

Kubernetes 系列（三十四）基于 Velero - 跨 StorageClass 迁移存储

上一篇讲到的跨 StorageClass 迁移过程比较繁琐，本篇介绍一种叫做 Velero 的工具，可以快速实现迁移。

关于 Velero

velero 是一种 Kubernetes 集群备份还原工具，它可以实现备份 crd 以及 pv 存储数据，基于备份数据实现还原。

安装

服务端

数据都备份到 minio，请查看本公众号 安装 minio 部分实现安装。

创建 minio 密钥文件 credentials:

```
[default]
aws_access_key_id = minio
aws_secret_access_key = minio123
```

服务端通过 helm chart 安装，

```
helm repo add vmware-tanzu https://vmware-tanzu.github.io/helm-charts

# 注意参数根据各自实际环境信息进行替换
helm install velero vmware-tanzu/velero \
  --namespace velero \
  --create-namespace \
  --set cleanUpCRDs=true \
  --set configuration.provider=aws \
  --set-file credentials.secretContents.cloud=./credentials \
  --set deployRestic=true \
  --set snapshotsEnabled=false \
  --set configuration.backupStorageLocation.name=minio \
  --set configuration.backupStorageLocation.bucket=velero \
  --set configuration.backupStorageLocation.config.region=minio \
  --set configuration.backupStorageLocation.config.s3ForcePathStyle=true \
  --set configuration.backupStorageLocation.config.s3Url=http://192.168.0.21:9000 \
  --set "initContainers[0].name=velero-plugin-for-aws \
  --set "initContainers[0].image=velero/velero-plugin-for-aws:v1.2.0 \
  --set "initContainers[0].volumeMounts[0].mountPath="/target \
  --set "initContainers[0].volumeMounts[0].name=plugins \
  vmware-tanzu/velero
```

服务全部安装到了 velero 命名空间中，且使用 minio 作为备份数据的存储。

客户端 Cli

客户端 cli，主要用于执行相关的查询、备份以及还原指令，可本地安装，执行时指定配置文件即可。

可根据各自操作系统类型: <https://github.com/vmware-tanzu/velero/releases/latest> 下载二进制程序, 配置环境变量。

跨 Storage Class 迁移

其实跨 Storage Class 迁移, 实际就是利用 velero 创建备份, 然后还原时, 指定新的 class 类型而已, velero 通过创建如下 configmap 配置即可实现:

```
apiVersion: v1
kind: ConfigMap
metadata:
  # 固定名称
  name: change-storage-class-config
  namespace: velero
  labels:
    velero.io/plugin-config: ""
    velero.io/change-storage-class: RestoreItemAction
data:
  # from class name : to class name
  nfs-client: nfs-client-new
```

此配置还原时, 会自动识别到如果是 nfs-client 类型会在恢复时使用 nfs-client-new 类型。

PS: 如果存储数据比较大, 建议不要使用本工具, 备份还原很有可能会 stuck ,不知道什么时候能结束

1. 创建备份

通过如下命令, 备份命名空间资源:

```
# velero backup create [backup name] --include-namespaces [namespace1,namespace2...] --default-volumes-to-restic
# --default-volumes-to-restic 一定要加, 如果pod 有pv 挂载, 会通过 restic 将数据备份到 minio 中去
[root@master yy-work]# velero backup create backup-test --include-namespaces test --default-volumes-to-restic
Backup request "backup-test" submitted successfully.
Run `velero backup describe backup-test` or `velero backup logs backup-test` for more details.
```

查看备份信息:

```
[root@master yy-work]# velero backup describe backup-test --details
Name:      backup-test
Namespace: velero
Labels:    velero.io/storage-location=minio
Annotations: velero.io/source-cluster-k8s-gitversion=v1.23.0
              velero.io/source-cluster-k8s-major-version=1
              velero.io/source-cluster-k8s-minor-version=23
## 备份阶段状态
Phase: Completed

Errors: 0
Warnings: 0

Namespaces:
  Included: test
  Excluded: <none>

Resources:
  Included: *
  Excluded: <none>
  Cluster-scoped: auto
Label selector: <none>
```

Label Selector: <none>

Storage Location: minio

Velero-Native Snapshot PVs: auto

TTL: 720h0m0s

Hooks: <none>

Backup Format Version: 1.1.0

Started: 2022-11-04 16:39:36 +0800 CST

Completed: 2022-11-04 16:40:18 +0800 CST

Expiration: 2022-12-04 16:39:36 +0800 CST

Total items to be backed up: 14

Items backed up: 14

备份的所有资源

Resource List:

analysis.crane.io/v1alpha1/Recommendation:

- test/workloads-rule-replicas-85wkz
- test/workloads-rule-resource-lcc4c

apiextensions.k8s.io/v1/CustomResourceDefinition:

- configauditreports.aquasecurity.github.io
- recommendations.analysis.crane.io

apps/v1/Deployment:

- test/busybox

apps/v1/ReplicaSet:

- test/busybox-5674bb4c67

aquasecurity.github.io/v1alpha1/ConfigAuditReport:

- test/replicaset-busybox-5674bb4c67

v1/ConfigMap:

- test/kube-root-ca.crt

v1/Namespace:

- test

v1/PersistentVolume:

- pvc-88524294-2aea-41b7-ac52-80f016b2c9c9

v1/PersistentVolumeClaim:

