SUSE

2025.10.15

SUSE Virtualization and Portworx Enterprise by Pure Storage

Integrating a Pure Storage FlashArray with SUSE Virtualization

Gopala Krishnan, Partner Solution Architect, SUSE

Suresh S, Partner Solution Architect, SUSE

Terry Smith, Ecosystem Partner Solution Innovation Director, SUSE

Bhumitra Nagar, Member of Technical Staff, Portworx by Pure Storage

James McShane, Consulting Cloud Native Architect, Portworx by Pure Storage

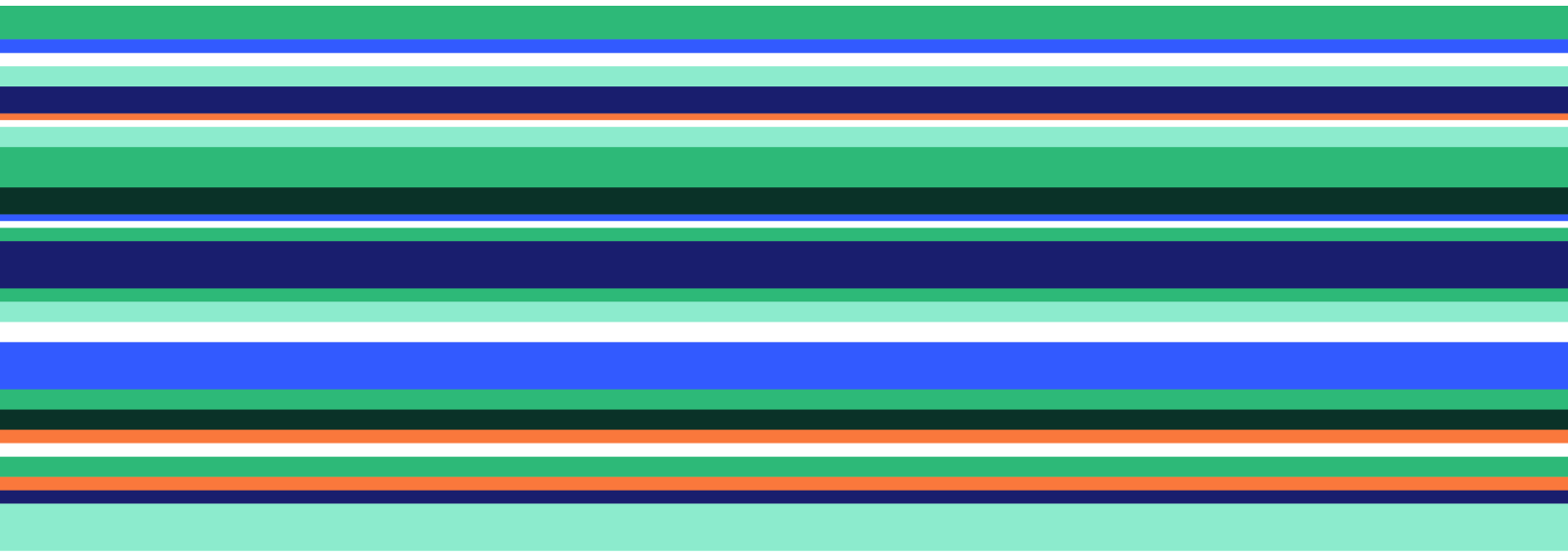


Table of Contents

Copyright © 2025 SUSE

Introduction

As organizations increasingly adopt cloud-native and containerized workloads, the demand for robust, enterprise-grade storage solutions in virtualized environments continues to grow. SUSE Virtualization, powered by Harvester and Kubernetes, delivers a flexible platform for running both virtual machines and modern containerized applications on a unified infrastructure. To unlock advanced storage features and ensure seamless data management across diverse environments, integrating Portworx Enterprise brings unparalleled benefits—ranging from high availability and data protection to hybrid cloud mobility and multi-cluster support.

In this document administrators and platform engineers learn how to deploy, configure, and validate a SUSE Virtualization environment with Portworx Enterprise and Pure Storage FlashArray, enabling them to deliver resilient, scalable, and efficient storage services for mission-critical workloads in virtualized and cloud-native environments.

This integrated solution addresses a wide range of scenarios, including:

- **Application Modernization:** Seamlessly migrating or running legacy VM workloads and stateful cloud-native applications on the same infrastructure.
- **Hybrid and Multi-Cloud Deployments:** Enabling workload and data mobility across on-premises, public cloud, and edge locations.
- **Business Continuity and Disaster Recovery:** Providing robust backup, granular restore, and cross-site failover for both VMs and containers.
- **Dev/Test and Self-Service Environments:** Allowing developers to quickly provision persistent, production-grade storage for any workload type.
- **Data-Intensive Workloads:** Supporting databases, analytics, and AI/ML apps that require high availability and consistent performance.

