Automated Backup & Restore for Kubernetes Clusters

Jack Munday / IT-PW-PI

August 15th, 2024

Backup & Disaster Recovery Operations is an increasing priority of the CERN IT Dept

> ***Kubernetes clusters deployed at CERN are not backed up by default.***

We provided the tooling and machinery to support BC/DR operations, but it is the user's responsibility to setup, manage and exercise disaster recovery.

Good BC/DR hygiene also helps to support:

- Cluster migrations.
- Environment hydrations from production to testing clusters.

https://about.gitlab.com/blog/2017/02/10/postmortem-of-database-outage-of-january-31/

Why can’t I just backup etcd?

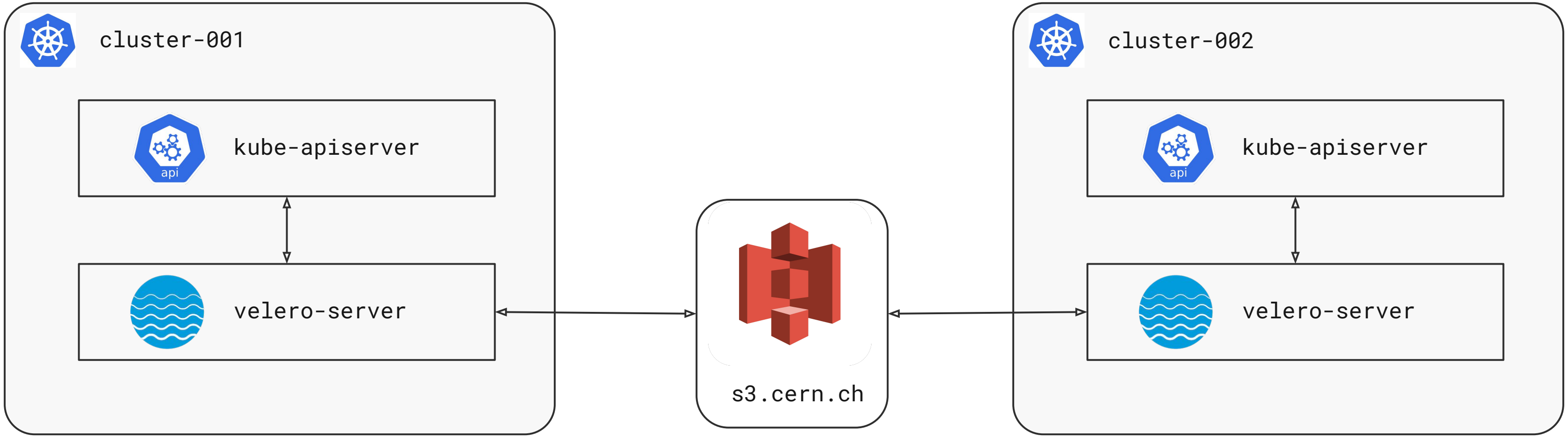
- Works well for a single stateless cluster but does not scale.
- Requires access to / healthy master nodes.
- No support for backing up persistent volumes.
- No support for cluster migration.
- Backups by definition are imperative.
- Much of the data in etcd is easily recoverable in a fully GitOps-managed cluster anyway.

Velero is an open-source tool for managing backups in Kubernetes

- Velero offers backup support for cluster resources (etcd) and the underlying data in persistent volumes.
- Data can be backed up to a number of different cloud storage buckets (i.e. s3, GCS, …).
- Backup / Restore operations are managed as kubernetes resources via a CLI or declaratively.
- Provides fine-grain control over what resources, namespaces to backup / restore.
- Avoids the need to interact with etcd directly.

github.com/vmware-tanzu/velero

Backup workflow



```
$ velero create backup my-backup --wait
```

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

Getting Started with Velero

`velero` is distributed at CERN as part of templates for >=1.30.x

```
cat << EOF >> /tmp/backup-config.yaml
velero:
 enabled: true
 configuration:
   backupStorageLocation:
    - name: default
      provider: aws
      bucket: "{{ S3_BUCKET_NAME }}"
      config:
        region: "default"
        s3Url: http://s3.cern.ch
 credentials:
   useSecret: true
   secretContents:
     cloud: |
       [default]
       aws_access_key_id = "{{ S3_ACCESS_KEY }}"
       aws_secret_access_key = "{{ S3_SECRET_KEY }}"
EOF

$ openstack coe cluster create ... --merge-labels --labels cern_chart_user_values="$(cat /tmp/backup-config.yaml |
base64 -w0)" my-backup-cluster
```

Backing up PVCs

Velero fully supports the backup of PVCs using the underlying volume snapshotting machinery of kubernetes.

However, snapshotting is not fully supported / enabled across all CERN storage types.

Storage Type	Snapshotting Supported	Volume Snapshot Classes	Notes
Meyrin CephFS	No	N/A	-

Meyrin CephFS B	Yes	`manila-cephfs-delete,`
		`manila-cephfs-retain`
			Not Production Ready

Cinder	Yes	`cinder-delete, cinder-retain`	-

Backing up PVCs

Backups can be triggered without using snapshotting, however the atomicity and integrity of these will be compromised if writes are made whilst the backup is running.

