ibm.com/redbooks

IBM Spectrum Scale as a Persistent Storage for Red Hat OpenShift on IBM Z: Quick Installation Guide

IBM Garage Technical Enablement Series

Narjisse Zaki

Storage

Redpaper

IBM Redbooks

IBM Spectrum Scale as a Persistent Storage for Red Hat OpenShift on IBM Z Quick Installation Guide IBM Garage Technical Enablement Series

August 2021

REDP-5645-00

Note: Before using this information and the product it supports, read the information in “Notices” on page v.

First Edition (August 2021)

This edition applies to IBM Spectrum Scale Release 5.1.0.1 and higher, Red Hat OpenShift Container Platform 4.6.6 and higher.

This document was created or updated on August 6, 2021.

Contents

Notices

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:
IBM Director of Licensing, IBM Corporation, North Castle Drive, MD-NC119, Armonk, NY 10504-1785, US

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at http://www.ibm.com/legal/copytrade.shtml

The following terms are trademarks or registered trademarks of International Business Machines Corporation, and might also be trademarks or registered trademarks in other countries.

IBM®
IBM Garage™
IBM Spectrum®
IBM Z®
Passport Advantage®
Redbooks®
Redbooks (logo) ®
z/VM®

The following terms are trademarks of other companies:

The registered trademark Linux® is used pursuant to a sublicense from the Linux Foundation, the exclusive licensee of Linus Torvalds, owner of the mark on a worldwide basis.

Red Hat and OpenShift are trademarks or registered trademarks of Red Hat, Inc. or its subsidiaries in the United States and other countries.

Other company, product, or service names may be trademarks or service marks of others.

Preface

IBM® Spectrum Scale, formerly named IBM General Parallel File System (GPFS), is a high-performance clustered file system software, that allows concurrent access to a single filesystem or set of filesystems from multiple nodes. IBM Spectrum® Scale Container Native Storage Access (CNSA) is the containerized version of IBM Spectrum Scale that provides a persistent file system for the latest generation of applications (called "containerized applications"), which takes advantage of high performance and benefits of IBM Spectrum Scale in a cloud containerized application world. IBM Spectrum Scale CNSA was introduced with IBM Spectrum Scale version 5.1.

This document describes the steps to deploy IBM Spectrum Scale in containers (CNSA) on Red Hat OpenShift Container Platform (OCP) running on IBM Z®. This Spectrum Scale cluster, called the "Local" storage cluster, will mount a filesystem from a second non-containerized Spectrum Scale storage cluster, called the "Remote" storage cluster.

This document also describes practical information for deploying IBM Spectrum Scale Container Storage Interface (CSI) driver on Red Hat OCP. This CSI driver allows Red Hat OCP Persistent Volumes (PV) to be provisioned from IBM Spectrum Scale. It handles storage provisioning requests at the OCP cluster level and makes API calls on the storage server to create the necessary volumes required by the applications.

Finally, this document contains the steps that explain how to use the CSI driver, after it is deployed, to set up dynamic provisioning of PVs on Red Hat OCP from IBM Spectrum Scale.

It is assumed that the reader has a good understanding of Red Hat OCP administration and IBM Spectrum Scale solution.

Authors

This paper was produced by a team of specialists from around the world working at IBM Garage™, Montpellier France.

Narjisse Zaki is an IT Architect in IBM Advanced Technology Services (ATS). She provides pre-sales technical support for Linux consolidation projects on Linux on Z and LinuxONE across Europe, Middle East, Africa (EMEA) countries, and more recently, she supports Red Hat OpenShift client engagements. She is based in Montpellier, France. Narjisse is also the engagement leader for LinuxONE pre-sales activities in Middle East and Africa (MEA) at the IBM Systems Center, Montpellier. In past years, Narjisse supported Oracle consolidation projects within the European IBM Oracle Center (IOC). Before joining the IOC, Narjisse was involved in Agile Project Management leading projects as a Scrum Leader. She has been an international speaker in several conferences and events around the world. She holds a Master of Engineering in Computer Science and Management.

Thanks to the following people for their contributions to this project:

Sylvain Carta
IBM z/VM® and Linux Specialist, ATS, IBM Montpellier

Gauthier Siri
Storage Specialist, Demo maker IBM Montpellier

Jérôme Farenq
IT Specialist, IBM IBM Montpellier

Now you can become a published author, too!

Here's an opportunity to spotlight your skills, grow your career, and become a published author—all at the same time! Join an IBM Redbooks® residency project and help write a book in your area of expertise, while honing your experience using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run from two to six weeks in length, and you can participate either in person or as a remote resident working from your home base.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

We want our papers to be as helpful as possible. Send us your comments about this paper or other IBM Redbooks publications in one of the following ways:

- Use the online **Contact us** review Redbooks form found at:

  ibm.com/redbooks

- Send your comments in an email to:

  redbooks@us.ibm.com

- Mail your comments to:

  IBM Corporation, IBM Redbooks
  Dept. HYTD Mail Station P099
  2455 South Road
  Poughkeepsie, NY 12601-5400

Stay connected to IBM Redbooks

- Find us on LinkedIn:

  http://www.linkedin.com/groups?home=&gid=2130806

- Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:

https://www.redbooks.ibm.com/Redbooks.nsf/subscribe?OpenForm

- Stay current on recent Redbooks publications with RSS Feeds:

  http://www.redbooks.ibm.com/rss.html

1

Overview and preparation

This chapter provides information on how to prepare your environment to install IBM Spectrum Scale CNSA on the Red Hat OpenShift Container Platform running on IBM Z. It also includes an architectural view of the components of this solution and information on the preparation tasks.

1.1 Global view

Figure 1-1 shows a high-level overview of IBM Spectrum Scale Container Native Storage Access (CNSA) and IBM Spectrum Scale Container Storage Interface (CSI) driver, which are implemented in this document, and their interactions with the non-containerized IBM Spectrum Scale storage server. We will install IBM Spectrum Scale CNSA and IBM Spectrum Scale CSI driver in the worker nodes of Red Hat OpenShift Container Platform (OCP). This local storage cluster mounts a filesystem from another remote IBM Spectrum Scale server, which is called "Remote storage cluster" in Figure 1-1. This filesystem will be mounted in the worker nodes under `/mnt`. The CSI driver allows you to perform data operations and make calls to the remote storage server to create volumes to serve to the containerized applications.

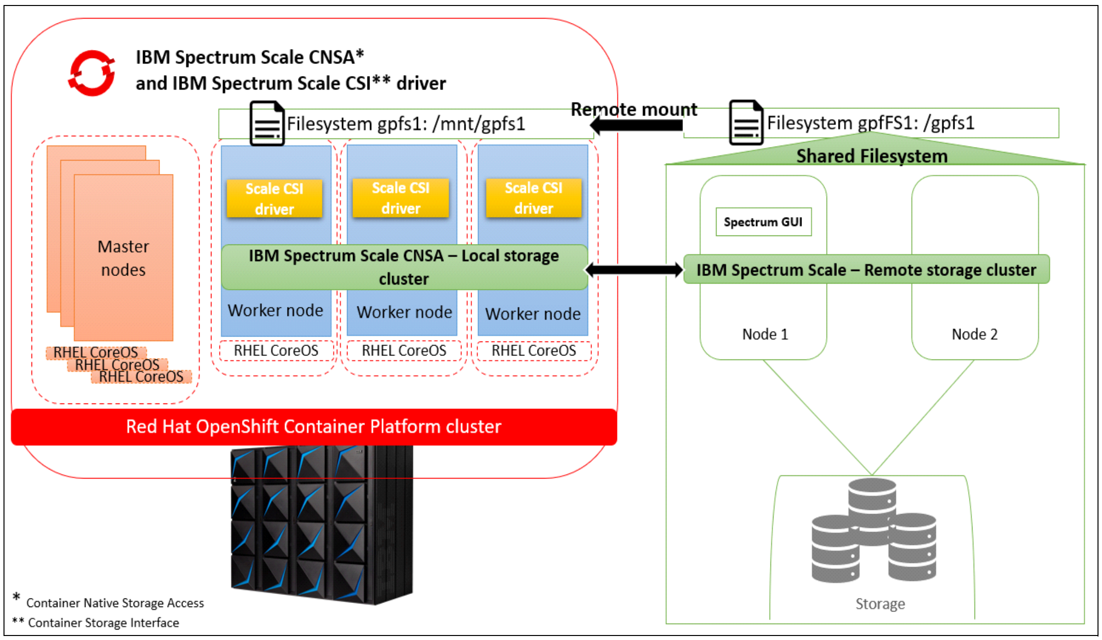


Figure 1-1 IBM Spectrum Scale CNSA and IBM Spectrum Scale CSI driver on Red Hat OCP

1.2 Hardware and software requirements

The hardware and software requirements for IBM Spectrum Scale CNSA installation on Red Hat OCP on Z are as follows:

- IBM Spectrum Scale remote storage server release 5.1.0.1 or higher with an installed GUI.
- Red Hat OCP 4.6.6 or higher with a minimum of three worker nodes with the specifications shown in Table 1-1.

Table 1-1 Worker node specifications

Resources per worker	CPU (min)	CPU (recommended)	Memory GB (min)	Memory GB (recommended)
Worker VM	4	8	8	16

- Podman.
- Network communication between all nodes in the IBM Spectrum Scale local storage server and the IBM Spectrum Scale remote storage server [1].

The current limitations are:

- Encrypted filesystem is not supported.
- Local storage is not supported.
- Support of one remote storage server and one remote filesystem.
- Support for a maximum of 128 worker nodes.

1.3 Prerequisite tasks

To prepare for installation, several prerequisite tasks must be completed.

1.3.1 Preparation

First, we need to allow the OCP worker nodes to resolve the nodes of the remote storage cluster. In our environment, the remote Spectrum Scale storage cluster is composed of two nodes, `spscale1` and `spscale2`.

Follow your own processes to update your Domain Name System (DNS) with new entries. The following steps are used in our environment to update the DNS configuration, and are therefore specific to the environment used in the context of this document.

1. Add the remote nodes to the existing DNS.

```
[root@opnshrh8~]# vi /var/named/test.mop.fr.ibm.com.zone
spscale1.mop.fr.ibm.com                IN A 10.3.57.15
spscale2.mop.fr.ibm.com                IN A 10.3.57.7

[root@opnshrh8]# vi /var/named/57.3.10.in-addr.arpa.zone
15 IN PTR spscale1.mop.fr.ibm.com.
7 IN PTR spscale2.mop.fr.ibm.com.
```

2. Restart the DNS.

```
[root@opnshrh8~]# systemctl restart named-chroot
```

3. Log in to each worker node and make sure you can ping the remote storage nodes.

```
[root@opnshrh8~]# ssh core@10.3.57.220
[core@worker-0 ~]$ ping spscale2.mop.fr.ibm.com
```

4. Login to the bastion-node VM and create a spectrum scale directory, where the IBM Spectrum Scale CNSA `.tar` archive file will be stored.

```
[root@opnshrh8~]# mkdir spectrum
```

5. Obtain the IBM Spectrum Scale CNSA `.tar` archive file from one of the following places:

   – IBM Fix Center at:
   https://www.ibm.com/support/fixcentral/options?selectionBean.selectedTab=find&selection=System+Storage%3bStorage+software%3bSoftware+defined+storage%3bibm%2fStorageSoftware%2fIBM%C2%AE+Spectrum+Scale

[1] A minimum of 10 Gb network is needed but 40 to 100 Gb is recommended. RDMA for Infiniband and RoCE for Ethernet are not supported.

– IBM Passport Advantage at:
https://www.ibm.com/software/passportadvantage/pao_customer.html

This archive contains the files and images that are needed to deploy IBM Spectrum Scale CNSA on Red Hat OCP, as shown in the following example.

```
[root@opnshrh8 ~]# cd spectrum
[root@opnshrh8 spectrum]# wget
https://ak-delivery04-mul.dhe.ibm.com/sdfdl/v2/sar/CM/SS/09cpu/0/Xa.2/Xb.jusyLT
Sp44SOMfK3we8r7mZvJkCpy7UeqM1PErPB/Xc.CM/SS/09cpu/0/Spectrum_Scale_Container_Na
tive_Storage_Access-5.1.0.1-s390x-Linux.tgz/Xd./Xf.Lpr./Xg.11107384/Xi.habanero
/XY.habanero/XZ.DNTy5EIZhPJvuGQddke6Roaj2-RbwNaN/Spectrum_Scale_Container_Nativ
e_Storage_Access-5.1.0.1-s390x-Linux.tgz
```

6. Extract the archive in the spectrum directory that was created in step 4.

```
[root@opnshrh8 spectrum]# tar xvfz
Spectrum_Scale_Container_Native_Storage_Access-5.1.0.1-s390x-Linux.tgz
```

7. Login to OCP cluster and create a new project named `ibm-spectrum-scale-ns`. In this namespace, we will deploy the component to install the IBM Spectrum Scale CNSA.

```
[root@opnshrh8 spectrum]# oc new-project ibm-spectrum-scale-ns
```

1.3.2 Red Hat OCP configuration

This section describes how to configure Red Hat OCP to allow Spectrum Scale CNSA to be installed and run properly.

The general configuration steps, described in this section, are as follows:

1. “Increase the PID_LIMITS”
2. “Increase the Vmalloc kernel parameter” on page 5
3. “Modify the Machine Config Operator” on page 6

Increase the PID_LIMITS

For each Red hat OCP worker node that hosts the IBM Spectrum Scale, increase the `PID_LIMITS` to 4096 to prevent crashes related to “out of PID (Process Identifier) resource” conditions during I/O operations by the GPFS daemon. To achieve this change, we use the `spectrumscale/preinstall/increase_pid_mco.yaml` file that is contained in the archive file that was extracted in step 6 on page 4.

To increase the **PID_LIMITS**, follow these steps:

1. Ensure that you are in the new spectrum folder.

```
[root@opnshrh8~]# cd /root/spectrum
```

2. Apply this configuration to Red Hat OCP by using the following command.

```
[root@opnshrh8 spectrum]# oc create -f
spectrumscale/preinstall/increase_pid_mco.yaml

containerruntimeconfig.machineconfiguration.openshift.io/increase-pid-limit
created
```

3. The **create** command creates a `ContaineRuntimeConfig` object. The following command allows you to verify that this object was created successfully.

```
[root@opnshrh8 spectrum]# oc get ContainerRuntimeConfig
NAME                    AGE
```

```
increase-pid-limit   31s
```

Note: The changes applied to the Red Hat OCP worker nodes are managed by the Machine Config Operator (MCO).

4. Apply the object to modify the `MachineConfigPool` of the Red Hat OCP worker nodes. This command starts an OCP rolling-update to apply the desired changes. Wait until the rolling update is complete. This might take some time depending on the size of your worker nodes.

