Red Hat OpenShift Data Foundation 4.17

4.17 Release Notes

Release notes for features and enhancements, known issues, and other important release information.

Last Updated: 2024-10-30

Red Hat OpenShift Data Foundation 4.17 4.17 Release Notes

Release notes for features and enhancements, known issues, and other important release information.

Legal Notice

Abstract

The release notes for Red Hat OpenShift Data Foundation 4.17 summarizes all new features and enhancements, notable technical changes, and any known bugs upon general availability.

Table of Contents

MAKING OPEN SOURCE MORE INCLUSIVE 4

CHAPTER 1. OVERVIEW 5
1.1. ABOUT THIS RELEASE 5

CHAPTER 2. NEW FEATURES 6
2.1. DISASTER RECOVERY SOLUTION 6
2.1.1. Disabling DR for a given workload 6
2.1.2. Disaster recovery protection for cloned and restored RBD volumes 6
2.1.3. Setting up DR in clusters with overlapping pod and service CIDR 6
2.1.4. VM DR protection for cloned volume 6
2.2. VMWARE VSPHERE8 SUPPORT 6
2.3. STORAGE CLASS WITH 2-WAY REPLICATION FOR CEPHFS 6
2.4. KEY ROTATION FOR PV ENCRYPTION 6
2.5. AZURE KEY VAULT SUPPORT ON OPENSHIFT DATA FOUNDATION KMS 7
2.6. MANUAL SETTING OF MULTICLOUD OBJECT GATEWAY CREDENTIALS TO AN ACCOUNT 7
2.7. DOWNLOADING THE OPENSHIFT DATA FOUNDATION CLI TOOL FROM THE USER INTERFACE 7

CHAPTER 3. ENHANCEMENTS 8
3.1. CAPACITY CONSUMPTION TREND CARD 8
3.2. BUCKET LOGGING AND LOG BASED REPLICATION OPTIMIZATION FOR MULTICLOUD OBJECT GATEWAY BUCKETS 8
3.3. CAPACITY ALERT THRESHOLDS INCREASED 8
3.4. CEPH FULL THRESHOLDS CONFIGURATIONS 8
3.5. PRESERVE THE CLI FLAGS THAT WERE PASSED DURING CREATION OF EXTERNAL CLUSTERS 8
3.6. MDS SCALABILITY 8
3.7. TRADITIONAL USER EXPERIENCE FOR MUST GATHER UTILITY 9
3.8. CEPHCSI POD LOG ROTATION 9
3.9. MULTIPLE FILESYSTEMS 9
3.10. TOPOLOGYSPREADCONSTRAINTS ADDED TO PV BACKINGSTORE PODS SO THEY GET SCHEDULED ON A SPREAD BASIS 9

CHAPTER 4. DEVELOPER PREVIEWS 10
4.1. MULTI-VOLUME CONSISTENCY FOR BACKUP - CEPHFS 10
4.2. EFFICIENT SELINUX VOLUME RELABELLING FOR ALL ACCESS MODES 10
4.3. CEPHFS BASED PERSISTENT VOLUMES TO BE SHARED WITH CONCURRENT IOS ACROSS CLUSTERS 10
4.4. ACCESS BUCKETS WITH DNS SUBDOMAIN STYLE (VIRTUAL HOST STYLE) FOR RGW 10

CHAPTER 5. BUG FIXES 11
5.1. DISASTER RECOVERY 11
5.2. MULTICLOUD OBJECT GATEWAY 12
5.3. CEPH 13
5.4. OPENSHIFT DATA FOUNDATION CONSOLE 13
5.5. ROOK 14
5.6. CEPH MONITORING 14
5.7. CSI DRIVER 14

CHAPTER 6. KNOWN ISSUES 16
6.1. DISASTER RECOVERY 16
6.2. MULTICLOUD OBJECT GATEWAY 20
6.3. CEPH 20
6.4. CSI DRIVER 21

6.5. OPENSHIFT DATA FOUNDATION CONSOLE 21
6.6. OCS OPERATOR 21

MAKING OPEN SOURCE MORE INCLUSIVE

Red Hat is committed to replacing problematic language in our code, documentation, and web properties. We are beginning with these four terms: master, slave, blacklist, and whitelist. Because of the enormity of this endeavor, these changes will be implemented gradually over several upcoming releases. For more details, see our CTO Chris Wright's message.

CHAPTER 1. OVERVIEW

Red Hat OpenShift Data Foundation is software-defined storage that is optimized for container environments. It runs as an operator on OpenShift Container Platform to provide highly integrated and simplified persistent storage management for containers.

Red Hat OpenShift Data Foundation is integrated into the latest Red Hat OpenShift Container Platform to address platform services, application portability, and persistence challenges. It provides a highly scalable backend for the next generation of cloud-native applications, built on a technology stack that includes Red Hat Ceph Storage, the Rook.io Operator, and NooBaa's Multicloud Object Gateway technology.