Scope

This document provides comprehensive instructions for installing Portworx Enterprise on a SUSE Virtualization cluster with a Pure Storage FlashArray. It details the necessary steps for configuring multipathing, deploying the Portworx operator and StorageCluster, and updating SUSE Virtualization's storage configurations.

Copyright © 2025 SUSE

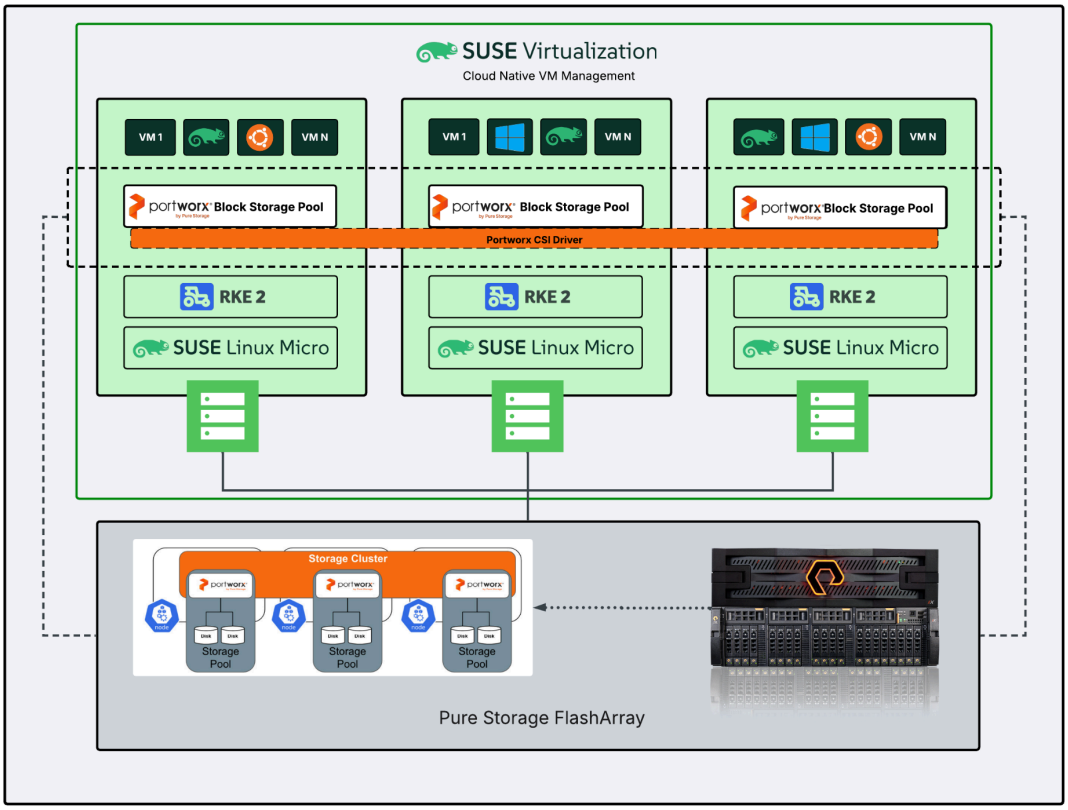
Audience

The intended audience for this guide are system administrators, platform engineers, DevOps engineers, and IT professionals responsible for designing, deploying, managing, and maintaining a modern infrastructure environment for container and virtual machine workloads. A basic understanding of Kubernetes and storage concepts is assumed.

Copyright © 2025 SUSE

Overview

This practical guide illustrates how SUSE Virtualization can leverage a Pure Storage FlashArray to provide a robust, persistent, block storage for virtual machine and container workloads. The environment is illustrated in the diagram below, showing a highly available, three-node SUSE Virtualization cluster connected to the Pure Storage FlashArray through a dedicated storage network.



Key components of this architecture are:

SUSE® Virtualization

SUSE Virtualization (formerly Harvester) is a modern, open, interoperable, hyperconverged infrastructure (HCI) solution built on Kubernetes. SUSE Virtualization leverages other components (such as SUSE Linux Micro and RKE2) to deliver a secure, resilient, and scalable platform for managing virtual machine and container workloads.

Copyright © 2025 SUSE

Access the SUSE Virtualization documentation for detailed technical information, including Hardware and Network Requirements and installation guidance. This guide references SUSE Virtualization v1.5.1 and later.