- test/pvc-busybox

v1/Pod:

- test/busybox-5674bb4c67-6kp2w

v1/Secret:

- test/default-token-k5c2j

v1/ServiceAccount:

- test/default

Velero-Native Snapshots: <none included>

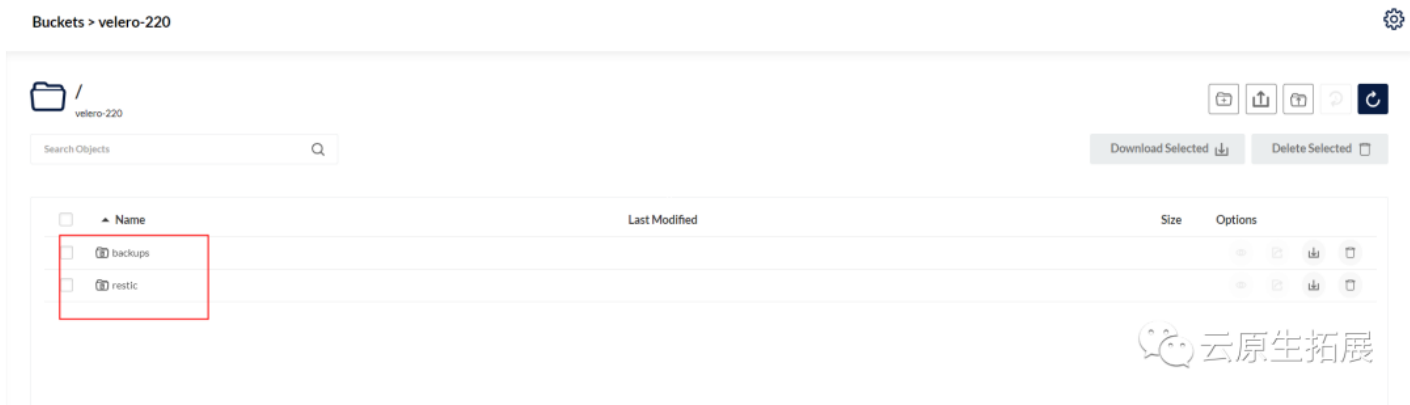
通过 restic 备份的 pv 数据

Restic Backups:

Completed:

test/busybox-5674bb4c67-6kp2w: attachment

最好确认以上信息一致后再进行后续操作，可登录 minio 检查数据：



<https://alidocs.oss-cn-zhangjiakou.aliyuncs.com/res/wYdgOkBLvm67q4BX/img/5c40ced6-81be-4dc6-95a9-ca2a7fe34320.png>

如果你不想全量备份整个命名空间，只想备份某个应用的相关存储，则参考下面的命令：

```
# 一定要加上 po 资源类型，否则 po 上面的存储不会备份，也可以通过 标签选择相应的资源，通过命令行帮助去使用
# 同时 deploy（根据各自的控制器类型） pv pvc ，也要备份，更新操作可能会卡住，建议都备份，对应的 在下面的删除资源环节，也要对应的删除
velero backup create backup-xxx --include-resources po,pv,pvc,deploy --include-namespaces test --default-volumes-to-restic
```

2. 删除集群中备份的资源

虽然 velero 支持恢复时，更新操作，对一些常规的非工作负载对象类型还好，对于 pod、pvc 这些我试过，最终会卡在那里，最好删除相关资源。

3. 执行备份恢复

通过如下命令恢复数据：

```
# velero restore create --from-backup [backup name]
[root@master yy-work]# velero restore create --from-backup backup-test
Restore request "backup-test-20221104165018" submitted successfully.
Run `velero restore describe backup-test-20221104165018` or `velero restore logs backup-test-20221104165018` for more details.
[root@master yy-work]
```

查看 restore 信息(上面的数据有 返回 restore id)：

```
[root@master yy-work]# velero restore describe backup-test-20221104165018 --details
Name:          backup-test-20221104165018
Namespace:     velero
Labels:        <none>
Annotations:   <none>

Phase:          Completed
Total items to be restored: 13
Items restored: 13

Started:        2022-11-04 16:50:19 +0800 CST
Completed:      2022-11-04 16:51:18 +0800 CST

Warnings:
  Velero:        <none>
  Cluster:       could not restore, CustomResourceDefinition "configauditreports.aquasecurity.github.io" already exists. Warning: the in
                  could not restore, CustomResourceDefinition "recommendations.analysis.crane.io" already exists. Warning: the in-cluster
  Namespaces:    <none>

Backup: backup-test

Namespaces:
  Included:      all namespaces found in the backup
  Excluded:      <none>

Resources:
  Included:      *
  Excluded:      nodes, events, events.events.k8s.io, backups.velero.io, restores.velero.io, resticrepositories.velero.io
  Cluster-scoped: auto

Namespace mappings: <none>

Label selector:  <none>

Restore PVs:    auto

Restic Restores:
  Completed:
    test/busybox-5674bb4c67-6kp2w: attachment

Existing Resource Policy: <none>

Preserve Service NodePorts: auto
[root@master yy-work]#
```

等待 到达 Complete 状态, 检查 pv class 以及存储的数据是否已恢复。

欢迎关注我的公众号“云原生拓展”, 原创技术文章第一时间推送。