Red Hat OpenShift Data Foundation is designed for FIPS. When running on RHEL or RHEL CoreOS booted in FIPS mode, OpenShift Container Platform core components use the RHEL cryptographic libraries submitted to NIST for FIPS Validation on only the x86_64, ppc64le, and s390X architectures. For more information about the NIST validation program, see Cryptographic Module Validation Program. For the latest NIST status for the individual versions of the RHEL cryptographic libraries submitted for validation, see Compliance Activities and Government Standards.

Red Hat OpenShift Data Foundation provides a trusted, enterprise-grade application development environment that simplifies and enhances the user experience across the application lifecycle in a number of ways:

- Provides block storage for databases.
- Shared file storage for continuous integration, messaging, and data aggregation.
- Object storage for cloud-first development, archival, backup, and media storage.
- Scale applications and data exponentially.
- Attach and detach persistent data volumes at an accelerated rate.
- Stretch clusters across multiple data-centers or availability zones.
- Establish a comprehensive application container registry.
- Support the next generation of OpenShift workloads such as Data Analytics, Artificial Intelligence, Machine Learning, Deep Learning, and Internet of Things (IoT).
- Dynamically provision not only application containers, but data service volumes and containers, as well as additional OpenShift Container Platform nodes, Elastic Block Store (EBS) volumes and other infrastructure services.

1.1. ABOUT THIS RELEASE

Red Hat OpenShift Data Foundation 4.17 (RHSA-2024:8676) is now available. New enhancements, features, and known issues that pertain to OpenShift Data Foundation 4.17 are included in this topic.

Red Hat OpenShift Data Foundation 4.17 is supported on the Red Hat OpenShift Container Platform version 4.17. For more information, see Red Hat OpenShift Data Foundation Supportability and Interoperability Checker.

For Red Hat OpenShift Data Foundation life cycle information, refer to the layered and dependent products life cycle section in Red Hat OpenShift Container Platform Life Cycle Policy .

CHAPTER 2. NEW FEATURES

This section describes new features introduced in Red Hat OpenShift Data Foundation 4.17.

2.1. DISASTER RECOVERY SOLUTION

2.1.1. Disabling DR for a given workload

Disaster recovery (DR) can be disabled for a given workload for managed applications and discovered applications through the OpenShift Web Console.

For more information, see Disabling disaster recovery for a disaster recovery enabled application .

2.1.2. Disaster recovery protection for cloned and restored RBD volumes

Disaster recovery protection can be enabled for cloned and restored RADOS Block Device (RBD) volumes during DR policy creation for Regional-DR.

For more information, see Disaster recovery protection for cloned and restored RBD volumes .

2.1.3. Setting up DR in clusters with overlapping pod and service CIDR

The ODF CLI tool can be used on existing OpenShift deployments with overlapping pod and service CIDR to verify the clusters meet the prerequisites for peering. The tool verifies the two clusters have non-overlapping service and pod Classless Inter-Domain Routing (CIDR).

For more information, see Requirements for enabling Regional-DR.

2.1.4. VM DR protection for cloned volume

DR protection is simplified and automatically decouples parent and child so that each one can independently be DR protected. This eliminates the dependency on the parent volume and VM and its DR status.

2.2. VMWARE VSPHERE8 SUPPORT

OpenShift Data Foundation supports deployment on VMware vSphere version 8.

2.3. STORAGE CLASS WITH 2-WAY REPLICATION FOR CEPHFS

Storage class with 2-way replication (replica-2) enables reduction in overhead with CephFS when data resiliency is not a primary concern. The 2-way replication reduces the amount of storage space used and decreases the level of fault tolerance.

For more information, see Using 2-way replication with CephFS.

2.4. KEY ROTATION FOR PV ENCRYPTION

Enabling key rotation when using the key management system (KMS) helps to achieve the security common practices that require periodic encryption key rotation.

For more information, see Enabling key rotation when using KMS.

2.5. AZURE KEY VAULT SUPPORT ON OPENSHIFT DATA FOUNDATION KMS

OpenShift Data Foundation on Microsoft Azure platform supports Azure Key Vault for cluster-wide PV encryption. This Key Vault helps to use a local native solution for Azure instead of a third party tool.

For more information, see Creating an OpenShift Data Foundation cluster .

2.6. MANUAL SETTING OF MULTICLOUD OBJECT GATEWAY CREDENTIALS TO AN ACCOUNT

Multicloud Object Gateway credentials can be set manually using the NooBaa CLI commands.

For more information, see Setting Multicloud Object Gateway account credentials using CLI command.