   ```
   [root@opnshrh8 spectrum]# oc label machineconfigpool worker pid-crio=config-pid

   machineconfigpool.machineconfiguration.openshift.io/worker labeled
   ```

5. You can check the status of the by update by using the following command.

   [root@opnshrh8 spectrum]# oc get MachineConfigPool

   When the update is complete, the following parameters are set for the worker nodes (Figure 1-2):

   – The `UPDATING` parameter is equal to `False`.

   – The `UPDATED` parameter is equal to `True`.

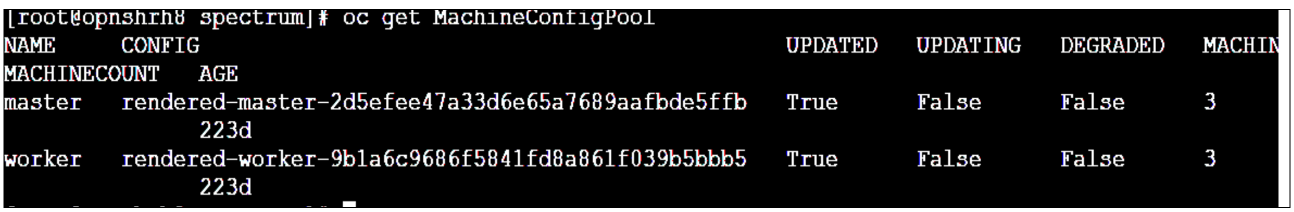


Figure 1-2 Update status

6. Verify that the changes have been taken into account for each worker by using the following command.

   The `pids_limit value` must be equal to `4096`, as shown in the example. Replace the value of `worker-2.test.mop.fr.ibm.com` with the name of your worker node.

   ```
   [root@opnshrh8 spectrum]# oc debug node/worker-2.test.mop.fr.ibm.com -- chroot
   /host crio-status config | grep pids_limit
   Creating debug namespace/openshift-debug-node-64d6x ...
   Starting pod/worker-2testmopfribmcom-debug ...
   To use host binaries, run `chroot /host`
       pids_limit = 4096
   Removing debug pod ...
   Removing debug namespace/openshift-debug-node-64d6x ...
   ```

Increase the Vmalloc kernel parameter

Increase the size of the `vmalloc kernel` parameter to allow IBM Spectrum Scale to run properly with Red Hat CoreOS. To achieve this change, we use the `spectrumscale/preinstall/99-openshift-machineconfig-worker-kargs.yaml` file contained in the archive file.

To increase the size of the `vmalloc kernel` parameter, follow these steps:

1. Ensure that you are in the new spectrum folder.

   ```
   [root@opnshrh8~]# cd /root/spectrum
   ```

2. Apply this configuration to Red Hat OCP by using the following command.

```
[root@opnshrh8 spectrum]# oc apply -f
spectrumscale/preinstall/99-openshift-machineconfig-worker-kargs.yaml

machineconfig.machineconfiguration.openshift.io/99-openshift-machineconfig-work
er-kargs created
```

The command starts an OCP rolling-update to apply the desired changes. Wait until the rolling update is complete.

3. Check the status of the update by using the following command.

```
[root@opnshrh8 spectrum]# oc get MachineConfigPool
```

When the update is complete, the following parameters are set for the worker nodes as follows (Figure 1-2):

– The `UPDATING` parameter is equal to `False`.

– The `UPDATED` parameter is equal to `True`.

4. Verify that the changes have been applied by using the following commands. The value of `vmalloc` must be equal to `4096G`.

```
[root@opnshrh8 spectrum]# oc describe machineconfig | grep vmalloc
[root@opnshrh8 spectrum]# oc debug node/<name_of_your_worker node> -- cat
/proc/cmdline
```

Modify the Machine Config Operator

The next steps provide the kernel module support necessary to properly deploy IBM Spectrum Scale CNSA.

To modify the MCO, follow these steps:

1. Ensure that you are in the spectrum folder.

```
[root@opnshrh8~]# cd /root/spectrum
```

2. Create the following `yaml` file to apply the changes to the Machine Config Operator and add the following content.

```
[root@opnshrh8 spectrum]# vi spectrumscale/machineconfigoperator.yaml

apiVersion: machineconfiguration.openshift.io/v1
kind: MachineConfig
metadata:
  labels:
   machineconfiguration.openshift.io/role: "worker"
  name: 02-worker-kernel-devel
spec:
  config:\ version: 3.1.0
  extensions:
     - kernel-devel
```

3. Apply the changes.

```
[root@opnshrh8 spectrumscale]# oc create -f
spectrumscale/machineconfigoperator.yaml

machineconfig.machineconfiguration.openshift.io/02-worker-kernel-devel created
```

This command starts an OCP rolling-update to apply the desired changes. Wait until the rolling update is complete.

4. Check the status of the update by using the following command.

   [root@opnshrh8 spectrum]# oc get MachineConfigPool

   When the update is complete, the following parameters are set for the worker nodes, as follows (Figure 1-2):

   – The `UPDATING` parameter is equal to `False`.

   – The `UPDATED` parameter is equal to `True`.

5. Verify that the changes have been taken into account by using the following command.

   The `kernel-devel` package must be displayed in the output of this command, as shown below in italics.

```
[root@opnshrh8 spectrumscale]# oc get nodes -lnode-role.kubernetes.io/worker=
-ojsonpath="{range .items[*]}{.metadata.name}{'\n'}" |xargs -I{} oc debug
node/{} -T -- chroot /host sh -c "rpm -q kernel-devel"

Creating debug namespace/openshift-debug-node-s5cjn ...
Starting pod/worker-0testmopfribmcom-debug ...
To use host binaries, run `chroot /host`
kernel-devel-4.18.0-193.29.1.el8_2.s390x
Removing debug pod ...
Removing debug namespace/openshift-debug-node-s5cjn ...
Creating debug namespace/openshift-debug-node-w5w7z ...
Starting pod/worker-1testmopfribmcom-debug ...
To use host binaries, run `chroot /host`
kernel-devel-4.18.0-193.29.1.el8_2.s390x

Removing debug pod ...
Removing debug namespace/openshift-debug-node-w5w7z ...
Creating debug namespace/openshift-debug-node-kz6mk ...
Starting pod/worker-2testmopfribmcom-debug ...
To use host binaries, run `chroot /host`
kernel-devel-4.18.0-193.29.1.el8_2.s390x
Removing debug pod ...
Removing debug namespace/openshift-debug-node-kz6mk ...
```

1.3.3 Pushing Spectrum Scale container images to the OCP registry

In this section, we push the container images necessary to deploy IBM Spectrum Scale to the Red Hat OCP registry. These images are contained in the downloaded archive file.

In this document, we use the internal OpenShift registry to upload these images. Therefore, the steps in this section correspond to the is OpenShift registry. If you plan to use your private registry, skip this section and use your usual commands to upload the images to your private registry. As described in this section, we tagged the images with the tag: "5.1.0.1".

To push the container images to the OCP registry, follow these steps:

1. Login to the Red Hat OCP cluster with a user ID that has access to the internal registry and ensure that Red Hat OCP registry[2] is enabled by using the following command.

```
[root@opnshrh8 spectrumscale]# oc get deployment image-registry -n
openshift-image-registry
NAME              READY   UP-TO-DATE   AVAILABLE   AGE
```

[2] In this documentation, we assume that the route registry has already been exposed: $ oc patch configs.imageregistry.operator.openshift.io/cluster --patch '{"spec":{"defaultRoute":true}}' --type=merge

```
image-registry   1/1     1            1           220d
```

2. Obtain the route of the OCP internal registry by using the following command.

```
[root@opnshrh8 spectrum]# oc get route default-route -n
openshift-image-registry --template='{{ .spec.host }}'

default-route-openshift-image-registry.apps.test.mop.fr.ibm.com
```

We will use this route to connect to the internal registry.

3. Log in to the Red Hat OCP internal container registry by using the following command.

> **Note:** The `default-route-openshift-image-registry.apps.test.mop.fr.ibm.com` value must be replaced by the name of your own registry.

```
[root@opnshrh8 spectrum]# podman login -u $(oc whoami) -p $(oc  whoami -t)
--tls-verify=false
default-route-openshift-image-registry.apps.test.mop.fr.ibm.com

Login Succeeded!
```

4. Before you load the images, navigate to the project (`ibm-spectrum-scale-ns`) created in step 7 on page 4.

```
[root@opnshrh8 spectrumscale]# oc project ibm-spectrum-scale-ns
```

5. Ensure you are in the new spectrum folder.

```
[root@opnshrh8~]# cd /root/spectrum
```

6. The IBM Spectrum Scale container images are present in the `.tar` file extracted in step 6 on page 4. Use the following commands to import these images. In this document the tag is 5.1.0.1.

   a. Create a script with the content shown below.

   ```
   [root@opnshrh8 spectrumscale]# vi importImages.sh

   TAG="5.1.0.1"
     for file in `ls spectrumscale/*.tar`; do
        tarname=${file##*/}
        tagname=`echo $tarname | sed 's/.tar//g' | sed "s/-$TAG/:$TAG/g"`
        echo "-- Loading $file as $tagname"
        # If using docker, the load and tagging cannot be done in a single step
        podman load -i $file localhost/$tagname
    done
   ```

   b. Run the script.

   ```
   [root@opnshrh8 spectrumscale]# ./importImages.sh
   ```

7. Tag and push the images to the internal registry.

   a. Create a script with the following content.

   ```
   [root@opnshrh8 spectrumscale]# vi pushImages.sh
   TAG="5.1.0.1"
   HOST=$(oc get route default-route -n openshift-image-registry --
   template='{{ .spec.host }}'
   NAMESPACE="ibm-spectrum-scale-ns"
   for file in `ls spectrumscale/*.tar`; do
       tarname=${file##*/}
   ```

```
    tagname=`echo $tarname | sed 's/.tar//g' | sed "s/-$TAG/:$TAG/g"`
    podman tag localhost/$tagname $HOST/$NAMESPACE/$tagname
    podman push $HOST/$NAMESPACE/$tagname --tls-verify=false
done
```

b. Run the script.

```
[root@opnshrh8 spectrumscale]# ./pushImages.sh
```

8. After you run the previous scripts, ensure that images were pushed successfully and stored in the right repository as shown below.

```
[root@opnshrh8 spectrum]# podman images | grep ibm-spectrum-scale-ns
default-route-openshift-image-registry.apps.test.mop.fr.ibm.com/ibm-spectrum-sc
ale-ns/scale-core            5.1.0.1   a3222ebfa7ec   2 months ago    840 MB
default-route-openshift-image-registry.apps.test.mop.fr.ibm.com/ibm-spectrum-sc
ale-ns/scale-pmcollector     5.1.0.1   6e977f3eee35   2 months ago    406 MB
default-route-openshift-image-registry.apps.test.mop.fr.ibm.com/ibm-spectrum-sc
ale-ns/scale-monitor         5.1.0.1   cf3a8a07e8af   2 months ago    451 MB
default-route-openshift-image-registry.apps.test.mop.fr.ibm.com/ibm-spectrum-sc
ale-ns/scale-gui             5.1.0.1   c4575c081fc5   2 months ago    811 MB
default-route-openshift-image-registry.apps.test.mop.fr.ibm.com/ibm-spectrum-sc
ale-ns/scale-core-operator   5.1.0.1   33f91b3682be   2 months ago    150 MB
```

9. Ensure that the images are contained in the `ImageStream` by using the following command.

```
[root@opnshrh8 spectrum]# for image in `oc get is -o
custom-columns=NAME:.metadata.name --no-headers`; do
> echo "---"
> oc get is $image -o yaml | egrep "name:|dockerImageRepository"
> done
---
  name: scale-core
  dockerImageRepository:
image-registry.openshift-image-registry.svc:5000/ibm-spectrum-scale-ns/scale-core
---
  name: scale-core-operator
  dockerImageRepository:
image-registry.openshift-image-registry.svc:5000/ibm-spectrum-scale-ns/scale-core-operat
or
---
  name: scale-gui
  dockerImageRepository:
image-registry.openshift-image-registry.svc:5000/ibm-spectrum-scale-ns/scale-gui
---
  name: scale-monitor
  dockerImageRepository:
image-registry.openshift-image-registry.svc:5000/ibm-spectrum-scale-ns/scale-monitor
---
  name: scale-pmcollector
  dockerImageRepository:
image-registry.openshift-image-registry.svc:5000/ibm-spectrum-scale-ns/scale-pmcollector
```

2

IBM Spectrum Scale CNSA installation

This chapter describes how to deploy the IBM Spectrum Scale Operator by using the container images pushed in the OpenShift Container Platform (OCP) registry and the `yaml` files that were downloaded in the archive.

2.1 OCP preparation

To prepare OCP for the CNSA installation, follow these steps:

1. Ensure that you are in `ibm-spectrum-scale-ns` namespace.

   `[root@opnshrh8 spectrumscale]# oc project ibm-spectrum-scale-ns`

2. Navigate to the folder from which the archive was extracted.

   `[root@opnshrh8 spectrum]# cd /root/spectrum/`

3. Edit the following template `yaml` file to add the name of the `namespace`. In Figure 2-1. the value of `namespace` is `ibm-spectrum-scale-ns`.

