Red Hat OpenShift Data Foundation 4.10

Recovering a stretch cluster

TECHNOLOGY PREVIEW: Instructions on how to recover applications and their storage from a disaster in Red Hat OpenShift Data Foundation. This solution is a technology preview feature and is not intended to be run in production environments.

Last Updated: 2022-09-22

Red Hat OpenShift Data Foundation 4.10 Recovering a stretch cluster

TECHNOLOGY PREVIEW: Instructions on how to recover applications and their storage from a disaster in Red Hat OpenShift Data Foundation. This solution is a technology preview feature and is not intended to be run in production environments.

Legal Notice

Abstract

This document explains how to recover from a stretch cluster disaster in Red Hat OpenShift Data Foundation. Recovering a stretch cluster is a Technology Preview feature. Technology Preview features are not supported with Red Hat production service level agreements (SLAs) and might not be functionally complete. Red Hat does not recommend using them in production. These features provide early access to upcoming product features, enabling customers to test functionality and provide feedback during the development process.

Table of Contents

PREFACE

Given that the stretch cluster disaster recovery solution is to provide resiliency in the face of a complete or partial site outage, it is important to understand the different methods of recovery for applications and their storage.

How the application is architected determines how soon it becomes available again on the active zone.

There are different methods of recovery for applications and their storage depending on the site outage. The recovery time depends on the application architecture. The different methods of recovery are as follows:

- Recovery for zone-aware HA applications with RWX storage .
- Recovery for HA applications with RWX storage .
- Recovery for applications with RWO storage .
- Recovery for StatefulSet pods.

CHAPTER 1. UNDERSTANDING ZONE FAILURE

For the purpose of this section, zone failure is considered as a failure where all OpenShift Container Platform, master and worker nodes in a zone are no longer communicating with the resources in the second data zone (for example, powered down nodes). If communication between the data zones is still partially working (intermittently up or down), the cluster, storage, and network admins should disconnect the communication path between the data zones for recovery to succeed.

CHAPTER 2. RECOVERY FOR ZONE-AWARE HA APPLICATIONS WITH RWX STORAGE

Applications that are deployed with **topologyKey: topology.kubernetes.io/zone**, have one or more replicas scheduled in each data zone, and are using shared storage, that is, ReadWriteMany (RWX) CephFS volume, recovers on the active zone within 30-60 seconds for new connections. The short pause is for **HAProxy** to refresh connections if a router pod is now offline in the failed data zone.

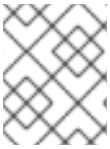
An example of this type of application is detailed in the Install Zone Aware Sample Application section.

IMPORTANT

When you install the sample application, power off the OpenShift Container Platform nodes (at least the nodes with OpenShift Data Foundation devices) to test the failure of a data zone in order to validate that your file-uploader application is available, and you can upload new files.

CHAPTER 3. RECOVERY FOR HA APPLICATIONS WITH RWX STORAGE

Applications that are using **topologyKey: kubernetes.io/hostname** or no topology configuration, have no protection against all of the application replicas being in the same zone.



NOTE

This can happen even with *podAntiAffinity* and **topologyKey: kubernetes.io/hostname** in the **Pod** spec because this anti-affinity rule is host-based and not zone-based.

If this happens and all replicas are located in the zone that fails, the application using ReadWriteMany (RWX) storage takes 6-8 minutes to recover on the active zone. This pause is for the OpenShift Container Platform nodes in the failed zone to become **NotReady** (60 seconds) and then for the default pod eviction timeout to expire (300 seconds).

CHAPTER 4. RECOVERING APPLICATIONS WITH RWO STORAGE

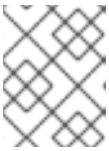
Applications that use ReadWriteOnce (RWO) storage have a known behavior described in this Kubernetes issue. Because of this issue, if there is a data zone failure, any application pods in that zone mounting RWO volumes (for example, **cephrbd** based volumes) are stuck with **Terminating** status after 6-8 minutes and is not re-created on the active zone without manual intervention.

Check the OpenShift Container Platform nodes with a status of **NotReady**. There may be an issue that prevents the nodes from communicating with the OpenShift control plane. However, the nodes may still be performing I/O operations against Persistent Volumes (PVs).

If two pods are concurrently writing to the same RWO volume, there is a risk of data corruption. Ensure that processes on the **NotReady** node are either terminated or blocked until they are terminated.

Example solutions:

- Use an out of band management system to power off a node, with confirmation, to ensure process termination.
- Withdraw a network route that is used by nodes at a failed site to communicate with storage.



NOTE

Before restoring service to the failed zone or nodes, confirm that all the pods with PVs have terminated successfully.

To get the **Terminating** pods to recreate on the active zone, you can either force delete the pod or delete the finalizer on the associated PV. Once one of these two actions are completed, the application pod should recreate on the active zone and successfully mount its RWO storage.

Force deleting the pod

Force deletions do not wait for confirmation from the kubelet that the pod has been terminated.

```
$ oc delete pod <PODNAME> --grace-period=0 --force --namespace <NAMESPACE>
```

<PODNAME>

Is the name of the pod

<NAMESPACE>

Is the project namespace

Deleting the finalizer on the associated PV

Find the associated PV for the Persistent Volume Claim (PVC) that is mounted by the Terminating pod and delete the finalizer using the **oc patch** command.

```
$ oc patch -n openshift-storage pv/<PV_NAME> -p '{"metadata":{"finalizers":[]}}' --type=merge
```

<PV_NAME>

Is the name of the PV

An easy way to find the associated PV is to describe the Terminating pod. If you see a multi-attach warning, it should have the PV names in the warning (for example, **pvc-0595a8d2-683f-443b-aee0-6e547f5f5a7c**).

```
$ oc describe pod <PODNAME> --namespace <NAMESPACE>
```

<PODNAME>

Is the name of the pod

<NAMESPACE>

Is the project namespace

Example output:

```
[...]
Events:
  Type     Reason                  Age   From                     Message
  ----     ------                  ----  ----                     -------
  Normal   Scheduled               4m5s  default-scheduler        Successfully assigned openshift-
storage/noobaa-db-pg-0 to perf1-mz8bt-worker-d2hdm
  Warning  FailedAttachVolume      4m5s  attachdetach-controller  Multi-Attach error for volume
"pvc-0595a8d2-683f-443b-aee0-6e547f5f5a7c" Volume is already exclusively attached to one
node and can't be attached to another
```

CHAPTER 5. RECOVERY FOR STATEFULSET PODS

Pods that are part of a StatefulSet have a similar issue as pods mounting ReadWriteOnce (RWO) volumes. More information is referenced in the Kubernetes resource StatefulSet considerations.

To get the pods part of a StatefulSet to re-create on the active zone after 6-8 minutes you need to force delete the pod with the same requirements (that is, OpenShift Container Platform node powered off or communication disconnected) as pods with RWO volumes.