Red Hat OpenShift Data Foundation 4.11

4.11 Release notes

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

Last Updated: 2023-11-02

Red Hat OpenShift Data Foundation 4.11 4.11 Release notes

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

Legal Notice

Abstract

The release notes for Red Hat OpenShift Data Foundation 4.11 summarizes all new features and enhancements, notable technical changes, and any known bugs upon general availability.

Table of Contents

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.

PROVIDING FEEDBACK ON RED HAT DOCUMENTATION

We appreciate your input on our documentation. Do let us know how we can make it better. To give feedback:

- For simple comments on specific passages:
  1. Make sure you are viewing the documentation in the *Multi-page HTML* format. In addition, ensure you see the **Feedback** button in the upper right corner of the document.
  2. Use your mouse cursor to highlight the part of text that you want to comment on.
  3. Click the **Add Feedback** pop-up that appears below the highlighted text.
  4. Follow the displayed instructions.
- For submitting more complex feedback, create a Bugzilla ticket:
  1. Go to the Bugzilla website.
  2. In the **Component** section, choose **documentation**.
  3. Fill in the **Description** field with your suggestion for improvement. Include a link to the relevant part(s) of documentation.
  4. Click **Submit Bug**.

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. OpenShift Data foundation also supports Red Hat OpenShift Data Foundation Logical Volume Manager as a technology preview feature for single node OpenShift clusters.

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 4.11 (RHSA-2022:6155 and RHSA-2022:6156) is now available. New enhancements, features, and known issues that pertain to OpenShift Data Foundation 4.11 are included in this topic.

Red Hat OpenShift Data Foundation 4.11 is supported on the Red Hat OpenShift Container Platform version 4.11. 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. ENHANCEMENTS

This section describes the major enhancements introduced in Red Hat OpenShift Data foundation 4.11.

Account credentials changeability option

With this release, you have an option that you can invoke externally to change the Multicloud Object Gateway (MCG) default account credentials. You can change and rotate credentials using the command-line interface to prevent issues with the applications. This option enables you to manage the credentials for all the service accounts in the system. For more details, see the Changing the default account credentials to ensure better security in the Multicloud Object Gateway.

Lifecycle policy for object expiration for the data buckets

Previously, the MCG was not aligned with newer lifecycle APIs. With this release update, MCG is now compatible with the lifecycle rest API and supports object expiration for the data buckets.

Termination policy added to the S3 route

Adding the **insecureEdgeTerminationPolicy** to the **s3-rout.yaml** adds the ability to change it from its default (Allow) to **Allow**, **Disable** or **Redirect**.

CHAPTER 3. REMOVED FUNCTIONALITY

This chapter lists functionalities that were supported in Red Hat OpenShift Data Foundation but are no longer available in OpenShift Data Foundation 4.11.

Multicloud Object Gateway Management console link

The external link to Multicloud Object Gateway (MCG) Management Console has been removed from the OpenShift Data Foundation console since the MCG console is no longer supported.

CHAPTER 4. TECHNOLOGY PREVIEWS

This section describes the technology preview features introduced in Red Hat OpenShift Data Foundation 4.11 under Technology Preview support limitations.

IMPORTANT

Technology Preview features are not supported with Red Hat production service level agreements (SLAs), might not be functionally complete, and Red Hat does not recommend using them for production. These features provide early access to upcoming product features, enabling customers to test functionality and provide feedback during the development process.

Technology Preview features are provided with a limited support scope, as detailed on the Customer Portal: Technology Preview Features Support Scope .

4.1. DISASTER RECOVERY FOR OPENSHIFT WORKLOADS

The OpenShift Data Foundation disaster recovery (DR) capability enables DR across multiple OpenShift Container Platform clusters, and is categorized as follows:

- **Regional disaster recovery (Regional-DR)**
  Regional-DR solution provides automated protection for block volumes, asynchronous replication, and protects business functionalities when a disaster strikes at a geographical location. In the public cloud this is similar to protecting from a region failure.

  For more information, see the planning guide and Regional-DR solution for OpenShift Data Foundation guide.

- **Metropolitan disaster recovery (Metro-DR)**
  Metro-DR solution ensures protection and business continuity during the unavailability of a data center with no data loss while using multiple clusters synchronous replication. In the public cloud these are similar to protecting from an Availability Zone failure. It offers instant protection of business functionalities, with a near zero RPO.

  For more information, see the planning guide and Metro-DR solution for OpenShift Data Foundation guide.