2.7. DOWNLOADING THE OPENSHIFT DATA FOUNDATION CLI TOOL FROM THE USER INTERFACE

With the OpenShift Data Foundation command line interface (CLI) tool, a Data Foundation environment can be effectively managed and troubleshot from a terminal. A compatible version of the CLI tool can be downloaded from the customer portal.

CHAPTER 3. ENHANCEMENTS

This section describes the major enhancements introduced in Red Hat OpenShift Data foundation 4.17.

3.1. CAPACITY CONSUMPTION TREND CARD

Consumption trend card in OpenShift Data Foundation dashboard provides information about the estimated days until the storage is full and this helps in new hardware procurement. Storage consumption rate is calculated based on the actual capacity utilization, historical usage, and current consumption rate and is displayed in GiB per day. The number of days left for the storage to reach the threshold capacity is also shown.

For more information, see Metrics in the Block and File dashboard .

3.2. BUCKET LOGGING AND LOG BASED REPLICATION OPTIMIZATION FOR MULTICLOUD OBJECT GATEWAY BUCKETS

Supports replication of large amounts of data between Multicloud Object Gateway (MCG) and Amazon Web Services (AWS) or MCG and MCG. The support for log-based replication for AWS S3 using bucket logging is extended to MCG bucket for optimization. The log-based replication optimization also supports object prefix filtering.

For more information, see Bucket logging for Multicloud Object .

3.3. CAPACITY ALERT THRESHOLDS INCREASED

The thresholds of the PersistentVolumeUsageCritical and PersistentVolumeUsageNearFull alerts are increased so that they are triggered only when the space is limited.

Previously, these PersistentVolumeUsageCritical and PersistentVolumeUsageNearFull alerts that were triggered even when there was still plenty of space available caused unnecessary worry about the state of the cluster.

3.4. CEPH FULL THRESHOLDS CONFIGURATIONS

The Ceph OSD full thresholds can be set using the ODF CLI tool or by updating the StorageCluster CR.

For more information, see Setting Ceph OSD full thresholds by updating the StorageCluster CR .

3.5. PRESERVE THE CLI FLAGS THAT WERE PASSED DURING CREATION OF EXTERNAL CLUSTERS

The command-line interface (CLI) flags passed during creation can be preserved during upgrade automatically. Passing the new flags during upgrade helps to use additional features.

For more information, see Creating an OpenShift Data Foundation Cluster for external Ceph storage system.

3.6. MDS SCALABILITY

Multiple active metadata services (MDS) can be run when MDS CPU usage becomes too high.

For more information, see the Troubleshooting guide.

3.7. TRADITIONAL USER EXPERIENCE FOR MUST GATHER UTILITY

In previous versions, running **must-gather** required an **< -arg>** flag to specify which logs to collect. With this update, **must-gather** no longer requires a **< -arg>** flag, and by default collects all logs.

3.8. CEPHCSI POD LOG ROTATION

Log rotation of CephCSI pods is enabled with **csi-operator** so client operator can use it.

3.9. MULTIPLE FILESYSTEMS

Creating multiple filesystems on the same cluster node for hybrid cluster or any other use case is supported.

3.10. TOPOLOGYSPREADCONSTRAINTS ADDED TO PV BACKINGSTORE PODS SO THEY GET SCHEDULED ON A SPREAD BASIS

Previously, when a PV backingstore was created, the backingstore pod was scheduled randomly on any node. This was not ideal because if any node went down, all the backingstore pods on that node would go down as well. To increase high availability, a topology spread constraint is added to the backingstore pod so that they get scheduled on a spread basis.

CHAPTER 4. DEVELOPER PREVIEWS

This section describes the developer preview features introduced in Red Hat OpenShift Data Foundation 4.17.

IMPORTANT

Developer preview feature is subject to Developer preview support limitations. Developer preview releases are not intended to be run in production environments. The clusters deployed with the developer preview features are considered to be development clusters and are not supported through the Red Hat Customer Portal case management system. If you need assistance with developer preview features, reach out to the ocs-devpreview@redhat.com mailing list and a member of the Red Hat Development Team will assist you as quickly as possible based on availability and work schedules.

4.1. MULTI-VOLUME CONSISTENCY FOR BACKUP - CEPHFS

Provides crash consistent multi-volume consistency groups for backup solutions to be used by applications that are deployed over multiple volumes. This provides support for OpenShift Virtualization and helps to better support applications.

For an example, see the knowledgebase article, Enabling Consistency Groups for CephFS Workloads.

4.2. EFFICIENT SELINUX VOLUME RELABELLING FOR ALL ACCESS MODES

OpenShift Data Foundation now supports efficient SELinux volume relabelling for all access modes. The context set for SELinux in the **PodSpec** or container is used by the driver to mount the volume directly with the correct SELinux labels. This feature helps to eliminate the need to recursively relabel the volume and pod startup can be significantly faster.