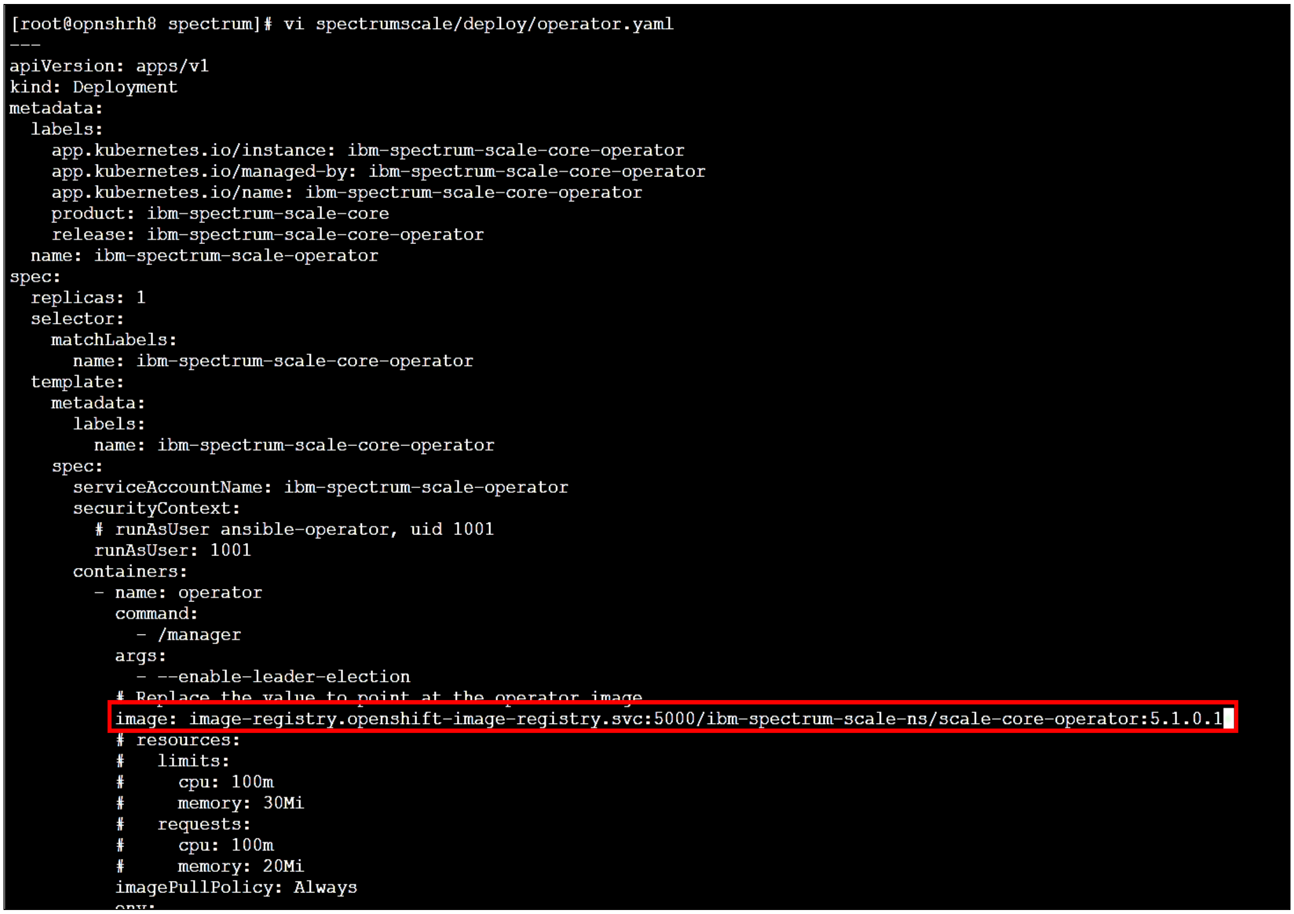
   `[root@opnshrh8 spectrumscale]# vi spectrumscale/deploy/cluster_role_binding.yaml`

```
[root@opnshrh8 spectrum]# vi spectrumscale/deploy/cluster_role_binding.yaml
---
kind: ClusterRoleBinding
apiVersion: rbac.authorization.k8s.io/v1
metadata:
  labels:
    app.kubernetes.io/instance: ibm-spectrum-scale-core-operator
    app.kubernetes.io/managed-by: ibm-spectrum-scale-core-operator
    app.kubernetes.io/name: ibm-spectrum-scale-core-operator
    product: ibm-spectrum-scale-core
    release: ibm-spectrum-scale-core-operator
  name: ibm-spectrum-scale-operator
roleRef:
  kind: ClusterRole
  name: ibm-spectrum-scale-operator
  apiGroup: rbac.authorization.k8s.io
subjects:
  - kind: ServiceAccount
    name: ibm-spectrum-scale-operator
    namespace: ibm-spectrum-scale-ns
~
```

Figure 2-1 yaml file template with namespace

4. Edit the `yaml` file to add the location of the container image that is needed to deploy the IBM Spectrum Scale (Figure 2-2). This image was pushed in the OCP internal registry in 1.3.3, “Pushing Spectrum Scale container images to the OCP registry” on page 7.

   `[root@opnshrh8 spectrumscale]# vi spectrumscale/deploy/operator.yaml`



```
[root@opnshrh8 spectrum]# vi spectrumscale/deploy/operator.yaml
---
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    app.kubernetes.io/instance: ibm-spectrum-scale-core-operator
    app.kubernetes.io/managed-by: ibm-spectrum-scale-core-operator
    app.kubernetes.io/name: ibm-spectrum-scale-core-operator
    product: ibm-spectrum-scale-core
    release: ibm-spectrum-scale-core-operator
  name: ibm-spectrum-scale-operator
spec:
  replicas: 1
  selector:
    matchLabels:
      name: ibm-spectrum-scale-core-operator
  template:
    metadata:
      labels:
        name: ibm-spectrum-scale-core-operator
    spec:
      serviceAccountName: ibm-spectrum-scale-operator
      securityContext:
        # runAsUser ansible-operator, uid 1001
        runAsUser: 1001
      containers:
        - name: operator
          command:
            - /manager
          args:
            - --enable-leader-election
          # Replace the value to point at the operator image
          image: image-registry.openshift-image-registry.svc:5000/ibm-spectrum-scale-ns/scale-core-operator:5.1.0.1
          # resources:
          #   limits:
          #     cpu: 100m
          #     memory: 30Mi
          #   requests:
          #     cpu: 100m
          #     memory: 20Mi
          imagePullPolicy: Always
```

Figure 2-2 Add location to the yaml file

To obtain the value that needs to be added to the line highlighted in Figure 2-2, you can use the output of the following command, shown in italics concatenated with the "5.1.0.1" tag.

`[root@opnshrh8 spectrum]# for image in `oc get is -o custom-columns=NAME:.metadata.name --no-headers`; do echo "---"; oc get is $image -o yaml | egrep "name:|dockerImageRepository"; done | grep scale-core-operator`

```
  name: scale-core-operator
```

`dockerImageRepository:` *`image-registry.openshift-image-registry.svc:5000/ibm-spectrum-scale-ns/scale-core-operator`*

5. Apply the following preparation `.yaml` files to your Red Hat OCP cluster. These files are already present in the downloaded archive file.

   `[root@opnshrh8 spectrum]# oc create -f spectrumscale/deploy/service_account.yaml -n ibm-spectrum-scale-ns`

```
 oc create -f spectrumscale/deploy/service_account_core.yaml -n ibm-spectrum-scale-ns
 oc create -f spectrumscale/deploy/role.yaml -n ibm-spectrum-scale-ns
 oc create -f spectrumscale/deploy/role_binding.yaml -n ibm-spectrum-scale-ns
 oc create -f spectrumscale/deploy/role_scale_core.yaml -n ibm-spectrum-scale-ns
```

```
 oc create -f spectrumscale/deploy/role_binding_scale_core.yaml -n
ibm-spectrum-scale-ns
 oc create -f spectrumscale/deploy/scc.yaml -n ibm-spectrum-scale-ns
 oc create -f spectrumscale/deploy/cluster_role.yaml -n ibm-spectrum-scale-ns
 oc create -f spectrumscale/deploy/cluster_role_binding.yaml -n
ibm-spectrum-scale-ns
 oc create -f spectrumscale/deploy/crds/ibm_v1_scalecluster_crd.yaml -n
ibm-spectrum-scale-ns
```

2.2 OCP parameter configuration

To deploy the IBM Spectrum Scale CNSA operator, several parameters must be set. The values of these parameters are defined in the `spectrumscale/deploy/scale-profile.yaml` file, which is present in the archive. In this documentation, we use the default values defined in this file. Thus, modification of the file has not been made. If you want to change the values, you can edit the file to add your own values.

1. Create the `ConfigMap` using the `spectrumscale/deploy/scale-profile.yaml` file.

   ```
   [root@opnshrh8 spectrum]#  oc create -f spectrumscale/deploy/scale-profile.yaml
   -n ibm-spectrum-scale-ns

   configmap/ibm-spectrum-scale-core-profile created
   ```

2. Verify that the `ConfigMap` object was created successfully.

   ```
   [root@opnshrh8 spectrum]#  oc get configmap -n ibm-spectrum-scale-ns | grep
   profile

   ibm-spectrum-scale-core-profile   1      9s
   ```

2.3 User and secret creation

In this section, we first create the user in the remote Spectrum Scale storage server, which will be used later by the Spectrum Scale Operator to mount the storage cluster filesystem. Then, we create a secret on Red Hat OCP to hold the username and password for the IBM Spectrum Scale Storage cluster GUI user and password.

2.3.1 Remote Spectrum Scale

To create the user in the remote storage server, follow these steps:

1. Connect to the remote storage server where the Spectrum Scale GUI node is installed.
2. Create the “`ContainerOperator`” GUI user group by using the following command.

   ```
   [root@spscale1 ~]# /usr/lpp/mmfs/gui/cli/mkusergrp ContainerOperator --role
   containeroperator
   ```

3. Create the IBM Spectrum Scale GUI user in the “`ContainerOperator`” GUI user group.

   ```
   [root@spscale1 ~]# /usr/lpp/mmfs/gui/cli/mkuser cnss_storage_gui_user -p
   cnss_storage_gui_password -g ContainerOperator

   EFSSG0019I The user cnss_storage_gui_user has been successfully created.
   ```

2.3.2 Red Hat OCP

To create the secret, go back to OCP cluster and create the "cnsa-remote-mount-storage-cluster-1" secret by using the following command.

```
[root@opnshrh8 spectrum]# oc create secret generic
cnsa-remote-mount-storage-cluster-1
--from-literal=username='cnss_storage_gui_user'
--from-literal=password='cnss_storage_gui_password' -n ibm-spectrum-scale-ns

secret/cnsa-remote-mount-storage-cluster-1 created
```

2.4 CNSA deployment

CNSA deployment consists of creating a custom resource and using it to deploy the IBM Spectrum Scale Operator.

2.4.1 Custom Resource customization

To deploy the IBM Spectrum Scale operator, we use a Custom Resource (CR). A CR template is already included in the downloaded archive. This template is: `spectrumscale/deploy/crds/ibm_v1_scalecluster_cr.yaml`. The entries in this `yaml` file must be carefully filled to allow the operator to be deployed successfully.

To customize the CR, follow these steps:

1. Go to the folder where the archive was extracted.

   ```
   [root@opnshrh8 spectrum]# cd /root/spectrum/
   ```

2. Edit the `spectrumscale/deploy/crds/ibm_v1_scalecluster_cr.yaml` file to set the values of the parameters as described in the remaining steps. These values must match your existing environment.

   ```
   [root@opnshrh8 ~]# vim spectrumscale/deploy/crds/ibm_v1_scalecluster_cr.yaml
   ```

3. In the **images** section in the file, replace "`REPLACE_CONTAINER_REGISTRY`" with the route of your container registry (as seen in 1.3.3, "Pushing Spectrum Scale container images to the OCP registry" on page 7 to load the IBM Spectrum Scale images), followed by the name of the image and its tag.

   For example, in our environment the route is `image-registry.openshift-image-registry.svc:5000/ibm-spectrum-scale-ns`, so the **images** section in our CR file looks like this:

   ```
   images:
     core:
   image-registry.openshift-image-registry.svc:5000/ibm-spectrum-scale-ns/scale-co
   re:5.1.0.1
     coreInit:
   image-registry.openshift-image-registry.svc:5000/ibm-spectrum-scale-ns/scale-co
   re:5.1.0.1
     gui:
   image-registry.openshift-image-registry.svc:5000/ibm-spectrum-scale-ns/scale-gu
   i:5.1.0.1
   ```

```
  pmcollector:
image-registry.openshift-image-registry.svc:5000/ibm-spectrum-scale-ns/scale-pm
collector:5.1.0.1
  sysmon:
image-registry.openshift-image-registry.svc:5000/ibm-spectrum-scale-ns/scale-mo
nitor:5.1.0.1
```

4. Also in the **images** section, point the "`logs`" parameter to the 8.2 tag of `ubi-minimal` (Figure 2-3).

Figure 2-3 Update the logs parameter in the images section

5. In the **filesystems** section in the file, enter the information related to the filesystem (Figure 2-4).

```
filesystems:
   - name: "gpfs1" => Choose a name to the filesystem that will be created for
  CNSA deployment
     remoteMount:
       storageCluster: "storageCluster1" => Choose a name to reference the
        remote storage cluster
       storageFs: "gpfsFS1" => The name of the filesystem in the remote
        Spectrum Scale cluster
     mountPoint: "/mnt/gpfs1" => The mount point where the remote filesystem
     will be mounted in the worker nodes. It must be under /mnt
```

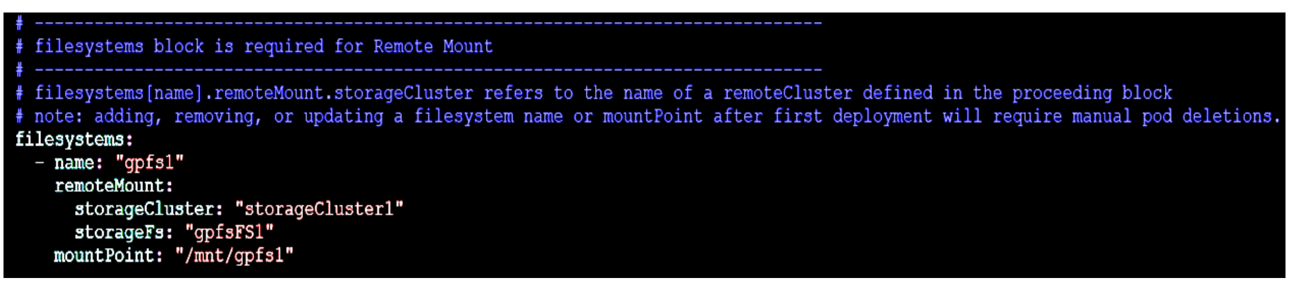


Figure 2-4 filesystems updates

6. In the **remoteClusters** section in the file, enter the information related to the remote Spectrum Scale storage server (Figure 2-5).

```
remoteClusters:
- name: storageCluster1 => It should be the same name defined the in the
Filesystems section, parameter "storageCluster"
  gui:
    host: "spscale1.mop.fr.ibm.com" => hostname of the GUI of the remote
      Spectrum Scale storage server
```

secretName: "cnsa-remote-mount-storage-cluster-1" => The name of the secret created in section User and secret creation
insecureSkipVerify: true => True is the only option supported at this moment

```
# ------------------------------------------------------------------------------
# The remoteClusters field is required for remote mount
# ------------------------------------------------------------------------------
# A remoteCluster definition provides the name, hostname, its GUI secret, and contact node.
# The remoteCluster name is referenced in the filesystems[name].remoteMount.storageCluster
# used for Remote Mount
remoteClusters:
  - name: "storageCluster1"
    gui:
      host: "spscale1.mop.fr.ibm.com"
      secretName: "cnsa-remote-mount-storage-cluster-1"
      insecureSkipVerify: true
    # contactNodes:
    #   - storageclusternode1
    #   - storageclusternode2
```