- **Multicluster monitoring in Red Hat Advanced Cluster Management console**
  Multicluster monitoring is a single simplified view of storage health and capacity spread across multiple clusters. This multicluster monitoring enables you to manage the storage capacity and monitor the OpenShift Data Foundation clusters from the Red Hat Advanced Cluster Management (RHACM) user interface. This monitoring capability applies to both DR and non-DR clusters.

  For more information, see Monitoring multicluster storage health .

4.2. SUPPORT FOR NETWORK FILE SYSTEM SERVICES

With this release, OpenShift Data Foundation supports the Network File System (NFS) service for any internal or external applications running in any operating system (OS), including Windows OS. The NFS service helps to migrate data from any environment to the OpenShift environment, for example, data migration from Red Hat Gluster Storage file system to OpenShift environment.

For more information, see Resource requirements for using Network File system and Creating exports using NFS.

4.3. THIN-PROVISIONING SOLUTION FOR SINGLE NODE OPENSHIFT CLUSTER

When OpenShift Data Foundation Logical Volume Manager Operator is set up, the volume group creates thin-provisioned volumes by default. This benefits the Edge users that are limited in storage capacity.

The thin-provisioned volumes use the Logical Volume Manager and provide dynamic provisioning of block storage on a single, limited resources single node OpenShift (SNO) cluster.

For more information, see Provisioning storage using Logical Volume Manager Operator .

4.4. VOLUME SNAPSHOT AND VOLUME CLONE FOR SINGLE NODE OPENSHIFT

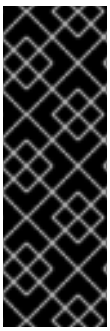
With this release, OpenShift Data Foundation for single node OpenShift provides features similar to the Rook-Ceph based OpenShift Data Foundation:

- Creating volume snapshots of persistent volumes that are provisioned by the OpenShift Data Foundation Logical Volume Manager Operator. This enables you to revert to a state at which the volume snapshot was taken.
- Creating volume snapshots of the cloned volumes.
- Restoring the volume snapshots, which creates a new Persistent Volume Claim (PVC).
- Creating clones of a volume to duplicate an existing storage volume that can be used like any standard volume.

For more information, see Volume Snapshot and Volume Clone for Single Node OpenShift.

CHAPTER 5. DEVELOPER PREVIEWS

This section describes the developer preview features introduced in Red Hat OpenShift Data Foundation 4.11.



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.

5.1. DELETION OF EXPIRED OBJECTS IN THE MULTICLOUD OBJECT GATEWAY LIFECYCLE

Deletion of expired objects is a simplified way that enables handling of unused data. The unused data is handled either by moving it to a different location or by deleting it. This can be achieved by an Multicloud Object Gateway (MCG) lifecycle feature where you can use data tiering and data expiration features. Tiering enables tiering of data transparently to another backing storage. The entire data set would be available for the application regardless of its location. Data expiration is a part of Amazon Web Services (AWS) lifecycle management and sets an expiration date for automatic deletion. The minimal time resolution of the lifecycle expiration is one day.

For more information on data expiration, refer to the AWS lifecycle management.

5.2. INSTALLING 3-NODE COMPACT MODE CLUSTER USING RED HAT ADVANCED CLUSTER MANAGER POLICY

With this release, OpenShift Data Foundation provides limited support to three node (3-node) compact mode clusters for all advanced storage features using the Red Hat Advanced Cluster Management (RHACM) policy.

NOTE

This deployment support is limited to the bare metal platforms with local disks dedicated to OpenShift Data Foundation.

This provides the ability to deploy Rados Block Device (RBD)-only OpenShift Data Foundation clusters to conserve the resources where ReadWriteMany (RWX) and Object is not currently needed. This option enables a 3-node cluster that behaves like a full-fledged mobile data center that is simple to install.

For instructions about how to install a 3-node OpenShift Data Foundation compact cluster using RHACM policy, see the Red Hat Knowledgebase solutions for all clusters and IBM Power specific deployments.

CHAPTER 6. BUG FIXES

This section describes the notable bug fixes introduced in Red Hat OpenShift Data Foundation 4.11.

6.1. MULTICLOUD OBJECT GATEWAY

- **Multicloud Object Gateway is making API requests to the incorrect URL when OpenShift Data Foundation is deployed on Microsoft Azure Government (MAG)**
  Previously, the default backing store was faulty, and the overall system health was degraded. This is because the Multicloud Object Gateway (MCG) creates a default backing store on the platform where OpenShift Data Foundation was deployed. The default backing store was created with the wrong values due to an issue in identifying the Microsoft Azure Government (MAG).