For more information, see the knowledgebase article, SELinux relabeling using mount options (RWX) .

4.3. CEPHFS BASED PERSISTENT VOLUMES TO BE SHARED WITH CONCURRENT IOS ACROSS CLUSTERS

OpenShift Data Foundation supports Active-Active High Availability between the multi-instances of the application. Also, supports shared volume for multi-instance applications that are running on different clusters. Ceph-FS based RWX volume is simultaneously mounted on different Pods in different clusters with concurrent IOs from multiple instances of the application.

For more information, see the knowledgebase article Enabling Consistency Groups for CephFS Workloads.

4.4. ACCESS BUCKETS WITH DNS SUBDOMAIN STYLE (VIRTUAL HOST STYLE) FOR RGW

Virtual-host-style S3 bucket addressing can be configured for OpenShift Data Foundation internal mode object stores and ObjectBucketClaims. The client access is secured with TLS using OpenShift builtin platform certificates.

CHAPTER 5. BUG FIXES

This section describes the notable bug fixes introduced in Red Hat OpenShift Data Foundation 4.17.

5.1. DISASTER RECOVERY

- **FailOver of applications are hung in FailingOver state**
  Previously, applications were not DR protected successfully because of the errors in protecting required resources to the provided S3 stores. So, failing over such applications resulted in FailingOver state.

  With this fix, a metric and a related alert is added to the application DR protection health that shows an alert to rectify protection issues after DR protects the applications. As a result, the applications that are successfully protected are failed over.

  (BZ#2248723)

- **Post hub recovery, applications which were in FailedOver state consistently report FailingOver**
  Previously, after recovering a DR setup from a hub and a ManageCluster loss to a passive hub, applications which were in **FailedOver** state to the lost ManagedCluster consistently reported **FailingOver** status. Failing over such applications to the surviving cluster was allowed but required checks were missing on the surviving cluster to ensure that the failover can be initiated.

  With this fix, Ramen hub operator ensures if the target cluster is ready for a failover operation before initiating the action. As a result, any failover initiated is successful or if stale resources still exist on the failover target cluster, the operator stalls the failover till the stale resources are cleaned up.

  (BZ#2247847)

- **Post hub recovery, subscription app pods now come up after Failover**
  Previously, post hub recovery, the subscription application pods did not come up after failover from primary to the secondary managed clusters. This caused RBAC error occurs in AppSub subscription resource on managed cluster due to a timing issue in the backup and restore scenario.

  This issue has been fixed, and subscription app pods now come up after failover from primary to secondary managed clusters.

  (BZ#2295782)

- **Application namespaces are no longer left behind in managed clusters after deleting the application**
  Previously, if an application was deleted on the RHACM hub cluster and its corresponding namespace was deleted on the managed clusters, the namespace reappeared on the managed cluster.

  With this fix, once the corresponding namespace is deleted, the application no longer reappears.

  (BZ#2059669)

- **odf-client-info config map is now created**
  Previously, the controller inside MCO was not properly filtering the **ManagedClusterView** resource. This lead to a key config map **odf-client-info** to not be created.

With this update, the filtering mechanism has been fixed, and **odf-client-info** config map is created as expected.

(BZ#2308144)

5.2. MULTICLOUD OBJECT GATEWAY

- **Ability to change log level of backingstore pods**
  Previously, there was no way to change the log level of backingstore pods. With this update, changing the NOOBAA_LOG_LEVEL in the config map will now change the debug level of the pv-pools backingstore pods accordingly.

  (BZ#2297448)

- **STS token expiration now works as expected**
  Previously, incorrect STS token expiration time calculations and printings caused STS tokens to remain valid long past after their expiration time. Users would see the wrong expiration time when trying to assume a role.

  With this update, the STS code was revamped and modified to fix the problems, as well as added support for the CLI flag **--duration-seconds**. Now STS token expiration works as expected, and is shown to the user properly.

  (BZ#2299801)

- **Block deletion of OBC via regular S3 flow**
  S3 buckets can be created both via object bucket claim (OBC) and directly via the S3 operation. When a bucket is created with an OBC and deleted via S3, it leaves the OBC entity dangling and the state is inconsistent. With this update, deleting an OBC via regular S3 flow is blocked, avoiding an inconsistent state.

  (BZ#2301657)

- **NooBaa Backingstore no longer stuck in Connecting post upgrade**
  Previoulsy, NooBaa backingstore blocked upgrade as it remained in the **Connecting** phase leaving the storagecluster.yaml in phase **Progressing**. This issue has been fixed, and upgrade progresses as expected.