Portworx Enterprise and Portworx Operator

Portworx Enterprise delivers elastic scalability, industry-leading availability, and self-service access to any storage infrastructure for the most widely used Kubernetes distribution. This fully-integrated storage solution offers automated capacity management, thin provisioning, and flexibility across hybrid, multi-cloud, and on-premises deployments.
This guide references Portworx Enterprise 3.4.0 or later and Portworx Operator 25.2.2 or later.

Pure Storage FlashArray

Pure FlashArray delivers high-performance, scalable, all-flash storage. Your Pure FlashArray must be configured to meet the Portworx environment prerequisites and be accessible from the SUSE Virtualization cluster nodes.

In the following pages, you will learn how to configure your SUSE Virtualization environment to leverage the Pure Storage FlashArray, including:

- Preparing the Pure Storage FlashArray
- Deploying and configuring the Portworx Operator and the Portworx Cluster
- Defining a Portworx StorageClass
- Deploying and validating a virtual machine

Copyright © 2025 SUSE

Preparation

Before starting the installation, perform the following preparatory steps:

1. Prepare the Pure Storage FlashArray Multipath configuration.

    a. Create a Harvester CloudInit resource to deploy node-level configuration files with the following contents:

```
None
apiVersion: node.harvesterhci.io/v1beta1
kind: CloudInit
metadata:
  name: pure-multipath
spec:
  contents: |
    stages:
      network:
      - name: "Configure pure storage"
        files:
        - path: /etc/udev/rules.d/99-pure-storage.rules
          permissions: 0644
          content: |
            #ACTION=="change", SUBSYSTEM=="scsi",
ENV{SDEV_UA}=="INQUIRY_DATA_HAS_CHANGED", TEST=="rescan", ATTR{rescan}="x"
            ACTION=="change", SUBSYSTEM=="scsi",
ENV{SDEV_UA}=="CAPACITY_DATA_HAS_CHANGED", TEST=="rescan", ATTR{rescan}="x"
            #ACTION=="change", SUBSYSTEM=="scsi",
ENV{SDEV_UA}=="THIN_PROVISIONING_SOFT_THRESHOLD_REACHED", TEST=="rescan",
ATTR{rescan}="x"
            #ACTION=="change", SUBSYSTEM=="scsi",
ENV{SDEV_UA}=="MODE_PARAMETERS_CHANGED", TEST=="rescan", ATTR{rescan}="x"
            ACTION=="change", SUBSYSTEM=="scsi",
ENV{SDEV_UA}=="REPORTED_LUNS_DATA_HAS_CHANGED", RUN+="scan-scsi-target
$env{DEVPATH}"
        - path: /etc/multipath.conf
          content: |
            defaults {
              user_friendly_names no
              enable_foreign "^$"
                    polling_interval    10
            }
            devices {
                device {
                    vendor                  "NVME"
                    product                 "Pure Storage FlashArray"
                    path_selector           "queue-length 0"
                    path_grouping_policy    group_by_prio
                    prio                    ana
                    failback                immediate
```


Copyright © 2025 SUSE

```
                fast_io_fail_tmo            10
                user_friendly_names         no
                no_path_retry               0
                features                    0
                dev_loss_tmo                60
            }
            device {
                vendor                  "PURE"
                product                 "FlashArray"
                path_selector           "service-time 0"
                hardware_handler        "1 alua"
                path_grouping_policy    group_by_prio
                prio                    alua
                failback                immediate
                path_checker            tur
                fast_io_fail_tmo        10
                user_friendly_names     no
                no_path_retry           0
                features                0
                dev_loss_tmo            600
            }
        }
        blacklist_exceptions {
                property "(SCSI_IDENT_|ID_WWN)"
        }
        blacklist {
              devnode "^pxd[0-9]*"
              devnode "^pxd*"
              device {
                vendor "VMware"
                product "Virtual disk"
              }
        }
      permissions: 0644
  - name: "Start multipathd service"
    systemctl:
      enable:
      - multipathd
      start:
      - multipathd
filename: 99_multipathd.yaml
matchSelector: {}
```

b. Apply the CloudInit resource resource to the SUSE Virtualization cluster.

Shell

```
kubectl apply -f pure-multipath.yaml
```


Copyright © 2025 SUSE

c. Restart the cluster nodes.