  With this update, a Persistent Volume (PV) pool is created when identifying the MAG instead of creating the usual Azure backing store, and later on, you can create a backing store on top of the MAG environment and use that new backing store.

  (BZ#1944572)

- **The default backing store 'noobaa-default-backing-store' is stuck in the 'Creating' state**
  Previously, the Cloud Credentials Operator (CCO) changed credentials after the creation of the default backing store. Due to this, the backing store used to get stuck. The update credentials on the default backing store are fixed with this release. The default Multicloud Object Gateway (MCG) backing store does not get stuck on creating while changing credentials.

  (BZ#1992090)

- **S3 listing goes into an infinite loop, when the prefixes are of names `99` and `990`**
  Previously, the query results for the listing operation were not in the order that the Multicloud Object Gateway (MCG) expected them to be as the collate algorithm was set incorrectly while starting the MCG DB.

  With this update, the collate algorithm is set correctly on new deployments while starting the MCG DB. Now, the query results are in the expected order.

  (BZ#2068110)

6.2. CEPHFS

- **Usage of MD5 in a FIPS-enabled environment is explicitly allowed and S3 multipart operations can be completed**
  Previously, in a FIPS-enabled environment, the usage of MD5 digest was not allowed by default, unless explicitly excluded for non-cryptographic purposes. Due to this, a segfault occurred during the S3 complete multipart upload operation.

  With this fix, the usage of MD5 for non-cryptographic purposes in a FIPS-enabled environment for S3 complete multipart **PUT** operations is explicitly allowed and the S3 multipart operations can be completed.

  (BZ#2086724)

- **OpenShift Data Foundation pods are restarting due to liveness probe failure**

Previously, the liveness probe on pods caused a restart of Ceph pods. This release update increases the default timeout for the liveness probe. The pods now get more time before restarting due to liveness where the nodes have more loads and fewer CPU/memory resources.

(BZ#2091951)

6.3. ROOK

- **OSD pods are in CrashLoopBackOff state after the upgrade to OpenShift Data Foundation 4.10.3 failed to deactivate the volume group for the local volume**
  Previously, LVM-based OSDs from older OpenShift Data Foundation releases (4.3, 4.4) were incompatible with version 4.10 and would crash when upgraded to 4.10. With this update, the OSDs from older OpenShift Data Foundation releases are running instead of crashing. Rook now adds the correct flag to handle these legacy OSDs so that they will be functional with 4.10.

  (BZ#2097268)

CHAPTER 7. KNOWN ISSUES

This section describes the known issues in Red Hat OpenShift Data Foundation 4.11.

7.1. DISASTER RECOVERY

- **Creating an application namespace for the managed clusters**
  Application namespace needs to exist on RHACM managed clusters for disaster recovery (DR) related pre-deployment actions and hence is pre-created when an application is deployed at the RHACM hub cluster. However, if an application is deleted at the hub cluster and its corresponding namespace is deleted on the managed clusters, they reappear on the managed cluster.

  Workaround: **openshift-dr** maintains a namespace **manifestwork** resource in the managed cluster namespace at the RHACM hub. These resources need to be deleted after the application deletion. For example, as a cluster administrator, execute the following command on the hub cluster: **oc delete manifestwork -n <managedCluster namespace> <drPlacementControl name>-<namespace>-ns-mw**.

  (BZ#2059669)

- **Failover action reports RADOS block device image mount failed on the pod with RPC error still in use**
  Failing over a disaster recovery (DR) protected workload might result in pods using the volume on the failover cluster to be stuck in reporting RADOS block device (RBD) image is still in use. This prevents the pods from starting up for a long duration (upto several hours).

  (BZ#2007376)

- **Failover action reports RADOS block device image mount failed on the pod with RPC error fsck**
  Failing over a disaster recovery (DR) protected workload may result in pods not starting with volume mount errors that state the volume has file system consistency check (fsck) errors. This prevents the workload from failing over to the failover cluster.

  (BZ#2021460)

- **Relocation fails when failover and relocate is performed within a few minutes of each action**
  When the user starts relocating an application from one cluster to another before the **PeerReady** condition status is **TRUE**, the condition status is seen through the DRPC YAML file or by running the following **oc** command:

  ```
  $ oc get drpc -o yaml -n <application-namespace>
  ```

  where **<application-namespace>** is the namespace where the workloads are present for deploying the application.