Figure 2-5 remoteClusters updates

2.4.2 IBM Spectrum Scale CNSA cluster deployment

To deploy the CNSA cluster, follow these steps:

1. Ensure that you are in `ibm-spectrum-scale-ns` namespace.

   [root@opnshrh8 spectrumscale]# oc project ibm-spectrum-scale-ns

2. Navigate to the folder in which the archive was extracted.

   [root@opnshrh8 spectrum]# cd /root/spectrum/

3. Deploy the Operator.

   [root@opnshrh8 spectrumscale]# oc create -f spectrumscale/deploy/operator.yaml -n ibm-spectrum-scale-ns

   An operator pod is created. The status of this pod must "`Running`"

   [root@opnshrh8 spectrumscale]# oc get pods -n ibm-spectrum-scale-ns

```
NAME                                      READY   STATUS    RESTARTS   AGE
ibm-spectrum-scale-operator-d7ffcbc7-7znj7   1/1     Running   0          15s
```

4. Apply the CR created in 2.4, “CNSA deployment” on page 15 to the cluster.

   [root@opnshrh8 spectrumscale]# oc create -f spectrumscale/deploy/crds/ibm_v1_scalecluster_cr.yaml -n ibm-spectrum-scale-ns

5. Verify that `scalecluster_cr` CR was created successfully. If so, the following pods will be up and running.

   [root@opnshrh8 ~]# oc get pods -n ibm-spectrum-scale-ns

```
NAME                                        READY   STATUS    RESTARTS   AGE
ibm-spectrum-scale-core-bpbk5               1/1     Running   0          27m
ibm-spectrum-scale-core-w9h8b               1/1     Running   0          26m
ibm-spectrum-scale-core-wjg6n               1/1     Running   0          27m
ibm-spectrum-scale-gui-0                    9/9     Running   0          26m
ibm-spectrum-scale-operator-7f99c44f5c-chlcr   1/1     Running   0          61m
ibm-spectrum-scale-pmcollector-0            2/2     Running   0          34m
ibm-spectrum-scale-pmcollector-1            2/2     Running   0
```

Work around: If a pod remains in incorrect state, try to delete the pod. This action can allow to the pod to be created successfully the second time.

6. Use the following commands to verify that the IBM Spectrum Scale CNSA cluster is properly created. The outputs should be similar to those shown below. Replace the name of the "`ibm-spectrum-scale-core-bpbk5`" pod with the name of one of the IBM Spectrum Scale cores that is running in your environment.

   – **[root@opnshrh8 ~]# oc exec ibm-spectrum-scale-core-bpbk5 -n ibm-spectrum-scale-ns -- mmlscluster**

```
GPFS cluster information
========================
  GPFS cluster name:         ibm-spectrum-scale.ibm-spectrum-scale-ns.test.mop.fr.ibm.com
  GPFS cluster id:           18306118698498958180
  GPFS UID domain:           ibm-spectrum-scale.ibm-spectrum-scale-ns.test.mop.fr.ibm.com
  Remote shell command:      /usr/bin/ssh
  Remote file copy command:  /usr/bin/scp
  Repository type:           CCR
Node Daemon node name             IP address   Admin node name              Designation
--------------------------------------------------------------------------------------------------
   1   worker-0.test.mop.fr.ibm.com  10.3.57.220  worker-0.test.mop.fr.ibm.com  quorum-manager-perfmon
   2   worker-1.test.mop.fr.ibm.com  10.3.57.222  worker-1.test.mop.fr.ibm.com  quorum-manager-perfmon
   3   worker-2.test.mop.fr.ibm.com  10.3.57.223  worker-2.test.mop.fr.ibm.com  quorum-manager-perfmon
```

   – **[root@opnshrh8 ~]# oc exec ibm-spectrum-scale-core-bpbk5 -n ibm-spectrum-scale-ns -- mmgetstate -a**

```
 Node number  Node name        GPFS state
-------------------------------------------
       1      worker-0         active
       2      worker-1         active
       3      worker-2         active
```

   – **[root@opnshrh8 ~]# oc exec ibm-spectrum-scale-core-2vmxt -- mmremotecluster show all**

```
Cluster name:    ClusterSpectrum.mop.fr.ibm.com
Contact nodes:   spscale1.mop.fr.ibm.com,spscale2.mop.fr.ibm.com
SHA digest:
0715f3f0aafe7ad075eb16f32599be9781671adfb43f8cc7627994ec9cd8aa21
File systems:    gpfs1 (gpfsFS1)
```

Important: If the File Systems parameter, highlighted above in ***italics*** shows an empty value, or if the following command shows the same output as below, check the logs of the Spectrum Scale operator.

```
[root@opnshrh8 spectrum]# oc exec ibm-spectrum-scale-core-4pftj -n
ibm-spectrum-scale-ns -- mmremotecluster show all

mmremotecluster: There are no remote cluster definitions.
```

If the logs show an error similar to the error displayed in Logs section below, please perform the steps described in "Workaround for cluster deployment failure" on page 19.

```
# [root@opnshrh8 ~]# oc logs ibm-spectrum-scale-operator-7f99c44f5c-kx974
-n ibm-spectrum-scale-ns
2021-02-16T10:05:08.726Z        ERROR   controllers.RemoteMount rest
error: Get
"https://ibm-spectrum-scale-gui.ibm-spectrum-scale-ns/scalemgmt/v2/remote
mount/remotefilesystems/gpfs1": context deadline exceeded (Client.Timeout
exceeded while awaiting headers)  {"ScaleCluster":
"ibm-spectrum-scale-ns/ibm-spectrum-scale", "error": "Get
\"https://ibm-spectrum-scale-gui.ibm-spectrum-scale-ns/scalemgmt/v2/remot
emount/remotefilesystems/gpfs1\": context deadline exceeded
(Client.Timeout exceeded while awaiting headers)"}
```

Workaround for cluster deployment failure

The IBM Spectrum Scale GUI REST credentials are stored in OCP using secrets. In some cases, they might become out of sync with what is known to IBM Spectrum Scale GUI.

To eliminate this issue, follow these steps:

1. Obtain the credentials from the OCP secret that was created by the operator to access the in-cluster IBM Spectrum Scale GUI REST API. In this example, the password is `t9hnXXk6CZ7IFiHzbOmO`.

   ```
    [root@opnshrh8 ~]# oc get secret ibm-spectrum-scale-gui-containeroperator -n
   ibm-spectrum-scale-ns -ojsonpath='{.data.password}' | base64 -d -

   t9hnXXk6CZ7IFiHzbOmO
   ```

2. Connect to the in-cluster IBM Spectrum Scale GUI pod.

   ```
    [root@opnshrh8 ~]# oc rsh -n ibm-spectrum-scale-ns ibm-spectrum-scale-gui-0
   ```

3. Update the password to the one expected by the operator, by typing the following command, concatenated with the password obtained in step 1 in the shell of the pod.

   ```
   /usr/lpp/mmfs/gui/cli/chuser ContainerOperator -p t9hnXXk6CZ7IFiHzbOmO
   ```

4. Wait a little bit then check if the filesystem has been configured correctly this time, by using the following command.

```
[root@opnshrh8 ~]# oc exec ibm-spectrum-scale-core-2vmxt  -- mmremotefs show

Local Name  Remote Name  Cluster name Mount Point Mount Options Automount Drive  Priority
gpfs1       gpfsFS1      ClusterSpectrum.mop.fr.ibm.com /mnt/gpfs1 rw yes -       0
```

5. Verify that the storage cluster filesystem has been remotely mounted.

```
[root@opnshrh8 ~]# oc exec ibm-spectrum-scale-core-2vmxt  -- mmlsmount gpfs1 -L

File system gpfs1 (ClusterSpectrum.mop.fr.ibm.com:gpfsFS1) is mounted on 5
nodes:
  10.3.57.15      spscale1                 ClusterSpectrum.mop.fr.ibm.com
  10.3.57.7       spscale2                 ClusterSpectrum.mop.fr.ibm.com
  10.3.57.222     worker-1.test
ibm-spectrum-scale.ibm-spectrum-scale-ns.test.mop.fr.ibm.com
  10.3.57.220     worker-0.test
ibm-spectrum-scale.ibm-spectrum-scale-ns.test.mop.fr.ibm.com
  10.3.57.223     worker-2.test
ibm-spectrum-scale.ibm-spectrum-scale-ns.test.mop.fr.ibm.com
```

3

Deploying the IBM Spectrum Scale CSI driver

Container Storage Interface (CSI) is a standard that allows storage vendors to develop a plugin to expose their storage products as persistent storage to containerized applications. Thus, the IBM Spectrum Scale CSI driver allows IBM Spectrum Scale to be used as a persistent storage for Red Hat OpenShift Container Platform (OCP) applications. Through this CSI driver, we can provision persistent volumes (PVs) from IBM Spectrum Scale and serve them to the applications.

This chapter describes how to deploy the IBM Spectrum Scale CSI driver with the IBM Spectrum Scale CNSA that is deployed in Chapter 2, "IBM Spectrum Scale CNSA installation" on page 11.

This deployment can be done by using the OCP Web Console or the command line interface (CLI). In this chapter, the Web Console is used to deploy the CSI driver.

3.1 Preparation tasks for the remote IBM Spectrum Scale Storage server

To prepare the remote IBM Spectrum Scale Storage server, follow these steps:

1. Connect to the remote storage server on which the Spectrum Scale GUI node is installed and create an IBM Spectrum Scale user group "CsiAdmin".

   ```
   [root@spscale1 ~]# /usr/lpp/mmfs/gui/cli/mkusergrp CsiAdmin --role csiadmin
   ```

2. Create an IBM Spectrum Scale user in the "CsiAdmin" group. This user will be used later during the configuration of the IBM Spectrum Scale CSI driver.

   ```
   [root@spscale1 ~]# /usr/lpp/mmfs/gui/cli/mkuser csi-storage-gui-user -p
   csi-storage-gui-password -g CsiAdmin

   EFSSG0019I The user csi-storage-gui-user has been successfully created.
   EFSSG1000I The command completed successfully.
   ```

3. Verify that the value of Perfileset quota is set to No. This quota pertains to the file system that will be used by IBM Spectrum Scale CSI driver. In our environment, the name of this file system is gpfsFS1.

   ```
   [root@spscale1 ~]# mmlsfs gpfsFS1 --perfileset-quota

   flag             value     description
   ----             -----    -----------------------------
   --perfileset-quota no      Per-fileset quota enforcement
   ```

4. Verify that the quota is enabled in the file system that will be used by IBM Spectrum Scale CSI driver. In our environment, the name of this file system is gpfsFS1.

   ```
   [root@spscale1 ~]# mmchfs gpfsFS1 -Q yes

   mmchfs: Propagating the cluster configuration data to all
     affected nodes.  This is an asynchronous process.
   ```

5. To verify that the quota is enabled, you can use the following command.

   ```
   [root@spscale1 ~]# mmlsfs gpfsFS1 -Q

   flag             value                 description
   ----             -----                 --------------
    -Q              user;group;fileset    Quotas accounting enabled
                    user;group;fileset    Quotas enforced
                    none                  Default quotas enabled
   ```

6. Enable the quota for the root user.

   ```
   [root@spscale1 ~]# mmchconfig enforceFilesetQuotaOnRoot=yes -i

   mmchconfig: Command successfully completed
   mmchconfig: Propagating the cluster configuration data to all
     affected nodes.  This is an asynchronous process.
   ```

7. Set the controlSetxattrImmutableSELinux parameter to yes.

   ```
   [root@spscale1 ~]# mmchconfig controlSetxattrImmutableSELinux=yes -i

   mmchconfig: Command successfully completed
   mmchconfig: Propagating the cluster configuration data to all
     affected nodes.  This is an asynchronous process.
   ```

8. Enable the `filesetdf` parameter for the file system.

```
[root@spscale1 ~]# mmchfs gpfsFS1 --filesetdf
```

3.2 Preparation tasks for the IBM Spectrum Scale CNSA cluster

Connect to the OCP where the CNSA cluster is deployed and perform the following pre-installation tasks:

1. Label the OCP worker nodes that were chosen to deploy the IBM Spectrum Scale CNSA with the label "`scale=true`"

```
[root@opnshrh8 ~]# oc label nodes -l node-role.kubernetes.io/worker= scale=true

I0217 12:02:23.778883   13346 request.go:645] Throttling request took
1.080881635s, request:
GET:https://api.test.mop.fr.ibm.com:6443/apis/packages.operators.coreos.com/v1?
timeout=32s
node/worker-0.test.mop.fr.ibm.com labeled
node/worker-1.test.mop.fr.ibm.com labeled
node/worker-2.test.mop.fr.ibm.com labeled
```

2. Create a CNSA-GUI user for CSI:

```
[root@opnshrh8 ~]# oc project ibm-spectrum-scale-ns

[root@opnshrh8 ~]# oc exec -c liberty ibm-spectrum-scale-gui-0 --
/usr/lpp/mmfs/gui/cli/mkuser csi-cnsa-gui-user -p csi-cnsa-gui-password -g
CsiAdmin

EFSSG0019I The user csi-cnsa-gui-user has been successfully created.
EFSSG1000I The command completed successfully.
```

3.3 Installing the IBM Spectrum Scale CSI driver

The configuration of the IBM Spectrum Scale CSI driver to work with IBM Spectrum Scale CNSA includes the following steps:

1. "CSI operator deployment"
2. "IBM Spectrum Scale CSI driver deployment" on page 26

This section describes these steps and uses the Web Console method to deploy the CSI driver. Thus, the Operator Lifecycle Manager (OLM), provided by Red Hat OCP. is used.

3.3.1 CSI operator deployment

To deploy the CSI operator using the Red Hat OCP Web Console, follow these steps:

1. Create a namespace for the IBM Spectrum Scale CSI driver: `ibm-spectrum-scale-csi-driver`

   From the OpenShift console, navigate to **Projects** and click **Create Project.**

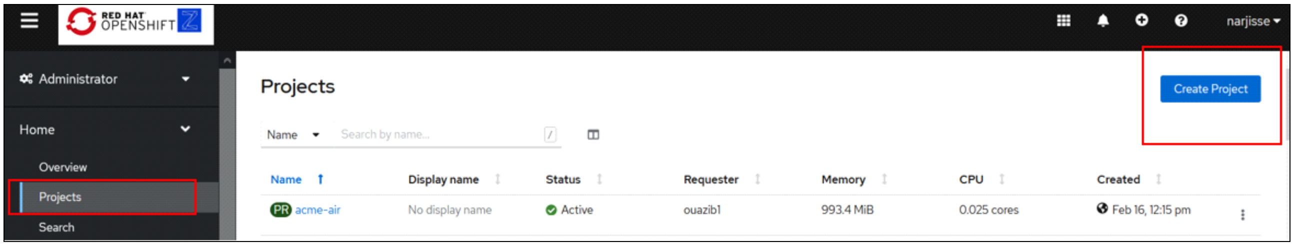


Figure 3-1 Create a project

In the **Create Project** window, enter the `namespace` in the Name field and click **Create**.

