_restore_status_percentage	Details on Restores and Percentage Complete
trilio_restore_completed_duration	Details on Backup Durations (Status = Completed)
trilio_<controller>_available	Availability for Trilio Controllers

Security & Permissions

TrilioVault for Kubernetes – Security

Admin Access

- Trilio does not require/depend on Kubernetes cluster or namespace admin roles

Security Context Constraints (SCC)

- Leverages existing/default SCC's provided (Trilio does not create any new SCC)

Service Accounts

- Trilio creates own service accounts and does not leverage default service accounts
- Service accounts for TVK operations created/deleted during runtime.

Permissions and Capabilities

- Own set of permissions and capabilities
- Complete details are provided within TVK's User documentation (docs.trilio.io)

Security Product Definitions

- Security section dedicated within User Documentation – coming soon.

DEMO – Backup Helm App on OpenShift

Compatibility & Support

Compatibility

CERTIFIED KUBERNETES DISTRIBUTION

RED HAT OPENSHIFT

Prod: 4.4+
Test/Dev: 4.1 - 4.3

Kubernetes

Prod: 1.17+
Test/Dev: 1.12-1.16

STORAGE

CONTAINER STORAGE INTERFACE

TARGET

Amazon S3

S3 Compatible Storage

NFS

ECOSYSTEM TOOLING

Grafana

Prometheus

fluentd

ANY CLOUD

Google Cloud

aws

Azure

IBM Cloud

On-Prem

Summary

Summary

- Traditional Approaches do not satisfy cloud-native application requirements
- TrilioVault is purpose-built as an Operator to protect cloud-native applications (helm/operator/label)
- Kubernetes native, Ecosystem fluidity and cutting-edge features are few of the pillars for TVK
- Enable a plethora of use cases to provide TCO savings, security and peace of mind
- Mature, robust copy/transfer protocols.

TRY IT NOW

Experience TrilioVault for yourself by visiting www.trilio.io today!

Watch
a Demo

Test Drive
Live Scenarios and Use Cases

Download a Free Trial
or Basic Edition

TRILIO

Thank You

prashanto.kochavara@trilio.io
@kochavara