  (BZ#2302507)

- **NooBaa DB cleanup no longer fails**
  Previously, NooBaa DB's cleanup would stop after **DB_CLEANER_BACK_TIME** elapsed from the start time of noobaa-core pod. This meant NooBaa DB PVC consumption would rise. This issue has been fixed, and NooBaa DB cleanup works as expected.

  (BZ#2305978)

- **MCG standalone upgrade working as expected**
  Previously, a bug caused NooBaa pods to have incorrect affinity settings, leaving them stuck in the pending state.

  This fix ensures that any previously incorrect affinity settings on the NooBaa pods are cleared. Affinity is now only applied when the proper conditions are met, preventing the issue from recurring after the upgrade.

After upgrading to the fixed version, the pending NooBaa pods won't automatically restart. To finalize the upgrade, manually delete the old pending pods. The new pods will then start with the correct affinity settings, allowing them to run successfully.

(BZ#2314636)

5.3. CEPH

- **New restored or cloned CephFS PVC creation no longer slows down due to parallel clone limit**
  Previously, upon reaching the limit of parallel CephFS clones, the rest of the clones would queue up, slowing down the cloning.

  With this enhancement, upon reaching the limit of parallel clones at one time, the new clone creation requests are rejected. The default parallel clone creation limit is 4.

  To increase the limit, contact customer support.

  (BZ#2190161)

5.4. OPENSHIFT DATA FOUNDATION CONSOLE

- **Pods created in openshift-storage by end users no longer cause errors**
  Previously, when a pod was created in **openshift-storage** by an end user it would cause the console topology page to break. This was because pods without any **ownerReferences** were not considered to be part of the design.

  With this fix, pods without owner references are filtered out, and only pods with correct **ownerReferences** are shown. This allows for the topology page to work correctly even when pods are arbitrarily added to the **openshift-storage** namespace.

  (BZ#2245068)

- **Applying an object bucket claim (OBC) no longer causes an error**
  Previously, when attaching an OBC to a deployment using the OpenShift Web Console, the error **Address form errors to proceed** was shown even when there were no errors in the form. With this fix, the form validations have been changed, and there is no longer an error.

  (BZ#2302575)

- **Automatic mounting of service account tokens disabled to increase security**
  By default, OpenShift automatically mounts a service account token into every pod, regardless of whether the pod needs to interact with the OpenShift API. This behavior can expose the pod's service account token to unintended use. If a pod is compromised, the attacker could gain access to this token, leading to possible privilege escalation within the cluster.

  If the default service account token is unnecessarily mounted, and the pod becomes compromised, the attacker can use the service account credentials to interact with the OpenShift API. This access could lead to serious security breaches, such as unauthorized actions within the cluster, exposure of sensitive information, or privilege escalation across the cluster.

  To mitigate this vulnerability, the automatic mounting of service account tokens is disabled unless explicitly needed by the application running in the pod. In the case of ODF console pod the fix involved disabling the automatic mounting of the default service account token by setting the **automountServiceAccountToken: false** in the pod or service account definition.

With this fix, pods no longer automatically mount the service account token unless explicitly needed. This reduces the risk of privilege escalation or misuse of the service account in case of a compromised pod.

(BZ#2302857)

- **Provider mode clusters no longer have the option to connect to external RHCS cluster**
  Previously, during provider mode deployment there was the option to deploy external RHCS. This resulted in an unsupported deployment.

  With this fix, connecting to external RHCS is now blocked so users do not end with an unsupported deployment.

  (BZ#2312442)

5.5. ROOK

- **Rook.io Operator no longer gets stuck when removing a mon from quorum**
  Previously, mon quorum could be lost when removing a mon from quorum due to a race condition. This was because there might not have been enough quorum to complete the removal of the mon from quorum.

  This issue has been fixed, and the Rook.io Operator no longer gets stuck when removing a mon from quorum.

  (BZ#2292435)

- **Network Fence for non-graceful node shutdown taint no longer blocks volume mount on surviving zone**
  Previously, Rook was creating NetworkFence CR with an incorrect IP address when a node was tainted as out-of-service. Fencing the wrong IP address was blocking the application pods from moving to another node when a taint was added.

  With this fix, auto NetworkFence has been disabled in Rook when the out-of-service taint is added on the node, and application pods are no longer blocked from moving to another node.

  (BZ#2315666)

5.6. CEPH MONITORING

- **Invalid KMIP configurations now treated as errors**
  Previously, Thales Enterprise Key Management (KMIP) was not added in the recognized KMS services. This meant that whenever an invalid KMIP configuration was provided, it was not treated as an error.

  With this fix, Thales KMIP service has been added as a valid KMS service. This enables KMS services to propagate KMIP configuration statuses correctly. Therefore, any mis-configurations are treated as errors.

  (BZ#2271773)

5.7. CSI DRIVER

- **Pods no longer get stuck during upgrade**
  Previously, if there was a node with an empty label, PVC mount would fail during upgrade.

