

# Kubernetes Backup and Migration Strategies using Project Velero

## Kubernetes Application Migration



Tom Spoonemore  
Velero Product SME, VMware  
[tspoonemore@vmware.com](mailto:tspoonemore@vmware.com)



Steve Kriss  
Velero Technical Lead, VMware  
[krisss@vmware.com](mailto:krisss@vmware.com)

[@tunemore](#)

[@krisst16](#)

# Agenda

Kubernetes Backup, Restore, Migration Use Cases

Introduction to Velero

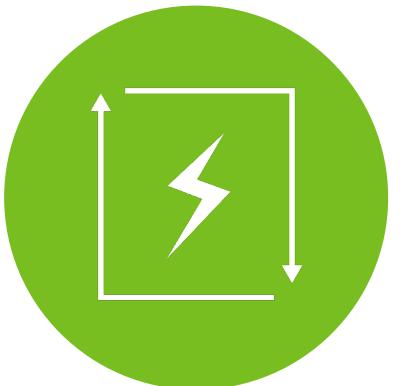
Strategies for Protecting Cloud Native Applications

Demo: Cloud Native Application Migration



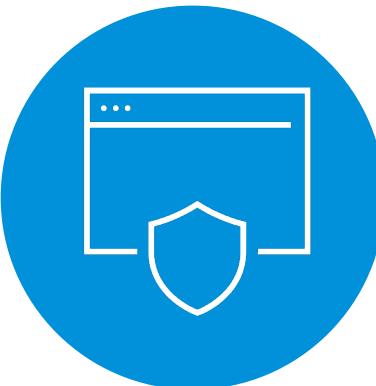
# kubernetes

Backup, Restore, Migrate Use Cases



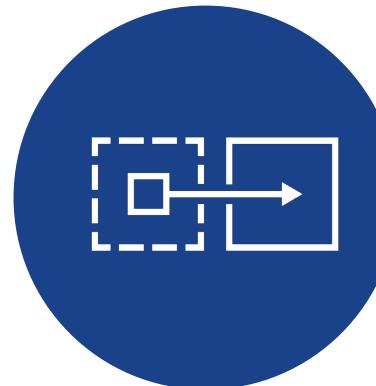
## Disaster Recovery

- Rebuild
- Restore
- Recover



## Data Protection

- Data Loss
- Data Corruption
- Archival Retention



## Data Migration

- Move from one platform to another
- Kubernetes Upgrades
- Hydrate dev, test, and staging environments



# VELERO

Velero is an open source tool to safely back up, recover, and migrate Kubernetes clusters and persistent volumes. It works both on premises and in public clouds.

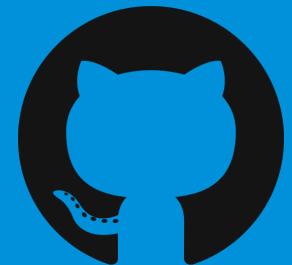
Built with focus on Kubernetes!

## Components

- Command line interface
- Kubernetes server application



Formerly Heptio Ark



[github.com/heptio/velero](https://github.com/heptio/velero)

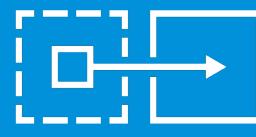
# Velero Use Case Benefits

## Disaster Recovery



Reduces time to recovery in case of infrastructure loss, data corruption, and/or service outages

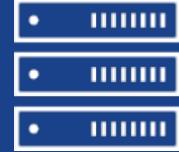
## Data Migration



Enables cluster portability by easily migrating Kubernetes resources from one cluster to another

Integrates with DevOps workflows to create ephemeral clones of Kubernetes namespaces

## Ephemeral Clusters



Provides a reliable tool to unlock new approaches to cluster lifecycle management treating clusters as “cattle”

# Traditional Backup

The server was the application



# Kubernetes Cluster



Master Nodes



etcd



Worker Nodes

# Kubernetes Data Types

## Stateless

Master nodes

Worker nodes

## Stateful

etcd database

Persistent volumes

# Protecting Kubernetes Data

## Stateless

Isolate unhealthy nodes

- kubectl cordon & drain

Provision replacement nodes

- Master nodes
- Worker nodes

Solve with automation

## Stateful

etcd database

Persistent volumes

This is why we are here today!