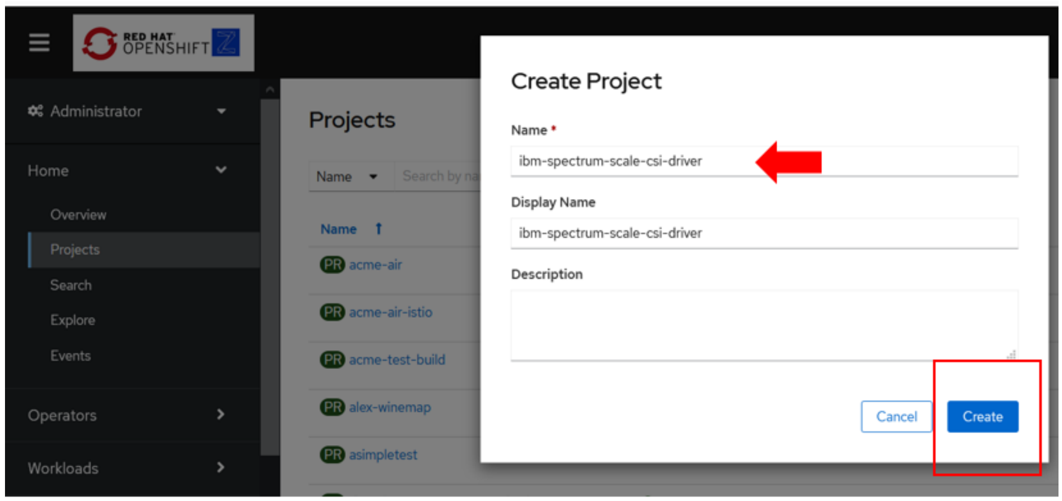


Figure 3-2 Project settings

2. To deploy the operator, navigate to the **OperatorHub** and search for and select the CSI plugin operator.

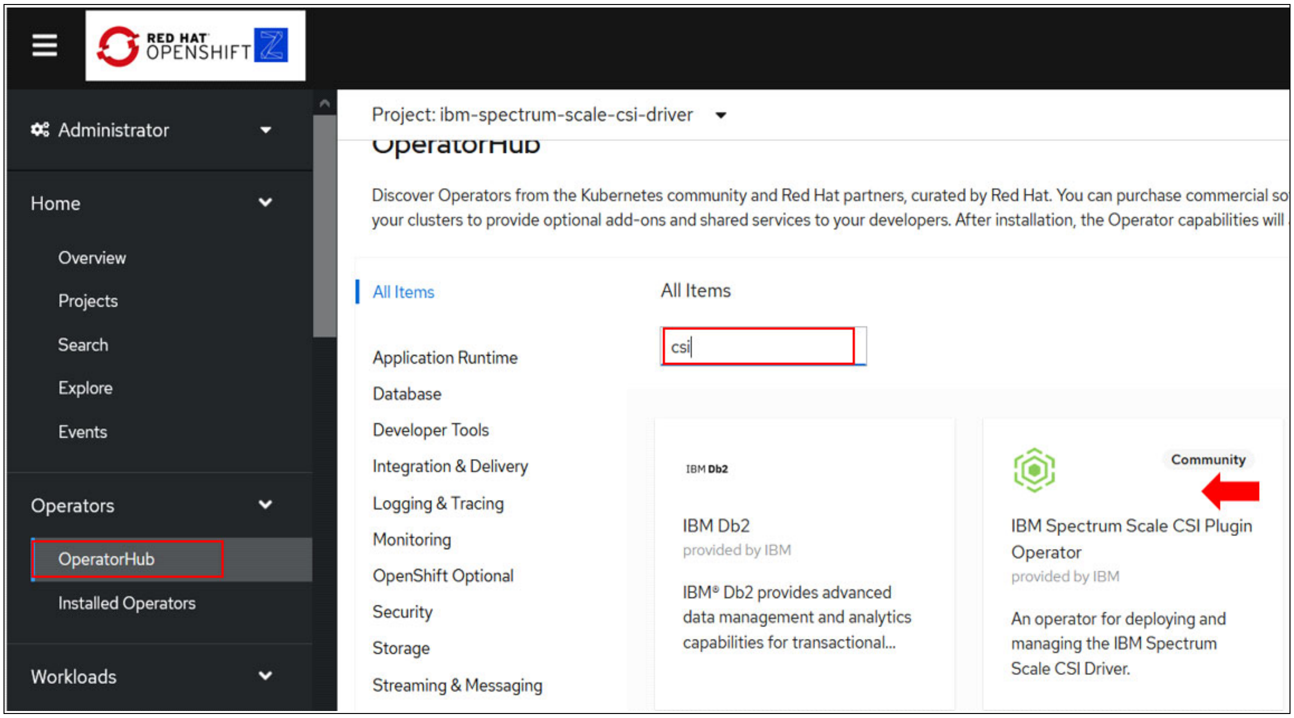


Figure 3-3   Select the CSI plugin operator

3. Complete the settings (Figure 3-4). The `namespace` is the one you created in step 1 on page 23. Click **Install.**

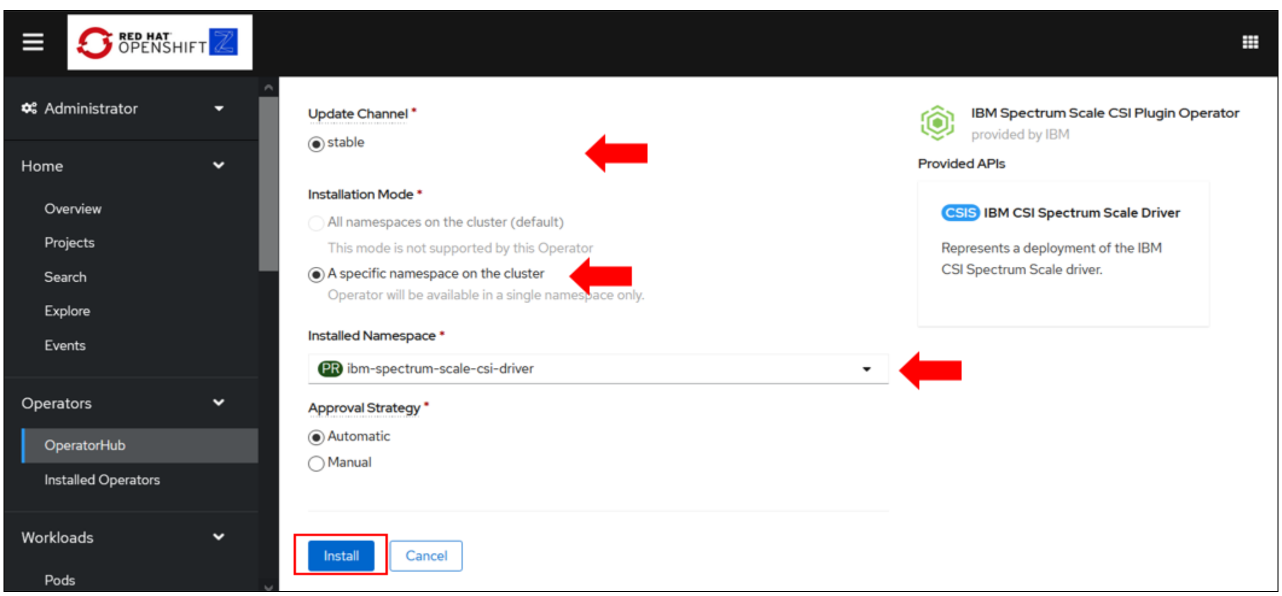


Figure 3-4   CSI plugin operator settings

The IBM Spectrum CSI operator is successfully installed (Figure 3-5).

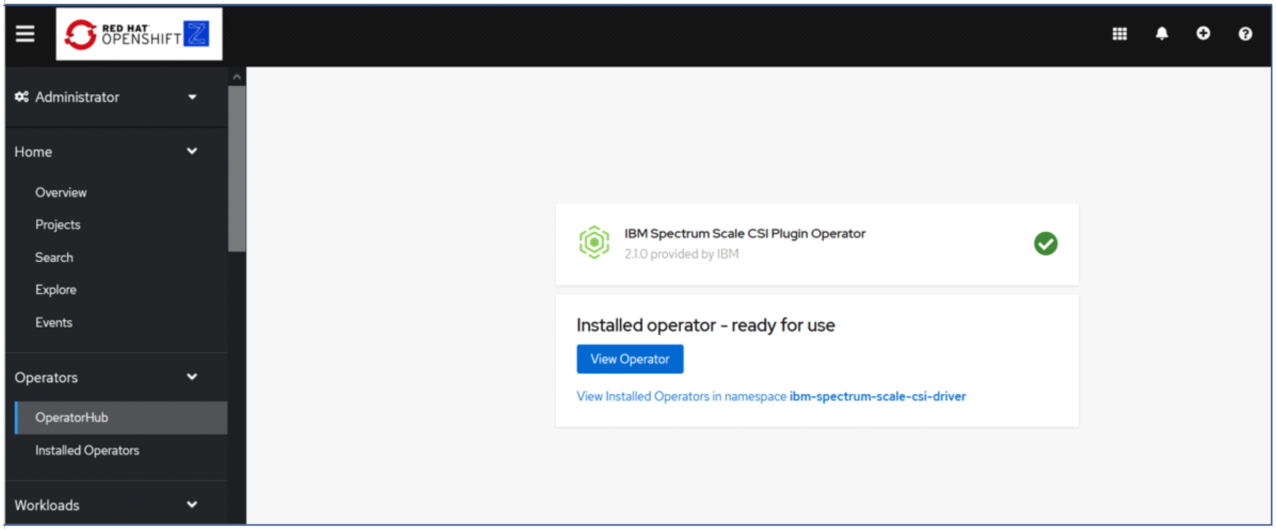


Figure 3-5 Successful installation

To verify that the Operator is deployed properly, the IBM Spectrum CSI Operator pod must be in a running state in the `ibm-spectrum-scale-csi-driver` namespace.

3.3.2 IBM Spectrum Scale CSI driver deployment

Before you deploy the IBM Spectrum Scale CSI driver, the following pre-installation steps must be completed:

- "Secret creation"
- "Configuration of the custom resource (CR)" on page 27
- "Deployment" on page 29

Secret creation

To create a secret for the CNSA GUI users, follow these steps:

1. Switch back to the Red Hat OCP CLI and navigate to the `ibm-spectrum-scale-csi-driver` namespace.

   ```
   [root@opnshrh8 crds]# oc project ibm-spectrum-scale-csi-driver
   ```

2. Create a secret for the CNSA GUI user that was created in 3.2, "Preparation tasks for the IBM Spectrum Scale CNSA cluster" on page 23. This secret is for the IBM Spectrum CNSA.

   ```
   [root@opnshrh8 crds]# oc create secret generic secret-cnsa-local
   --from-literal=username=csi-cnsa-gui-user
   --from-literal=password=csi-cnsa-gui-password -n ibm-spectrum-scale-csi-driver

   secret/secret-cnsa-local created

   [root@opnshrh8 spectrum]# oc label secret secret-cnsa-local
   product=ibm-spectrum-scale-csi

   secret/secret-cnsa-local labeled
   ```

3. Create a secret for the CNSA GUI user that was created for the storage server in 2.3, "User and secret creation" on page 14. This secret is for the remote Spectrum Scale.

```
[root@opnshrh8 crds]# oc create secret generic secret-storage-remote
--from-literal=username=csi-storage-gui-user
--from-literal=password=csi-storage-gui-password -n
ibm-spectrum-scale-csi-driver

secret/secret-storage-remote created

[root@opnshrh8 spectrum]# oc label secret secret-storage-remote
product=ibm-spectrum-scale-csi

secret/secret-storage-remote labeled
```

Configuration of the custom resource (CR)

This step is the configuration of the CR for the deployment of the IBM Spectrum Scale CSI driver from the CNSA cluster. You will gather information related to your environment, which you must provide during the deployment of the IBM Spectrum Scale CSI driver. Make sure you write down his information, because you will need it during the configuration of the CR that will be used to deploy the CSI driver.

To configure the CR, follow these steps:

1. Switch back to `ibm-spectrum-scale-ns` namespace

   ```
   [root@opnshrh8 crds]# oc project ibm-spectrum-scale-ns

   Now using project "ibm-spectrum-scale-ns" on server
   "https://api.test.mop.fr.ibm.com:6443",
   ```

2. Retrieve the value of the following parameters.

   – Name of the filesystem in the CNSA cluster: `gpfs1` in our environment.

     This is the value of the "`Primary Fs`" parameter, which you will need to complete when you create the CSI driver.

   – Mount point in the CNSA cluster: `/mnt/gpfs1` in our environment.

     This is the value of the "`scaleHostPath`" parameter, which you will need to complete when you create the CSI driver.

   The command in Example 3-1 will return these values. Replace "`ibm-spectrum-scale-core-2vmxt`" with the name of one of your IBM Spectrum scale-core pods (`oc get pods -n ibm-spectrum-scale-ns`).