With this fix, nodes labeled with empty value aren't considered for the **crush_location** mount, so they no longer block PVC mounting.

(BZ#2297265)

CHAPTER 6. KNOWN ISSUES

This section describes the known issues in Red Hat OpenShift Data Foundation 4.17.

6.1. DISASTER RECOVERY

- **ceph df reports an invalid MAX AVAIL value when the cluster is in stretch mode**
  When a crush rule for a Red Hat Ceph Storage cluster has multiple "take" steps, the **ceph df** report shows the wrong maximum available size for the map. The issue will be fixed in an upcoming release.

  (BZ#2100920)

- **Both the DRPCs protect all the persistent volume claims created on the same namespace**
  The namespaces that host multiple disaster recovery (DR) protected workloads, protect all the persistent volume claims (PVCs) within the namespace for each DRPlacementControl resource in the same namespace on the hub cluster that does not specify and isolate PVCs based on the workload using its **spec.pvcSelector** field.

  This results in PVCs that match the DRPlacementControl **spec.pvcSelector** across multiple workloads. Or, if the selector is missing across all workloads, replication management to potentially manage each PVC multiple times and cause data corruption or invalid operations based on individual DRPlacementControl actions.

  Workaround: Label PVCs that belong to a workload uniquely, and use the selected label as the DRPlacementControl **spec.pvcSelector** to disambiguate which DRPlacementControl protects and manages which subset of PVCs within a namespace. It is not possible to specify the **spec.pvcSelector** field for the DRPlacementControl using the user interface, hence the DRPlacementControl for such applications must be deleted and created using the command line.

  Result: PVCs are no longer managed by multiple DRPlacementControl resources and do not cause any operation and data inconsistencies.

  (BZ#2128860)

- **MongoDB pod is in CrashLoopBackoff because of permission errors reading data in cephrbd volume**
  The OpenShift projects across different managed clusters have different security context constraints (SCC), which specifically differ in the specified UID range and/or **FSGroups**. This leads to certain workload pods and containers failing to start post failover or relocate operations within these projects, due to filesystem access errors in their logs.

  Workaround: Ensure workload projects are created on all managed clusters with the same project-level SCC labels, allowing them to use the same filesystem context when failed over or relocated. Pods will no longer fail post-DR actions on filesystem-related access errors.

  (BZ#2081855)

- **Disaster recovery workloads remain stuck when deleted**
  When deleting a workload from a cluster, the corresponding pods might not terminate with events such as **FailedKillPod**. This might cause delay or failure in garbage collecting dependent DR resources such as the **PVC**, **VolumeReplication**, and **VolumeReplicationGroup**. It would also prevent a future deployment of the same workload to the cluster as the stale resources are not yet garbage collected.

Workaround: Reboot the worker node on which the pod is currently running and stuck in a terminating state. This results in successful pod termination and subsequently related DR API resources are also garbage collected.

(BZ#2159791)

- **Regional DR CephFS based application failover show warning about subscription**
  After the application is failed over or relocated, the hub subscriptions show up errors stating, "Some resources failed to deploy. Use View status YAML link to view the details." This is because the application persistent volume claims (PVCs) that use CephFS as the backing storage provisioner, deployed using Red Hat Advanced Cluster Management for Kubernetes (RHACM) subscriptions, and are DR protected are owned by the respective DR controllers.

  Workaround: There are no workarounds to rectify the errors in the subscription status. However, the subscription resources that failed to deploy can be checked to make sure they are PVCs. This ensures that the other resources do not have problems. If the only resources in the subscription that fail to deploy are the ones that are DR protected, the error can be ignored.

  (BZ-2264445)

- **Disabled PeerReady flag prevents changing the action to Failover**
  The DR controller executes full reconciliation as and when needed. When a cluster becomes inaccessible, the DR controller performs a sanity check. If the workload is already relocated, this sanity check causes the **PeerReady** flag associated with the workload to be disabled, and the sanity check does not complete due to the cluster being offline. As a result, the disabled **PeerReady** flag prevents you from changing the action to Failover.

  Workaround: Use the command-line interface to change the DR action to Failover despite the disabled **PeerReady** flag.

  (BZ-2264765)

- **Ceph becomes inaccessible and IO is paused when connection is lost between the two data centers in stretch cluster**
  When two data centers lose connection with each other but are still connected to the Arbiter node, there is a flaw in the election logic that causes an infinite election between the monitors. As a result, the monitors are unable to elect a leader and the Ceph cluster becomes unavailable. Also, IO is paused during the connection loss.

  Workaround: Shutdown the monitors of any one of the data zone by bringing down the zone nodes. Additionally, you can reset the connection scores of surviving mon pods.

  As a result, monitors can form a quorum and Ceph becomes available again and IOs resume.