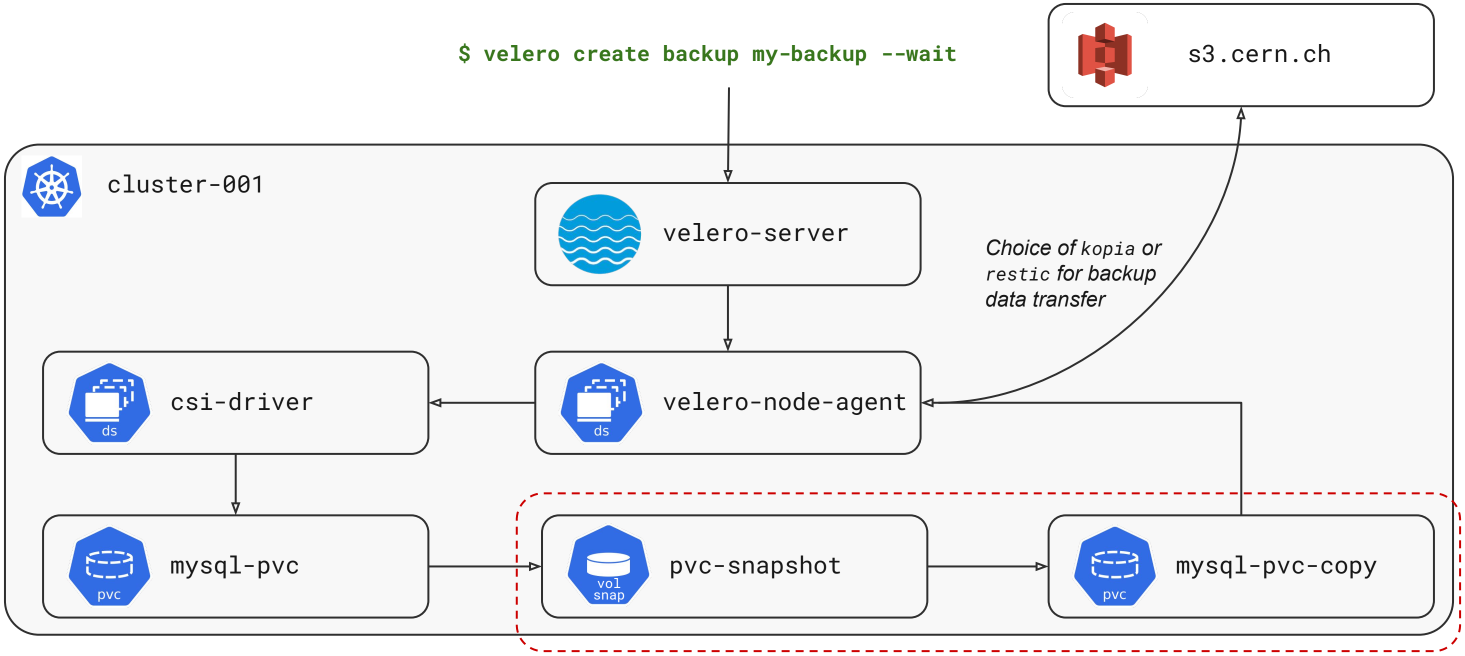
This will lead to incomplete restores.

Velero uses the container storage interface (CSI) to manage the snapshotting and backing up of `PersistentVolumeClaims`.

To opt-in, Annotate each of your `PersistentVolumeClaims` with the volume snapshot class that should be used by velero to backup this `pvc`.

```
annotations:
  velero.io/csi-volumesnapshot-class: "manila-cephfs-retain"
```

PVC Snapshotting Workflow



```
$ velero create backup my-backup --wait
```

Deleted on completion of backup data transfer to s3.

Velero offers fine grain control over what resources to backup and when

- Backups can be triggered on a cron schedule (recommended).
- Resources can be selectively in/excluded using command line flags or resource labels.

```
labels:
    velero.io/exclude-from-backup=true
```

```
 $ velero create backup … \
–include-namespace=core,elastic \
 –exclude-resources=statefulsets
```

- As backups are running in-cluster, alertmanager integration can be used for monitoring of backups and schedules.

```
apiVersion: velero.io/v1
kind: Schedule
metadata:
 name: my-nightly-backup
 namespace: kube-system
spec:
 schedule: 0 0 * * *
 skipImmediately: false
 template:
   includedNamespaces: ['*']
   snapshotMoveData: true
   storageLocation: default
   ttl: 720h0m0s
```

A pathway to truly ephemeral clusters?

Velero is also designed as a cluster migration tool.

Performing in-place upgrades of the kubernetes api version (particularly of a major version) in a cluster can be a risky operation.

Often you will have no failover, which can lead to an outage if your upgrade fails.

Using velero makes it easy to treat “clusters as cattle” by providing tooling to make it easy to move your application from one cluster to another.

3-2-1 Backup Rule

> **3 copies of your Data (Production & 2 Replicas)**
>
> **Across 2 different Mediums**
>
> **With One Off-Site (PDC, Cloud)**

This is the Gold Standard, however in some cases this is not always possible.

Obvious care should be taken when choosing where and what backup mediums.

This requires some understanding of the underlying infrastructure your applications are consuming: backing up a manila share to s3 bucket is no use if they are backed by the same Ceph cluster.

Demo(s)

`https://gitlab.cern.ch/jmunday/webinars`

For more information, please visit:

https://kubernetes.docs.cern.ch/docs/operations/backups/
https://velero.io/docs/main/

home.cern