Example 3-1 Command to get the file system name and the mount point

```
[root@opnshrh8 ~]# oc exec ibm-spectrum-scale-core-2vmxt  -- mmremotefs show

Local Name  Remote Name  Cluster name       Mount Point        Mount Options    Automount  Drive  Priority
gpfs1       gpfsFS1      ClusterSpectrum.mop.fr.ibm.com /mnt/gpfs1         rw                yes
```

Example 3-2 shows a command that can be used to gather more information about your CNSA cluster installation

Example 3-2 Command to get the file system name and the mount point

```
[root@opnshrh8 ~]# oc exec ibm-spectrum-scale-core-2vmxt -- curl -k
https://ibm-spectrum-scale-gui.ibm-spectrum-scale-ns/scalemgmt/v2/filesystems/gpfs1?fields=mount
-u "csi-cnsa-gui-user:csi-cnsa-gui-password"
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
```

```
100   659  100   659    0     0    127      0  0:00:05  0:00:05 --:--:--   153
{
  "filesystems" : [ {
    "name" : "gpfs1",
    "mount" : {
      "additionalMountOptions" : "none",
      "automaticMountOption" : "yes",
      "mountPoint" : "/mnt/gpfs1",
      "mountPriority" : 0,
      "nodesMountedInternally" : [ ],
      "nodesMountedReadOnly" : [ ],
      "nodesMountedReadWrite" : [ "worker-0.test.mop.fr.ibm.com",
"worker-1.test.mop.fr.ibm.com", "worker-2.test.mop.fr.ibm.com", "spscale1", "spscale2" ],
      "readOnly" : false,
      "remoteDeviceName" : "ClusterSpectrum.mop.fr.ibm.com:gpfsFS1",
      "status" : "mounted"
    }
  } ],
  "status" : {
    "code" : 200,
    "message" : "The request finished successfully."
```

3. Retrieve the IBM Spectrum Scale CNSA cluster ID.

   [root@opnshrh8 ~]# oc exec ibm-spectrum-scale-core-2vmxt -- curl -s -k https://ibm-spectrum-scale-gui.ibm-spectrum-scale-ns/scalemgmt/v2/cluster -u "csi-cnsa-gui-user:csi-cnsa-gui-password" | grep clusterId

   "clusterId" : ***18306118698498958180,***

4. Connect to the IBM Spectrum Scale remote storage cluster and retrieve its ID.

   [root@spscale1 ~]# curl -s -k https://spscale1.mop.fr.ibm.com/scalemgmt/v2/cluster -u "csi-storage-gui-user:csi-storage-gui-password" | grep clusterId

   "clusterId" : ***15287841211602322336*,**

5. Retrieve the hostname of the GUI for the remote Spectrum Scale server. It's the same value that was used previously in the definition of the CR file aimed to deploy the CNSA cluster (2.4, "CNSA deployment" on page 15).

   In this documentation, the hostname is "spscale1.mop.fr.ibm.com". This is the value of the Rest API "GUI host" parameter for the remote cluster storage, which you will need to complete when you create the CSI driver.

6. Write down the value of CNSA GUI Host, which is "ibm-spectrum-scale-gui.ibm-spectrum-scale-ns". This is the value of the Rest API "GUI host" parameter for the primary cluster storage, which you will need to complete when you create the CSI driver.

7. Write down the values of the secrets created during the IBM Spectrum Scale CSI driver deployment.

   In this documentation, the values are "secret-cnsa-local" and "secret-storage-remote".

   – Use the "secret-cnsa-local" as a value to complete the **Secrets** section for the Primary cluster when you create the CSI driver.

   – Use the "secret-storage-remote" as a value for the Remote cluster in the "Deployment" on page 29.

Deployment

To deploy the CSI driver, using Red Hat OCP Web console, follow these steps.

1. Login into the **Red Hat OPENSHIFT** web console as an administrator user and go to the Installed Operators panel.
2. Select **IBM CSI Spectrum Scale Driver**.

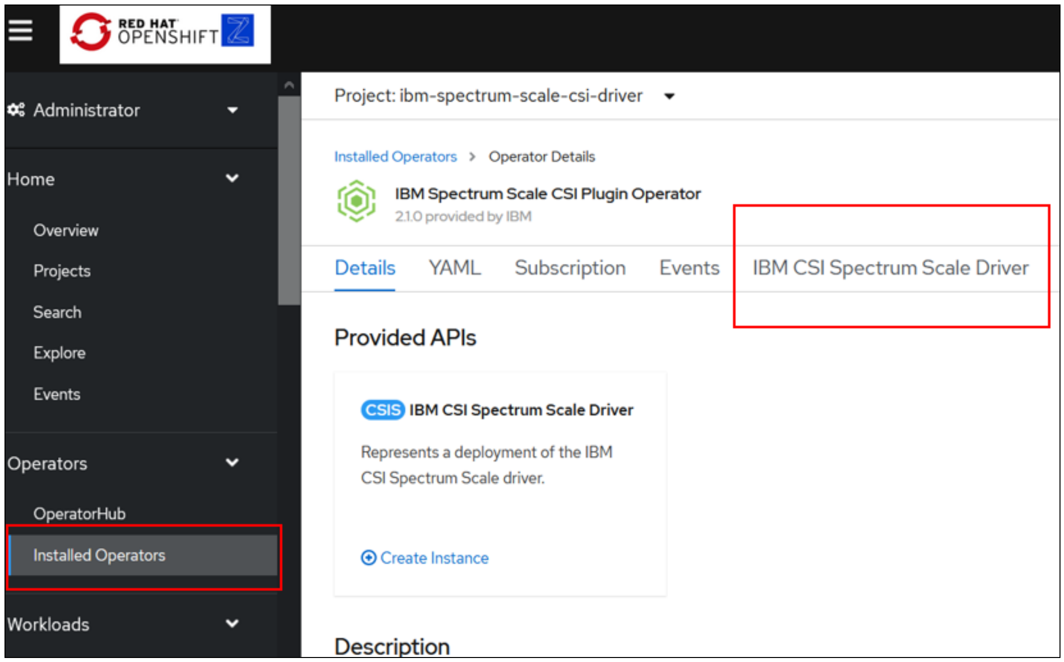


Figure 3-6 Open the IBM CSI Spectrum Scale driver

3. Click **Create CSIScaleOperator.**

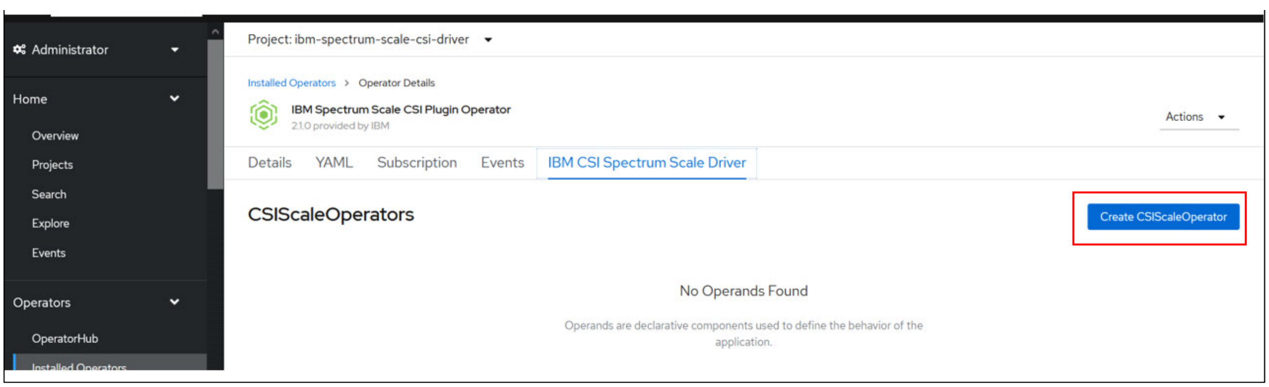


Figure 3-7 Create the operator to start the deployment process

4. Navigate through the create-operator process and enter the values that you collected in "Configuration of the custom resource (CR)" on page 27. The entries are shown in the following sequence of screen shots.
   a. Enter the required fields for **Name** and **Spectrum Scale Hostpath**.

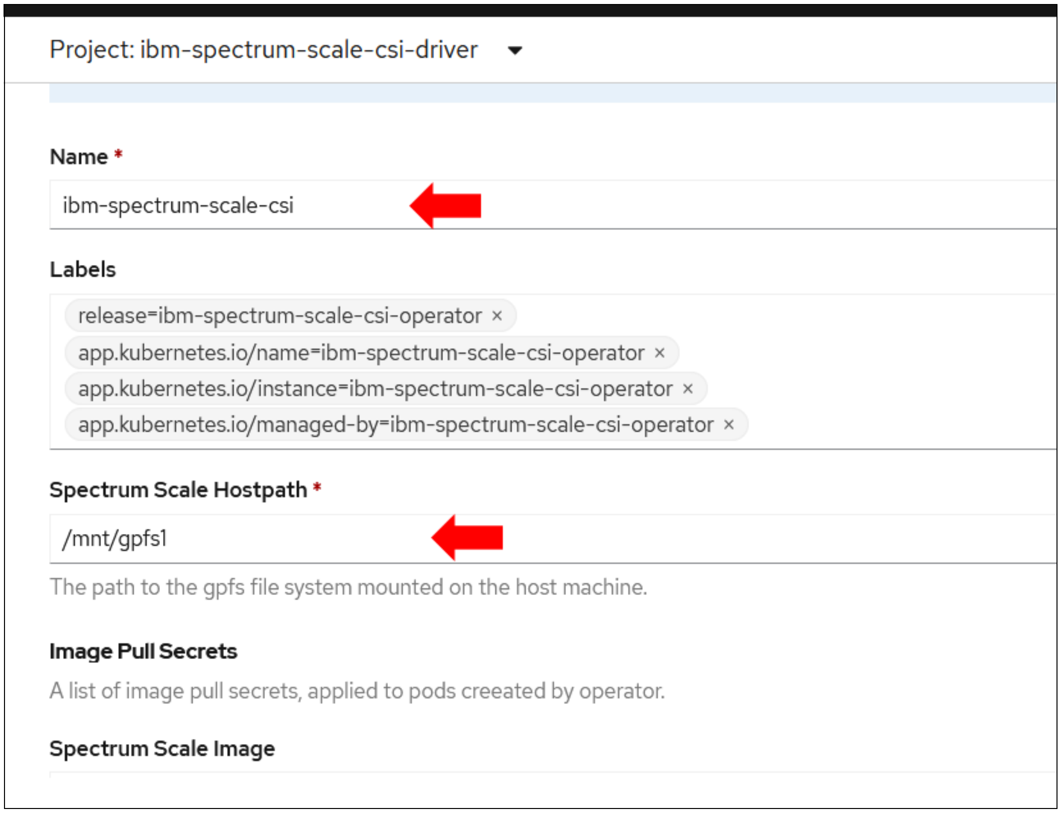


Figure 3-8 Operator values

b. In the **Cluster** section, enter the cluster ID in the ID field.

c. In the **Primary** section:

- Enter the primary file system in the **Primary Fs** field.
- Enter the remote cluster ID in the **Remote cluster** field.

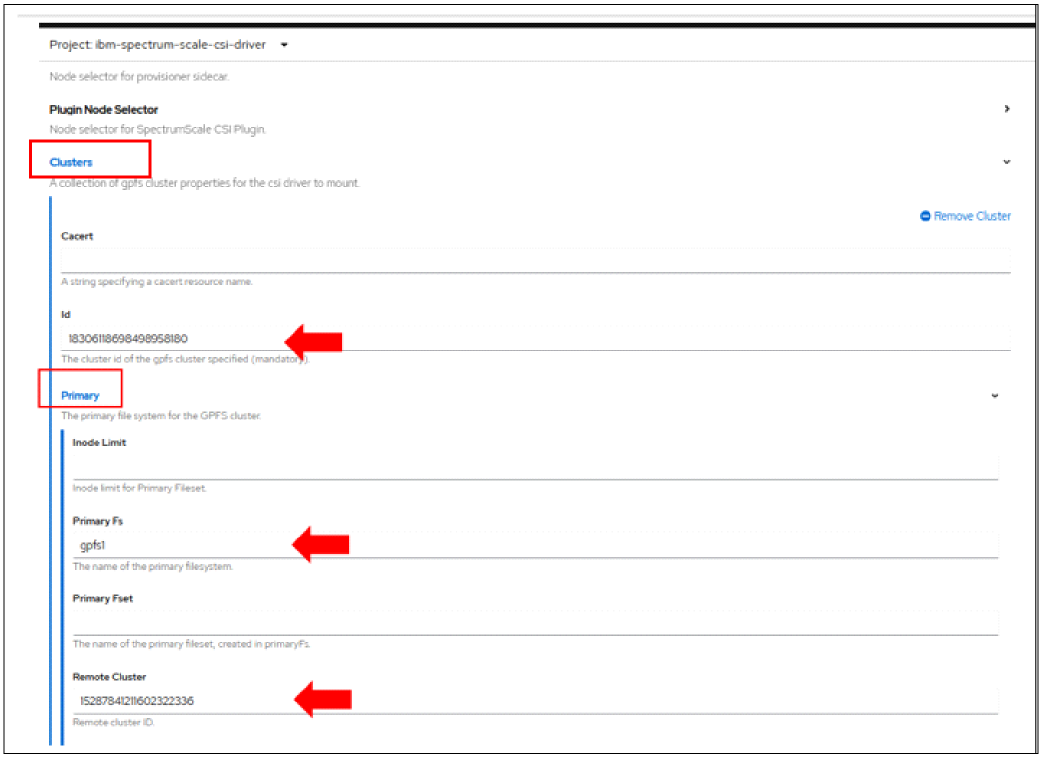


Figure 3-9 Operator values - continued

d. In the **Rest Api** section, enter the following information:

- **Gui Host**: hostname of the REST server
- **Secrets**: name of the secret created for the CNSA GUI user

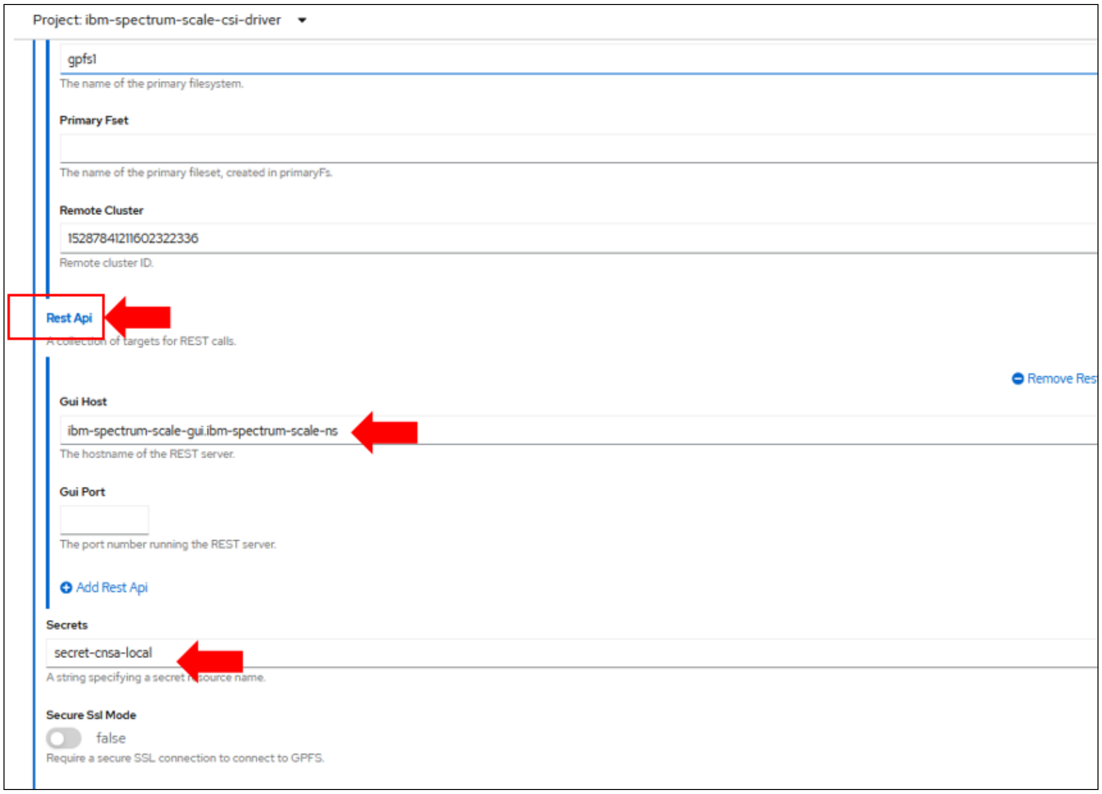


Figure 3-10 Operator values - continued

e. Click **Add Cluster** to add the IBM Spectrum Scale remote storage cluster.