2. Set up user access in the Pure Storage FlashArray.
3. Create a Kubernetes secret, named px-pure-secret, containing your Pure Storage FlashArray credentials. It will be added to the Portworx namespace. Run the following kubectl command to do this:

```shell
kubectl create secret generic px-pure-secret --namespace portworx
--from-file=pure.json
```


Copyright © 2025 SUSE

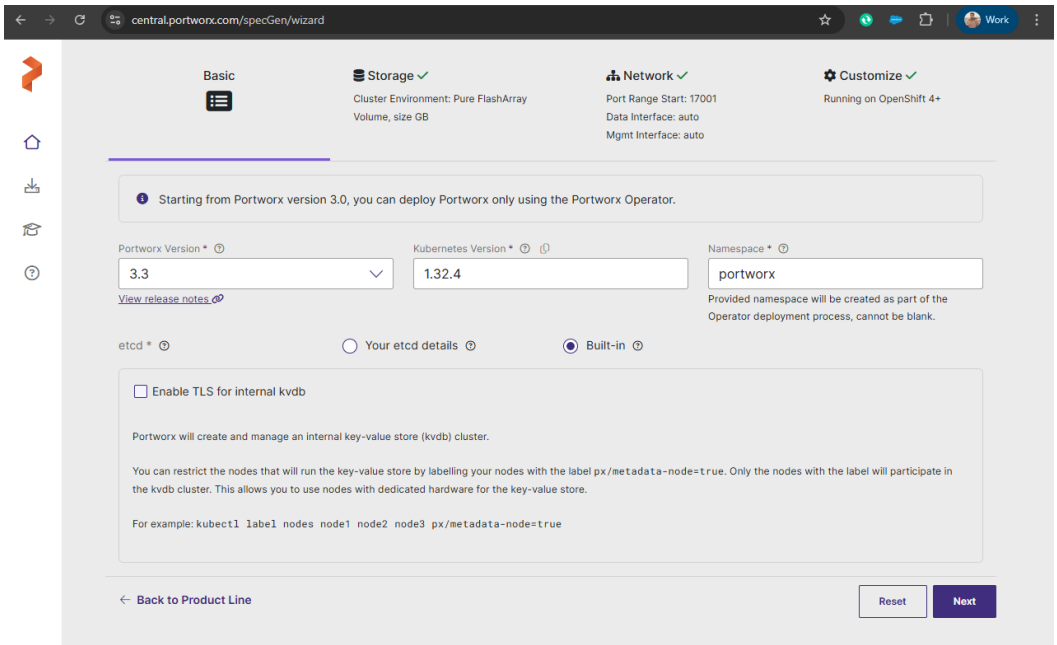
Installation

Portworx Enterprise is a cloud-native storage and data management platform for Kubernetes. It offers native Container Storage Interface (CSI) integration, providing drivers that expose block and file storage systems to workloads running on Kubernetes, such as the containers and virtual machines in the SUSE Virtualization environment.

The Portworx Operator is designed to efficiently manage the installation and upgrade workflow of all Portworx components. The following pages provide guidance for configuring and installing Portworx Enterprise via the Portworx Operator.

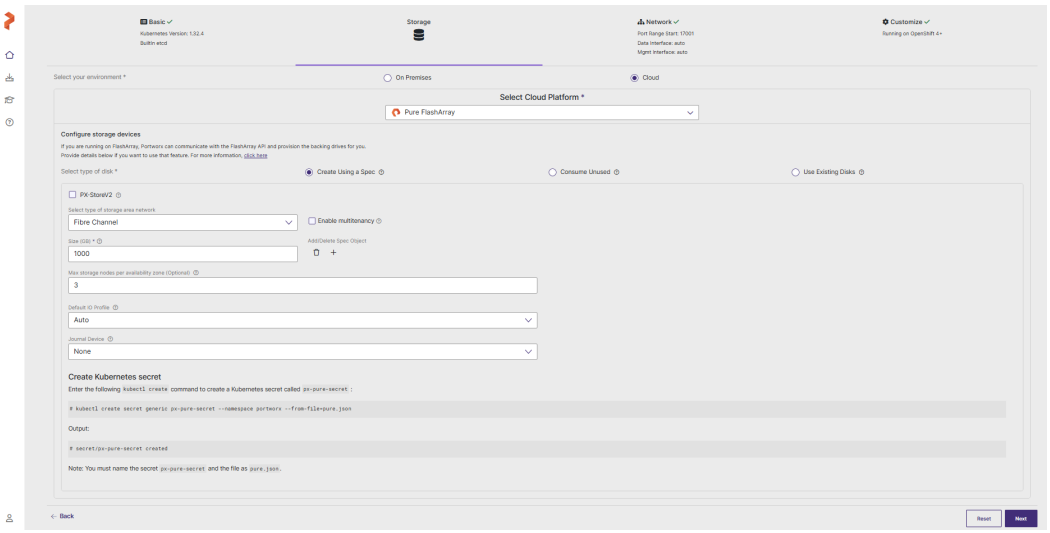
Configure the Portworx Operator

1. Log in to Portwork Central https://central.portworx.com/.
2. Select *Portworx Enterprise* → *Generate Spec*, then select the Portworx Version and choose *Pure Storage FlashArray* for Platform.
3. Click *Customize* at the bottom of the menu.
4. After the Portworx version is validated, specify the Kubernetes version for your SUSE Virtualization cluster. Then, provide the desired name for the namespace and select the *Built-in* etcd option.