  If the Relocation is initiated before the peer (target cluster) is in a clean state, then the relocation will stall forever.

  Workaround: Change the DRPC **.Spec.Action** back to **Failover**, and wait until the **PeerReady** condition status is TRUE. After applying the workaround, change the Action to Relocate, and the relocation will take effect.

(BZ#2056871)

- **User is able to set the value to zero minutes as the Sync schedule while creating DR Policy and it reports 'Sync' as Replication policy and gets validated on a Regional-DR setup**
  The **DRPolicyList** page uses the **sync** interval value to display the replication type. If it is set to zero then the replication type is considered as Sync(synchronous) for the metro as well as regional clusters. This issue confuses the users because the backend is considering **Async** even when the user interface shows it as **Sync** scheduling type.

  Workaround: Need to fetch Ceph **Fsid** from DRCluster CR status to decide **sync** or **async**.

  (BZ#2114501)

- **Deletion of the Application deletes the pods but not PVCs**
  When deleting an application from the RHACM console, DRPC does not get deleted. Not deleting DRPC leads to not deleting the VRG as well as the VR. If the VRG/VR is not deleted, the PVC finalizer list will not be cleaned up, causing the PVC to stay in a **Terminating** state.

  Workaround: Manually delete DRPC on the hub cluster using the following command

  ```
  $ oc delete drpc <name> -n <namespace>
  ```

  Result:

  1. DRPC deletes the VRG
  2. VRG deletes VR
  3. VR removes its finalizer from the PVC's finalizer list
  4. VRG removes its finalizer from the PVC's finalizer list

  (BZ#2108716)

- **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#2111163)

- **RBD mirror scheduling is getting stopped for some images**
  The Ceph manager daemon gets blocklisted due to different reasons, which causes the scheduled RBD mirror snapshot from being triggered on the cluster where the image(s) are primary. All RBD images that are mirror enabled (hence DR protected) do not list a schedule when examined using **rbd mirror snapshot schedule status -p ocs-storagecluster-cephblockpool**, and hence are not actively mirrored to the peer site.

  Workaround: Restart the Ceph manager deployment, on the managed cluster where the images are primary, to overcome the blocklist against the currently running instance, this can be done by scaling down and then later scaling up the ceph manager deployment as follows:

  ```
  $ oc -n openshift-storage scale deployments/rook-ceph-mgr-a  --replicas=0
  $ oc -n openshift-storage scale deployments/rook-ceph-mgr-a  --replicas=1
  ```

  Result: Images that are DR enabled and denoted as primary on a managed cluster start reporting mirroring schedules when examined using **rbd mirror snapshot schedule status -p ocs-storagecluster-cephblockpool**

  (BZ#2067095)

- **Ceph does not recognize the global IP assigned by Globalnet**
  Ceph does not recognize global IP assigned by Globalnet, so disaster recovery solution cannot be configured between clusters with overlapping service CIDR using Globalnet. Due to this disaster recovery solution does not work when service **CIDR** overlaps.

  (BZ#2102397)

- **Volume replication group deletion is stuck on a fresh volume replication created during deletion, which is stuck as the persistent volume claim cannot be updated with a finalizer**
  Due to a bug in the disaster recovery (DR) reconciler, during deletion of the internal **VolumeReplicaitonGroup** resource on a managed cluster, from where a workload failed over or relocated from, a persistent volume claim (PVC) is attempted to be protected. The resulting cleanup operation does not complete and reports the **PeerReady** condition on the **DRPlacementControl** for the application.

  This results in the application that was failed over or relocated, cannot be relocated or failed over again due to **DRPlacementControl** resource reporting its **PeerReady** condition as **false**.

  Workaround: Before applying the workaround, determine if the cause is due to protecting a PVC during **VolumeReplicationGroup** deletion as follows:

  1. Ensure the **VolumeReplicationGroup** resource in the workload namespace on the managed cluster from where it was relocated or failed over from has the following values:
     - VRG **metadata.deletionTimestamp** is **non-zero**
     - VRG **spec.replicationState** is **Secondary**
  2. List the **VolumeReplication** resources in the workload namespace as above, and ensure the resource have the following values:
     - **metadata.generation** is set to **1**
     - **spec.replicationState** is set to **Secondary**
     - The VolumeReplication resource reports no status