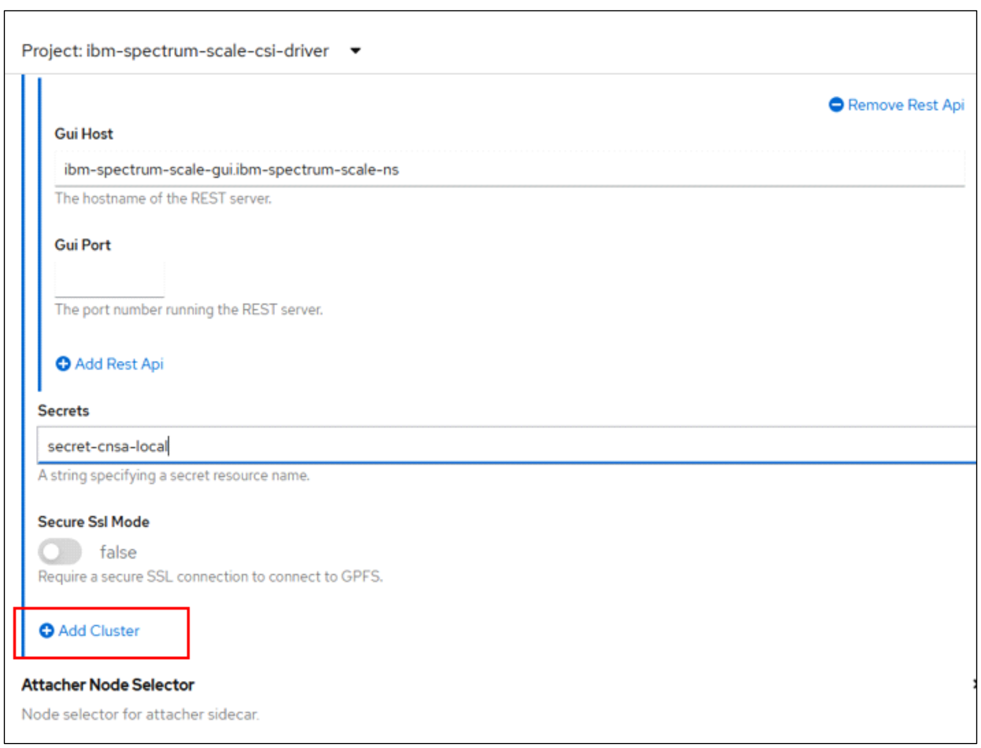


Figure 3-11 Operator values - continued

f. Enter the cluster ID of the gpfs cluster in the **id** field.

g. Enter the following information for the **Rest API**:

- **Gui Host**: hostname of the REST server
- **Secrets**: name of the secret created for the CNSA GUI user

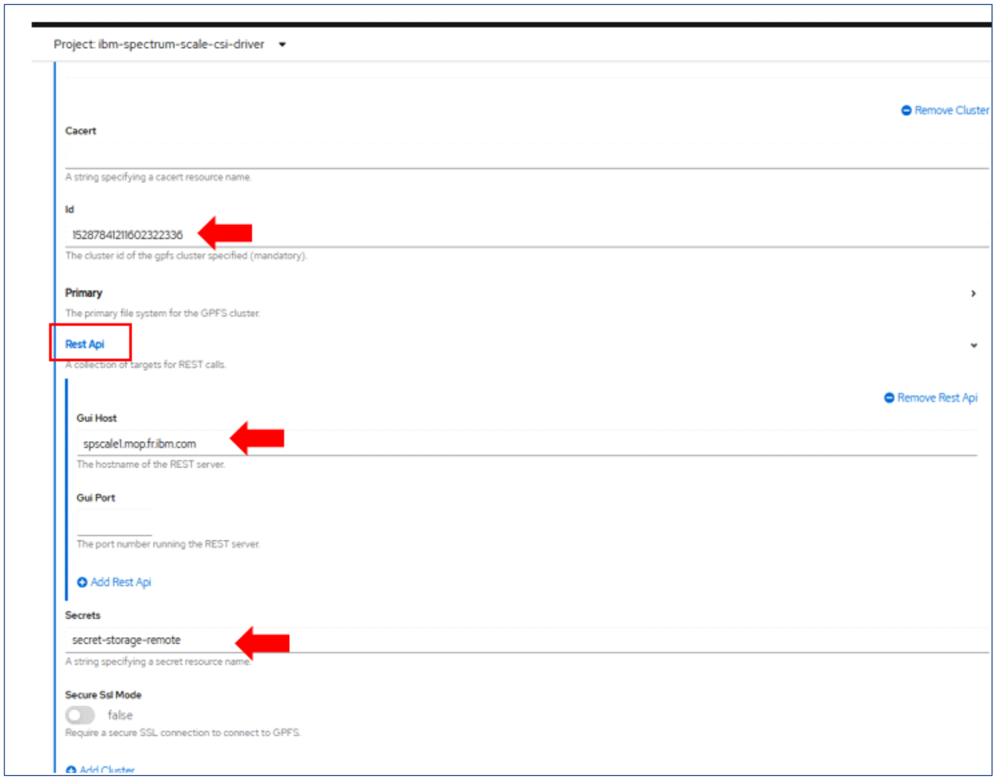


Figure 3-12 Operator values - continued

5. Click **Create** to start the deployment of the CSI driver.

Steps 1 through 5 show the form view of the process to create and deploy the CSI driver. Figure 3-13 shows the yaml file view.

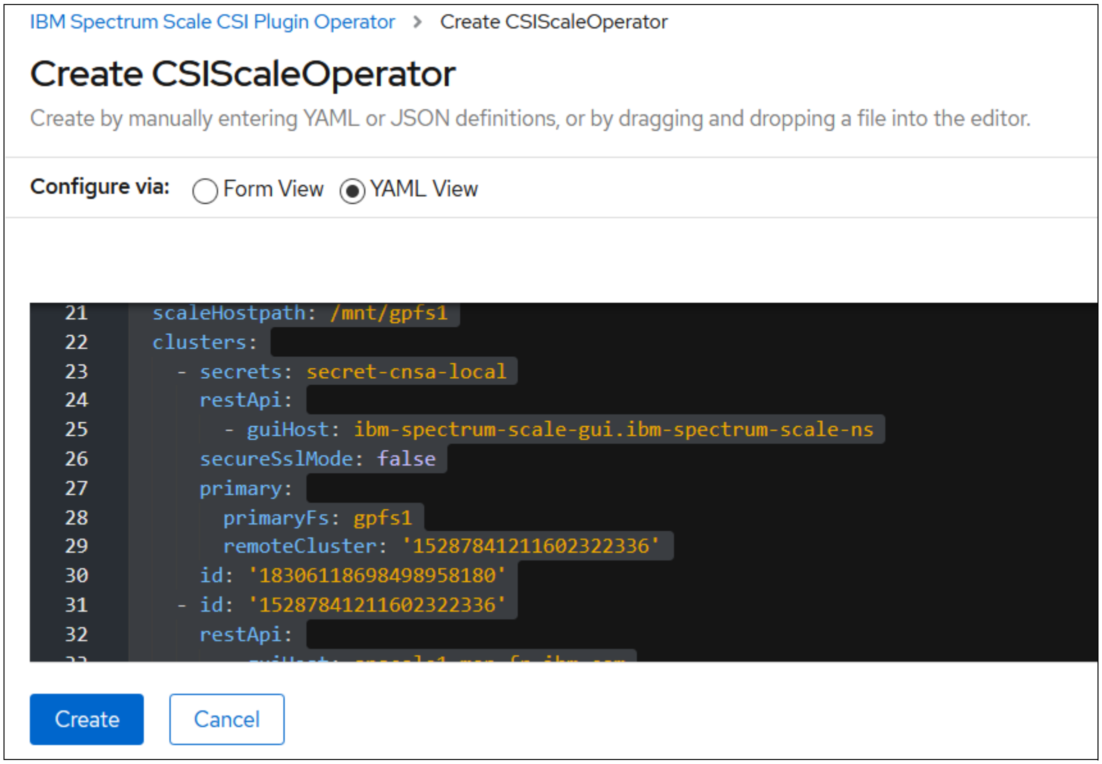


Figure 3-13 Create the CSI driver using the yaml view

Example 3-3 shows the complete `yaml` file.

Example 3-3 yaml file to create the CSI driver

```
apiVersion: csi.ibm.com/v1
kind: CSIScaleOperator
metadata:
namespace: ibm-spectrum-scale-csi-driver
name: ibm-spectrum-scale-csi
labels:
release: ibm-spectrum-scale-csi-operator
app.kubernetes.io/name: ibm-spectrum-scale-csi-operator
app.kubernetes.io/instance: ibm-spectrum-scale-csi-operator
app.kubernetes.io/managed-by: ibm-spectrum-scale-csi-operator
spec:
provisionerNodeSelector:
- key: scale
value: 'true'
attacherNodeSelector:
- key: scale
value: 'true'
pluginNodeSelector:
- key: scale
value: 'true'
scaleHostpath: /mnt/gpfs1
clusters:
- secrets: secret-cnsa-local
restApi:
- guiHost: ibm-spectrum-scale-gui.ibm-spectrum-scale-ns
```

```
secureSslMode: false
primary:
primaryFs: gpfs1
remoteCluster: '15287841211602322336'
id: '18306118698498958180'
- id: '15287841211602322336'
restApi:
- guiHost: spscale1.mop.fr.ibm.com
secrets: secret-storage-remote
status: {}
```

Figure 3-14 shows that the CSI driver has been deployed.

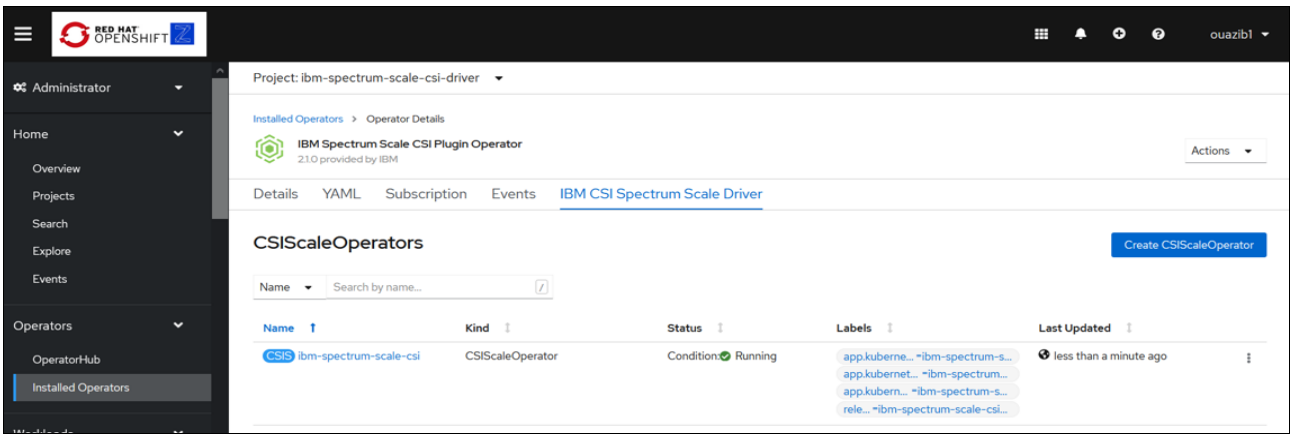


Figure 3-14 CSI driver has deployed

To verify that the IBM Spectrum Scale CSI driver has been deployed successfully, the pods highlighted in Figure 3-15 must be in a running state.

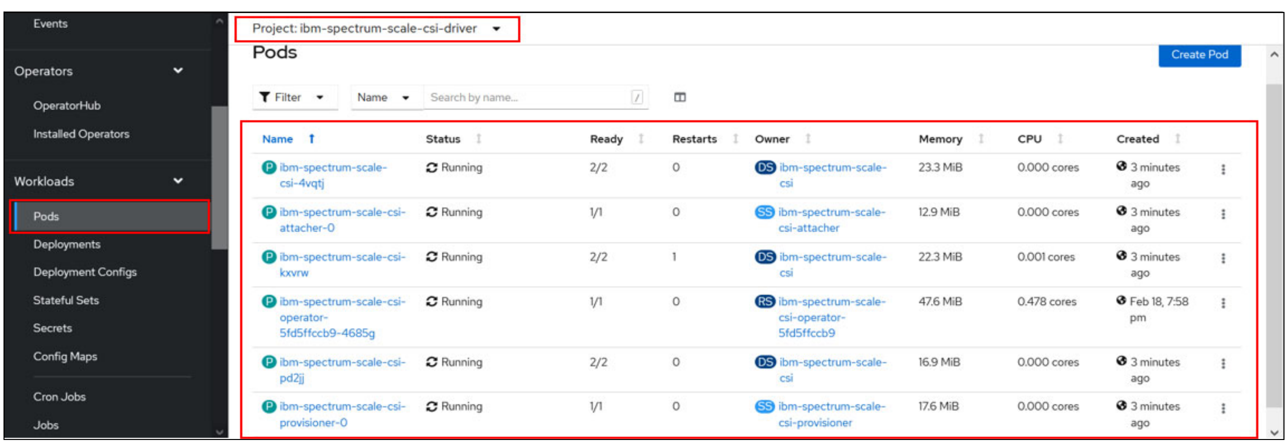


Figure 3-15 Pod status

4

Dynamic provisioning with IBM Spectrum Scale CSI

This chapter describe how to use IBM Spectrum CSI driver to dynamically provision storage volumes for your application running in OCP, by using storage class and Persistent Volume Claims (PVCs).

The steps to implement the dynamic provisioning with the CSI driver consist of:

- “Creating the dynamic storage class” on page 38
- “Creating the PVC” on page 39

4.1 Creating the dynamic storage class

To create the dynamic storage class, follow these steps:

1. Using the OCP Web Console, log in to the OCP Web console as an administrator user, select **Storage Classes**, and click **Create Storage Class**.