# Strategies for Kubernetes Stateful Data

## etcd

1. Block
2. File system
3. etcdctl
4. Kubernetes API Discovery

## Persistent Volumes

1. Cloud/Storage provider snapshots
2. File system backup
3. CSI snapshots (alpha)

# Velero Features

---

## Backup and restore of Kubernetes objects

- Uses Kubernetes Discovery API
- Does not need to talk directly to etcd
- Backups stored in Cloud Object Storage

## Backup and restore persistent volumes

- Uses cloud provider snapshot APIs
- Restic support (BETA) for file system backups

## Flexibility

Scheduled backups

## Filtering

- Namespaces
- Resources
- Label Selectors

Restore to different namespaces

---

## Extensibility

Hooks

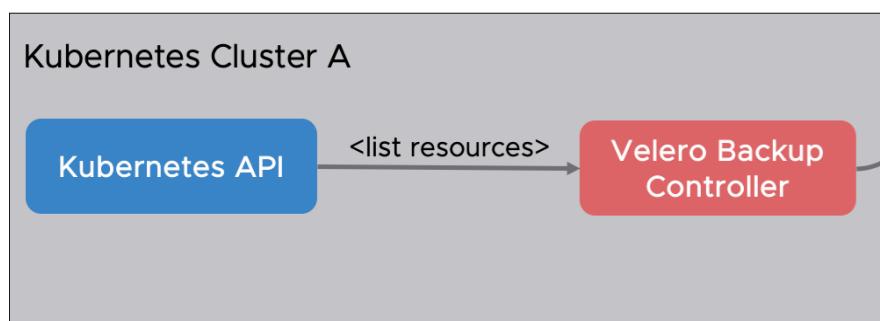
## Plugins

- Object Storage
- Block Storage
- Item Backup Actions
- Item Restore Actions

# Simple Velero Workflow

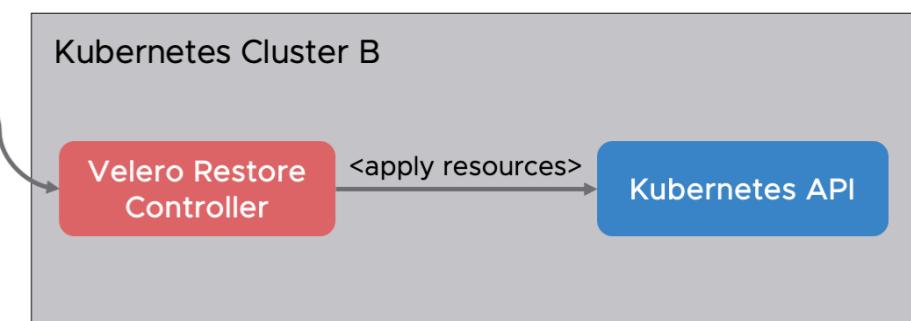
## Backup

```
velero backup create my-backup \
--include-namespaces example \
--snapshot-volumes
```



## Restore

```
velero restore create my-restore \
--from-backup my-backup
```



# Demo

# Velero Current and Future

## Current Release: Velero 1.0

Velero Install Command

Helm Chart Support

Improved Plug-in Support

Restic Integration Moves to BETA

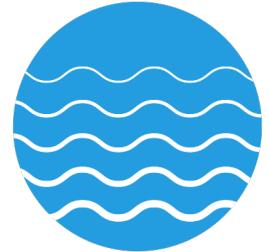
## Next Release: Velero 1.1

CSI Plug-in (ALPHA)

Restic Stability Improvements

Read-only Storage Locations

Resource Usage Limits



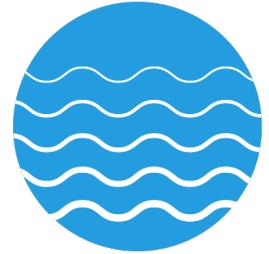
# VELERO Data Protection Survey

Take the Kubernetes Backup User Survey at:

<https://velero.io/survey>

Tell us how you are thinking about Data Protection in the world of Kubernetes and Containers

**Maximize community learning:** anonymized responses will be shared back with the community



# VELERO



[velero.io](https://velero.io)



[github.com/heptio/velero](https://github.com/heptio/velero)



[@projectvelero](https://twitter.com/projectvelero)



Kubernetes Slack channel: [#velero](#)



Mailing list: [groups.google.com/forum/#!forum/projectvelero](https://groups.google.com/forum/#!forum/projectvelero)



Velero community meetings every 1<sup>st</sup> and 3<sup>rd</sup> Tuesday:  
[github.com/heptio/velero-community](https://github.com/heptio/velero-community)

# Q&A



# Thank You