3. For each VolumeReplication resource in the above state, their corresponding PVC resource (as seen in the VR **spec.dataSource** field) should have the values **metadata.deletionTimestamp** as **non-zero**

4. To recover, remove the finalizer

    - **volumereplicationgroups.ramendr.openshift.io/vrg-protection** from the VRG resource

    - **volumereplicationgroups.ramendr.openshift.io/pvc-vr-protection** from the respective PVC resources

Result: **DRPlacementControl** at the hub cluster reports **PeerReady** condition as **true** and enables further workload relocation or failover actions. (BZ#2116605)

- **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#2114573)

- **While failover to secondary cluster, some of PVC remains in Primary cluster**
The behavior before Kubernetes v1.23 was that the Kubernetes control plane never cleaned up the PVCs created for StatefulSets. That's left to the cluster administrator or a software operator managing the StatefulSets. Due to this, the PVCs of the StatefulSets were left untouched when their Pods are deleted. This prevents Ramen from failing back an application to its original cluster.

  Workaround: If the workload uses StatefulSets, then do the following before failing back or relocating to another cluster

  1. Run **oc get drpc -n <namespace> -o wide**

  2. If the PeerReady column shows "TRUE" then you can proceed with the failback or relocation. Otherwise, do the following on the peer cluster:

      a. Run **oc get pvc -n <namespace>**

      b. For each bounded PVC for that namespace that belongs to the StatefulSet, run **oc delete pvc <pvcname> -n namespace**

      c. Once all PVCs are deleted, Volume Replication Group (VRG) transitions to secondary, and then gets deleted.

  3. Run the following command again **oc get drpc -n <namespace> -o wide**. After a few seconds to a few minutes, the PeerReady column changes to **TRUE**. Then you can proceed with the failback or relocation.

  Result: The peer cluster gets cleaned up and ready for new 'Action'. (BZ#2118270)

- **Application is stuck in Relocating state during failback**

Multicloud Object Gateway allowed multiple persistent volume (PV) objects of the same name or namespace to be added to the S3 store on the same path. Due to this, Ramen does not restore the PV because it detected multiple versions pointing to the same **claimRef**.

Workaround: Use S3 CLI or equivalent to clean up the duplicate PV objects from the S3 store. Keep only the one that has a timestamp closer to the failover or relocate time.

Result: The restore operation will proceed to completion and the failover or relocate operation proceeds to the next step.

(BZ#2120201)

- **Application is stuck in a FailingOver state when a zone is down**
  At the time of a failover or relocate, if none of the s3 stores are reachable then the failover or relocate process hangs. If the Openshift DR logs indicate that the S3 store is not reachable, then troubleshooting and getting the s3 store operational will allow the OpenShift DR to proceed with the failover or relocate operation.

  (BZ#2121680)

- **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)

7.2. MULTICLOUD OBJECT GATEWAY

- **rook-ceph-operator-config ConfigMap is not updated when OpenShift Container Storage is upgraded from version 4.5 to other version**
  ocs-operator uses the rook-ceph-operator-config ConfigMap to configure rook-ceph-operator behaviors, however it only creates it once and then does not reconcile it. This raises the problem that it will not update the default values for the product as they evolve.

  Workaround: Administrators can manually change the rook-ceph-operator-config values.

  (BZ#1986016)

- **Storage cluster and storage system ocs-storagecluster is in an error state for a few minutes when installing the storage system**
  During storage cluster creation, there is a small window of time where it will appear in an error state before moving on to a successful or ready state. This is an intermittent state, so it will usually resolve by itself and become successful or ready.

  Workaround: Wait and watch status messages or logs for more information.

  (BZ#2004027)

7.3. CEPHFS

- **Ceph OSD snap trimming is no longer blocked by a running scrub**
  Previously, OSD snap trimming, once blocked by a running scrub, was not restarted. As a result, no trimming was performed until an OSD reset. This release fixes the handling of restarting the trimming if blocked after the scrub and snap trimming works as expected.

(BZ#2067056)

- **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 OpenShift Data Foundation clusters.

  (BZ#1982116)

- **Restoring snapshot fails with size constraint when the parent PVC is expanded**
  If we create a new restored persistent volume claim (PVC) from a volume snapshot whose size is the same as the volume snapshot, then it will fail if the parent PVC is resized after taking the volume snapshot and before creating the new restored PVC.