  (Partner BZ#2265992)

- **RBD applications fail to Relocate when using stale Ceph pool IDs from replacement cluster**
  For the applications created before the new peer cluster is created, it is not possible to mount the RBD PVC because when a peer cluster is replaced, it is not possible to update the CephBlockPoolID's mapping in the CSI configmap.

  Workaround: Update the **rook-ceph-csi-mapping-config** configmap with cephBlockPoolID's mapping on the peer cluster that is not replaced. This enables mounting the RBD PVC for the application.

  (BZ#2267731)

- **Information about lastGroupSyncTime is lost after hub recovery for the workloads which are primary on the unavailable managed cluster**
  Applications that are previously failed over to a managed cluster do not report a **lastGroupSyncTime**, thereby causing the trigger of the alert **VolumeSynchronizationDelay**. This is because when the ACM hub and a managed cluster that are part of the DRPolicy are unavailable, a new ACM hub cluster is reconstructed from the backup.

  Workaround: If the managed cluster to which the workload was failed over is unavailable, you can still failover to a surviving managed cluster.

  (BZ#2275320)

- **MCO operator reconciles the veleroNamespaceSecretKeyRef and CACertificates fields**
  When the OpenShift Data Foundation operator is upgraded, the **CACertificates** and **veleroNamespaceSecretKeyRef** fields under **s3StoreProfiles** in the Ramen config are lost.

  Workaround: If the Ramen config has the custom values for the **CACertificates** and **veleroNamespaceSecretKeyRef** fields, then set those custom values after the upgrade is performed.

  (BZ#2277941)

- **Instability of the token-exchange-agent pod after upgrade**
  The **token-exchange-agent** pod on the managed cluster is unstable as the old deployment resources are not cleaned up properly. This might cause application failover action to fail.

  Workaround: Refer the knowledgebase article, "token-exchange-agent" pod on managed cluster is unstable after upgrade to ODF 4.17.0.

  Result: If the workaround is followed, "token-exchange-agent" pod is stabilized and failover action works as expected.

  (BZ#2293611)

- **virtualmachines.kubevirt.io resource fails restore due to mac allocation failure on relocate**
  When a virtual machine is relocated to the preferred cluster, it might fail to complete relocation due to unavailability of the mac address. This happens if the virtual machine is not fully cleaned up on the preferred cluster when it is failed over to the failover cluster.

  Ensure that the workload is completely removed from the preferred cluster before relocating the workload.

  (BZ#2295404)

- **Relocating of CephFS gets stuck in WaitForReadiness**
  There is a scenario where the DRPC progression gets stuck in WaitForReadiness. If it remains in this state for an extended period, it's possible that a known issue has occurred, preventing Ramen from updating the PlacementDecision with the new Primary.

  As a result, the relocation process will not complete, leaving the workload undeployed on the new primary cluster. This can cause delays in recovery until the user intervenes.

  Workaround: Manually update the PlacementDecision to point to the new Primary.

- For workload using PlacementRule:

  1. Edit the PlacementRule:

```
$ oc edit placementrule --subresource=status -n [namespace] [name of the placementrule]7
```

For example:

```
$ oc edit placementrule --subresource=status -n busybox-workloads-cephfs-2  busybox-placement
```

2. Add the following to the placementrule status:

```
status:
  decisions:
  - clusterName: [primary cluster name]
    reason: [primary cluster name]
```

- For workload using Placement:

1. Edit the PlacementRule:

```
$ oc edit placementdecision --subresource=status -n [namespace] [name of the placementdecision]
```

For example:

```
$ oc get placementdecision --subresource=status -n openshift-gitops busybox-3-placement-cephfs-decision-1
```

2. Add the following to the placementrule status:

```
status:
  decisions:
  - clusterName: [primary cluster name]
    reason: [primary cluster name]
```

As a result, the PlacementDecision is updated and the workload is deployed on the Primary cluster.

(BZ#2319334)

- **Failover process fails when the`ReplicationDestination` resource has not been created yet**
If the user initiates a failover before the **LastGroupSyncTime** is updated, the failover process might fail. This failure is accompanied by an error message indicating that the **ReplicationDestination** does not exist.

Workaround:

Edit the **ManifestWork** for the VRG on the hub cluster.

Delete the following section from the manifest:

```
/spec/workload/manifests/0/spec/volsync
```

Save the changes.

Applying this workaround correctly ensures that the VRG skips attempting to restore the PVC using the **ReplicationDestination** resource. If the PVC already exists, the application uses it as is. If the PVC does not exist, a new PVC is created.

(BZ#2283038)

6.2. MULTICLOUD OBJECT GATEWAY

- **NooBaa Core cannot assume role with web identity due to a missing entry in the role's trust policy**
  For OpenShift Data Foundation deployments on AWS using AWS Security Token Service (STS), you need to add another entry in the trust policy for **noobaa-core** account. This is because with the release of OpenShift Data Foundation 4.17, the service account has changed from **noobaa** to **noobaa-core**.