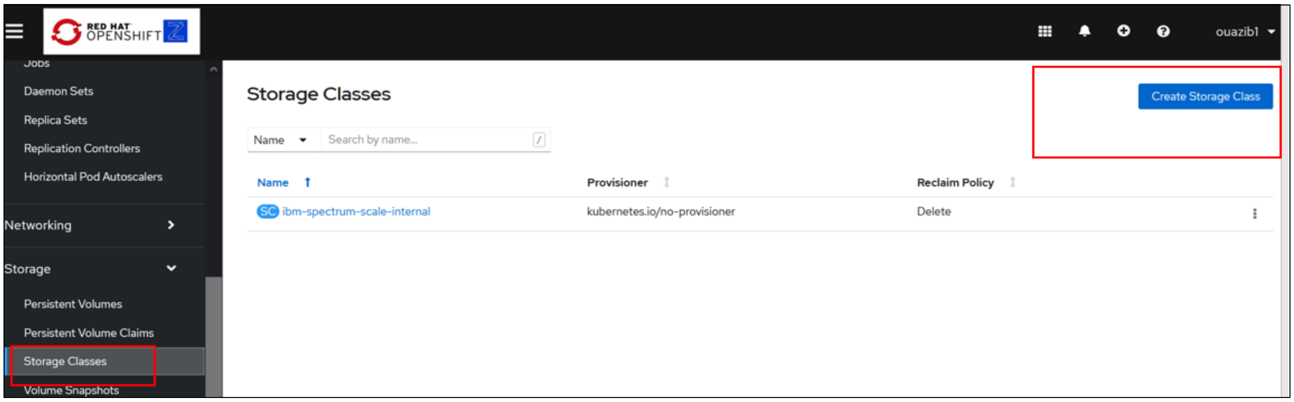


Figure 4-1 Creating the dynamic storage class

2. Select **Edit YAML**.

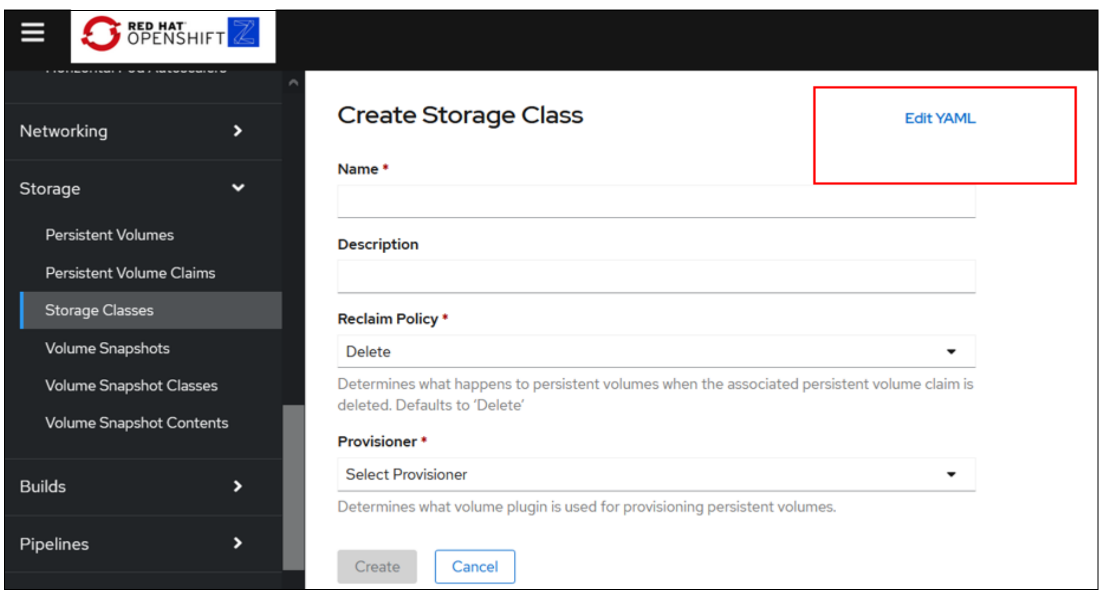


Figure 4-2 Edit the YAML file

3. Add the following content to the `yaml` file and customize this content to match your environment. The `clusterId` value is the ID of the remote storage server (Figure 4-3).

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: ibm-spectrum-scale-csi-fileset
provisioner: spectrumscale.csi.ibm.com
parameters:
  volBackendFs: "gpfs1"
  clusterId: "15287841211602322336"
```

```
reclaimPolicy: Delete
```

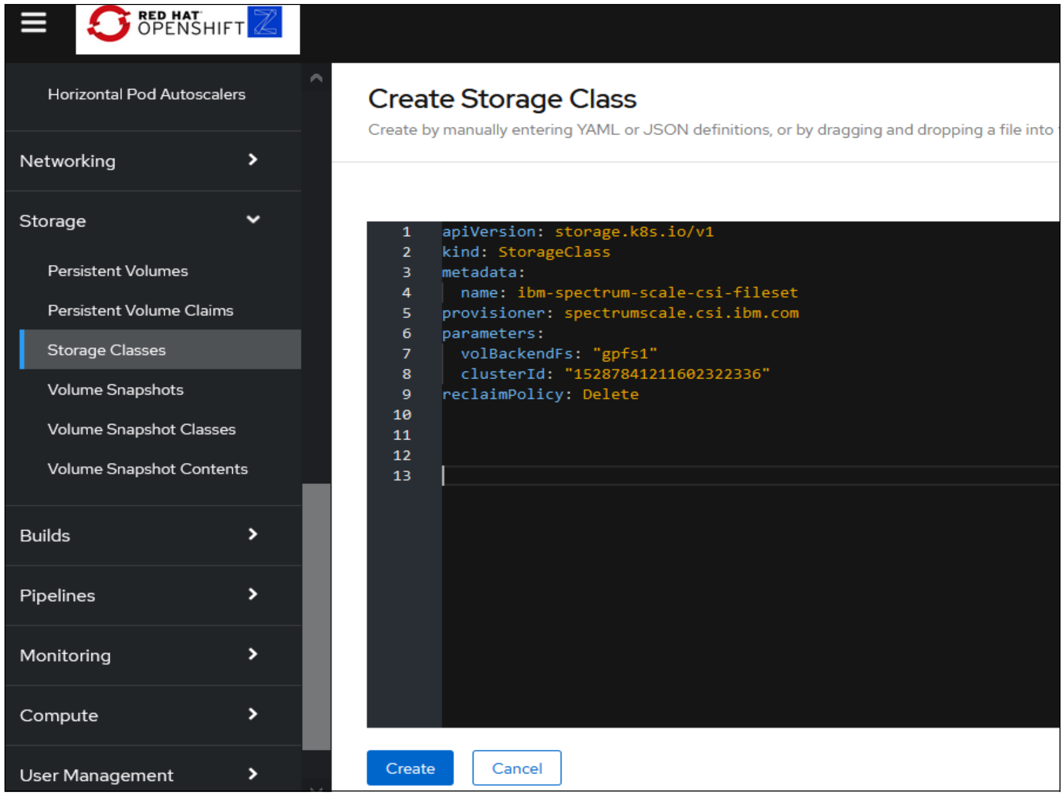


Figure 4-3 Create a storage class

4. Click **Create**.

4.2 Creating the PVC

To create a PVC for your application that needs persistent storage, follow these steps:

1. In the console, select **Storage > Persistent Voliume Claims**.

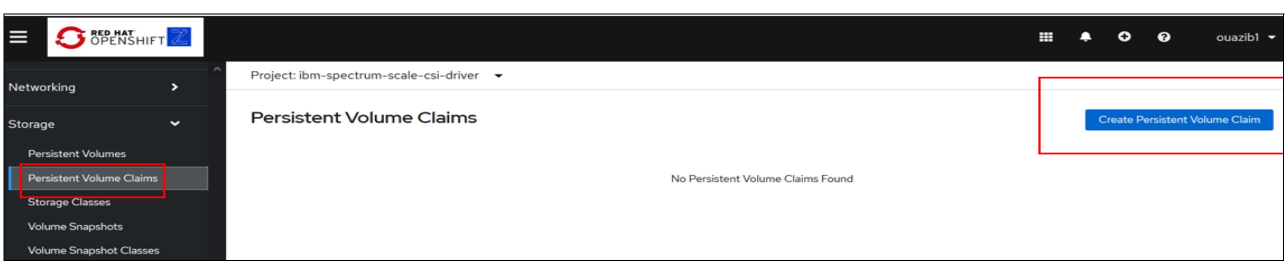


Figure 4-4 Create a persistent volume claim

2. Complete the settings and click **Create**.

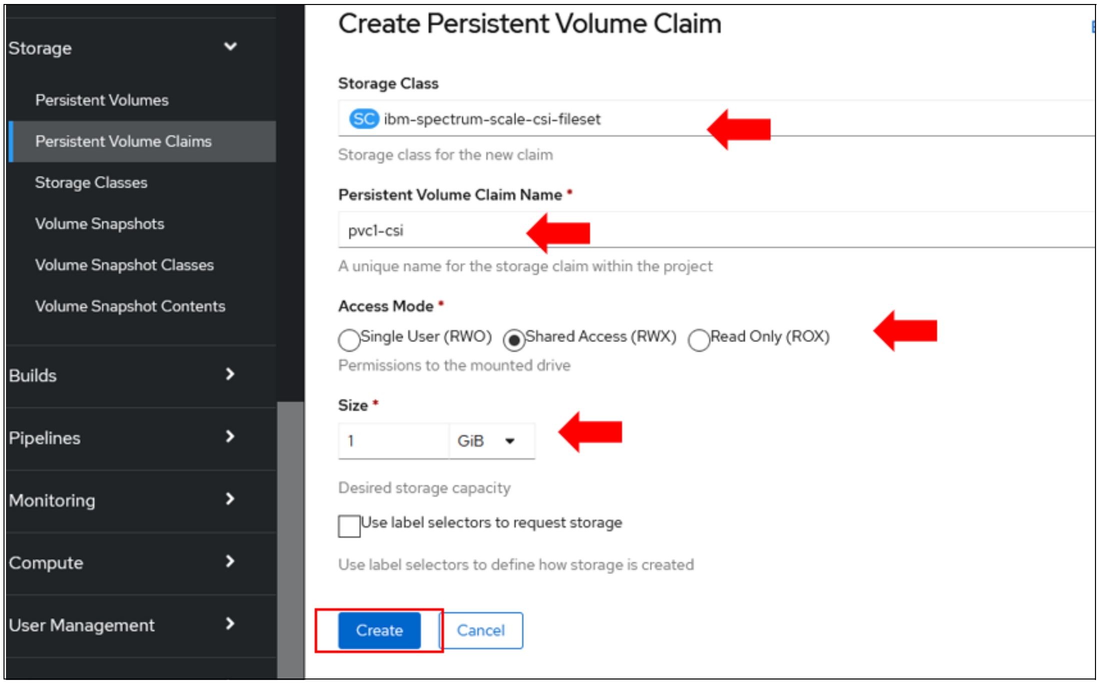


Figure 4-5 Create the persistent volume claim

3. When the PVC is created, the **Persistent Volumes** panel shows that a PV was automatically created and bounded to this PVC.

Congratulations, your PVC is ready to be mapped to your application. From now on, you can use IBM Spectrum Scale as persistent volume storage on Red Hat OCP cluster!

A

Commands to uninstall and clean the environment

This appendix contains information for uninstalling the IBM Spectrum Scale CSI Driver, IBM Spectrum Scale CNSA, and to clean the worker nodes and the remote storage server. These will be helpful in a test environment.

Commands to uninstall the IBM Spectrum Scale CSI driver

1. kubectl delete -f csiscaleoperators.csi.ibm.com_cr.yaml
2. kubectl delete -f ibm-spectrum-scale-csi-operator.yaml
3. kubectl delete namespace ibm-spectrum-scale-csi-driver

Commands to uninstall IBM Spectrum Scale CNSA

1. oc delete -f spectrumscale/deploy/crds/ibm_v1_scalecluster_cr.yaml -n ibm-spectrum-scale-ns
2. oc delete -f spectrumscale/deploy/operator.yaml -n ibm-spectrum-scale-ns
3. oc delete project ibm-spectrum-scale-ns
4. oc delete -f spectrumscale/deploy/crds/ibm_v1_scalecluster_crd.yaml -n ibm-spectrum-scale-ns
5. oc delete -f spectrumscale/deploy/cluster_role_binding.yaml -n ibm-spectrum-scale-ns
6. oc delete -f spectrumscale/deploy/cluster_role.yaml -n ibm-spectrum-scale-ns
7. oc delete -f spectrumscale/deploy/role_binding.yaml -n ibm-spectrum-scale-ns
8. oc delete -f spectrumscale/deploy/role.yaml -n ibm-spectrum-scale-ns
9. oc delete -f spectrumscale/deploy/role_binding_scale_core.yaml -n ibm-spectrum-scale-ns
10. oc delete -f spectrumscale/deploy/role_scale_core.yaml -n ibm-spectrum-scale-ns
11. oc delete scc ibm-spectrum-scale-restricted
12. oc delete scc ibm-spectrum-scale-privileged
13. oc get pv -l app=scale-pmcollector
14. oc delete pv -l app=scale-pmcollector
15. oc get pvc | grep pmcollector
16. oc delete pvc <pvc-name>
17. oc delete sc -l app=scale-pmcollector

Clean the worker nodes

1. oc get nodes -l 'node-role.kubernetes.io/worker=' -o jsonpath="{range .items[*]}{.metadata.name}{'\n'}"

```
worker_node_name1
worker_node_name2
```

2. Based on the output of the previous command, run the following command on each node displayed.

   Examples:

   – oc debug node/<worker_node_name1> -T -- chroot /host sh -c "rm -rf /var/mmfs; rm -rf /var/adm/ras"

– `oc debug node/<worker_node_name2> -T -- chroot /host sh -c "rm -rf /var/mmfs; rm -rf /var/adm/ras"`

3. `oc get nodes -ojsonpath="{range .items[*]}{.metadata.name}{'\n'}" | xargs -I{} oc annotate node {} scale.ibm.com/nodedesc-`

Clean the remote storage server

1. Go to the IBM Spectrum Scale remote storage server and run the following command.

```
mmauth show all | grep ibm-spectrum-scale
Cluster name:        XX
```

2. Based on the output of the previous command, run the following command

```
mmauth delete XX
```

REDP-5645-00

ISBN 0738459844

Printed in U.S.A.

Get connected

Redbooks®

ibm.com/redbooks