  Workaround: You can use any one of the following workarounds

  - Do not resize the parent PVC if you have any volume snapshot created from it and you have a plan to restore the volume snapshot to a new PVC.
  - Create a restored PVC of the same size as the parent PVC.
  - If the restored PVC is already created and is in the pending state, delete the PVC and recreate it with the same size as the parent PVC.

  (BZ#2120730)

7.4. OPENSHIFT DATA FOUNDATION OPERATOR

- **PodSecurityViolation alert starts to fire when the OpenShift Data Foundation operator is installed**
  OpenShift introduced Pod Security Admission to enforce security restrictions on Pods when scheduled such that OpenShift 4.11 has an audit and warn events with enforcing privileged (same as 4.10).

  As a result, you will see warnings in events since the **openshift-storage** namespace doesn't have the required enforcement labels for Pod Security Admission.

  (BZ#2110628)

CHAPTER 8. ASYNCHRONOUS ERRATA UPDATES

8.1. RHBA-2023:6176 OPENSHIFT DATA FOUNDATION 4.11.12 BUG FIXES AND SECURITY UPDATES

OpenShift Data Foundation release 4.11.12 is now available. The bug fixes that are included in the update are listed in the RHBA-2023:6176 advisory.

8.2. RHBA-2023:5393 OPENSHIFT DATA FOUNDATION 4.11.11 BUG FIXES AND SECURITY UPDATES

OpenShift Data Foundation release 4.11.11 is now available. The bug fixes that are included in the update are listed in the RHBA-2023:5393 advisory.

8.3. RHBA-2023:4775 OPENSHIFT DATA FOUNDATION 4.11.10 BUG FIXES AND SECURITY UPDATES

OpenShift Data Foundation release 4.11.10 is now available. The bug fixes that are included in the update are listed in the RHBA-2023:4775 advisory.

8.4. RHSA-2023:4238 OPENSHIFT DATA FOUNDATION 4.11.9 BUG FIXES AND SECURITY UPDATES

OpenShift Data Foundation release 4.11.9 is now available. The bug fixes that are included in the update are listed in the RHSA-2023:4238 advisory.

8.5. RHBA-2023:3293 OPENSHIFT DATA FOUNDATION 4.11.8 BUG FIXES AND SECURITY UPDATES

OpenShift Data Foundation release 4.11.8 is now available. The bug fixes that are included in the update are listed in the RHBA-2023:3293 advisory.

8.6. RHSA-2023:2023 OPENSHIFT DATA FOUNDATION 4.11.7 BUG FIXES AND SECURITY UPDATES

OpenShift Data Foundation release 4.11.7 is now available. The bug fixes that are included in the update are listed in the RHSA-2023:2023 advisory.

8.7. RHBA-2023:1230 OPENSHIFT DATA FOUNDATION 4.11.6 BUG FIXES AND SECURITY UPDATES

OpenShift Data Foundation release 4.11.6 is now available. The bug fixes that are included in the update are listed in the RHBA-2023:1230 advisory.

8.8. RHBA-2023:0764 OPENSHIFT DATA FOUNDATION 4.11.5 BUG FIXES AND SECURITY UPDATES

OpenShift Data Foundation release 4.11.5 is now available. The bug fixes that are included in the update are listed in the RHBA-2023:0764 advisory.

With this release, OpenShift Data Foundation 4.10 users can upgrade to OpenShift Data Foundation 4.11, which was earlier not possible due to object storage data accessibility issues.

8.9. RHBA-2022:8877 OPENSHIFT DATA FOUNDATION 4.11.4 BUG FIXES AND SECURITY UPDATES

OpenShift Data Foundation release 4.11.4 is now available. The bug fixes that are included in the update are listed in the RHBA-2022:8877 advisory.

8.10. RHBA-2022:7912 OPENSHIFT DATA FOUNDATION 4.11.3 BUG FIXES AND SECURITY UPDATES

OpenShift Data Foundation release 4.11.3 is now available. The bug fixes that are included in the update are listed in the RHBA-2022:7912 advisory.

8.11. RHBA-2022:6888 OPENSHIFT DATA FOUNDATION 4.11.2 BUG FIXES AND SECURITY UPDATES

OpenShift Data Foundation release 4.11.2 is now available. The bug fixes that are included in the update are listed in the RHBA-2022:6888 advisory.

8.12. RHBA-2022:6525 OPENSHIFT DATA FOUNDATION 4.11.1 BUG FIXES AND SECURITY UPDATES

OpenShift Data Foundation release 4.11.1 is now available. The bug fixes that are included in the update are listed in the RHBA-2022:6525 advisory.