  For instructions to add an entry in the trust policy for **noobaa-core** account, see the final bullet in the prerequisites section of Updating Red Hat OpenShift Data Foundation 4.16 to 4.17.

  (BZ#2322124)

- **Multicloud Object Gateway instance fails to finish initialization**
  Due to a race in timing between the pod code run and OpenShift loading the Certificate Authority (CA) bundle into the pod, the pod is unable to communicate with the cloud storage service. As a result, the default backing store cannot be created.

  Workaround: Restart the Multicloud Object Gateway (MCG) operator pod:

  ```
  $ oc delete pod noobaa-operator-<ID>
  ```

  With the workaround the backing store is reconciled and works.

  (BZ#2271580)

- **Upgrade to OpenShift Data Foundation 4.17 results in noobaa-db pod CrashLoopBackOff state**
  Upgrading to OpenShift Data Foundation 4.17 from OpenShift Data Foundation 4.15 fails when the PostgreSQL upgrade fails in Multicloud Object Gateway which always start with PostgresSQL version 15. If there is a PostgreSQL upgrade failure, the **NooBaa-db-pg-0** pod fails to start.

  Workaround: Refer to the knowledgebase article Recover NooBaa's PostgreSQL upgrade failure in OpenShift Data Foundation 4.17.

  (BZ#2298152)

6.3. CEPH

- **Poor performance of the stretch clusters on CephFS**
  Workloads with many small metadata operations might exhibit poor performance because of the arbitrary placement of metadata server (MDS) on multi-site Data Foundation clusters.

  (BZ#1982116)

- **SELinux relabelling issue with a very high number of files**
  When attaching volumes to pods in Red Hat OpenShift Container Platform, the pods sometimes

do not start or take an excessive amount of time to start. This behavior is generic and it is tied to how SELinux relabelling is handled by the Kubelet. This issue is observed with any filesystem based volumes having very high file counts. In OpenShift Data Foundation, the issue is seen when using CephFS based volumes with a very high number of files. There are different ways to workaround this issue. Depending on your business needs you can choose one of the workarounds from the knowledgebase solution https://access.redhat.com/solutions/6221251.

(Jira#3327)

6.4. CSI DRIVER

- **Automatic flattening of snapshots is not working**
  When there is a single common parent RBD PVC, if volume snapshot, restore, and delete snapshot are performed in a sequence more than 450 times, it is further not possible to take volume snapshot or clone of the common parent RBD PVC.

  To workaround this issue, instead of performing volume snapshot, restore, and delete snapshot in a sequence, you can use PVC to PVC clone to completely avoid this issue.

  If you hit this issue, contact customer support to perform manual flattening of the final restored PVCs to continue to take volume snapshot or clone of the common parent PVC again.

  (BZ#2232163)

6.5. OPENSHIFT DATA FOUNDATION CONSOLE

- **Optimize DRPC creation when multiple workloads are deployed in a single namespace**
  When multiple applications refer to the same placement, then enabling DR for any of the applications enables it for all the applications that refer to the placement.

  If the applications are created after the creation of the DRPC, the PVC label selector in the DRPC might not match the labels of the newer applications.

  Workaround: In such cases, disabling DR and enabling it again with the right label selector is recommended.

  (BZ#2294704)

6.6. OCS OPERATOR

- **Incorrect unit for the`ceph_mds_mem_rss` metric in the graph**
  When you search for the **ceph_mds_mem_rss** metrics in the OpenShift user interface (UI), the graphs show the y-axis in Megabytes (MB), as Ceph returns **ceph_mds_mem_rss** metric in Kilobytes (KB). This can cause confusion while comparing the results for the **MDSCacheUsageHigh** alert.

  Workaround: Use **ceph_mds_mem_rss * 1000** while searching this metric in the OpenShift UI to see the y-axis of the graph in GB. This makes it easier to compare the results shown in the **MDSCacheUsageHigh** alert.

  (BZ#2261881)

- **Increasing MDS memory is erasing CPU values when pods are in CLBO state**

When the metadata server (MDS) memory is increased while the MDS pods are in a crash loop back off (CLBO) state, CPU request or limit for the MDS pods is removed. As a result, the CPU request or the limit that is set for the MDS changes.

Workaround: Run the **oc patch** command to adjust the CPU limits.

For example:

```
$ oc patch -n openshift-storage storagecluster ocs-storagecluster \
    --type merge \
    --patch '{"spec": {"resources": {"mds": {"limits": {"cpu": "3"},
    "requests": {"cpu": "3"}}}}}'
```

(BZ#2265563)