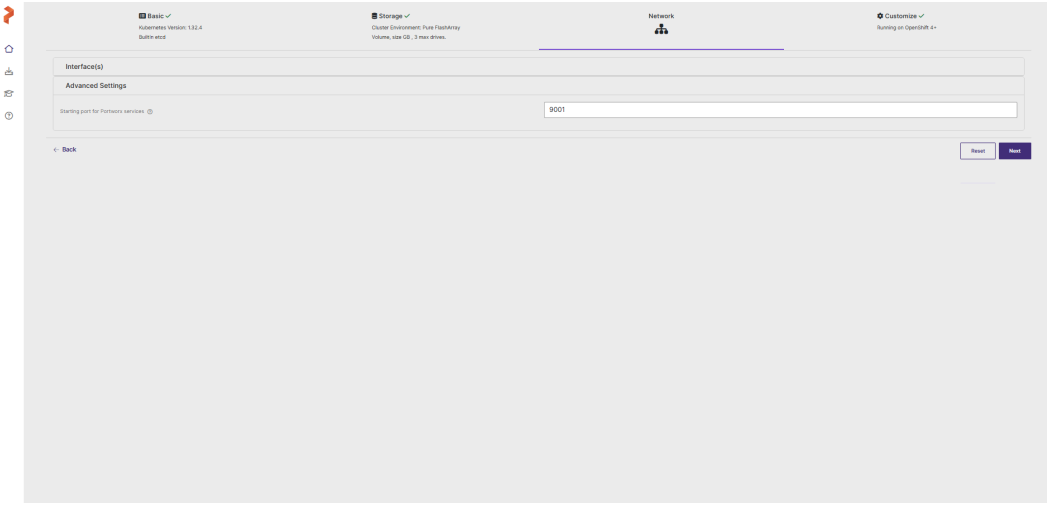


Copyright © 2025 SUSE

5. In the *Storage* section, select *Next* and ensure *Pure Storage FlashArray* is chosen. Under the Select type of disk selection, enable *Create Using a Spec*. Then, enable *PX-Storev2* and select *Fiber Channel* as the storage area network. Enter the size of the disk mapped to the SUSE Virtualization cluster. Finally, update the maximum storage nodes per availability node to *3* for your 3-node cluster.

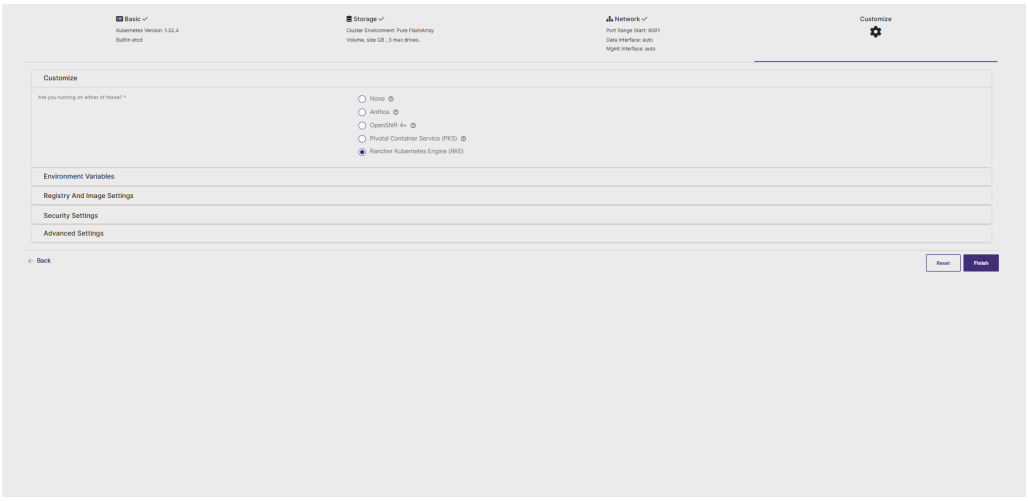


6. Proceed to the *Network* section and configure the Portworx service port. The default port is *9001*.

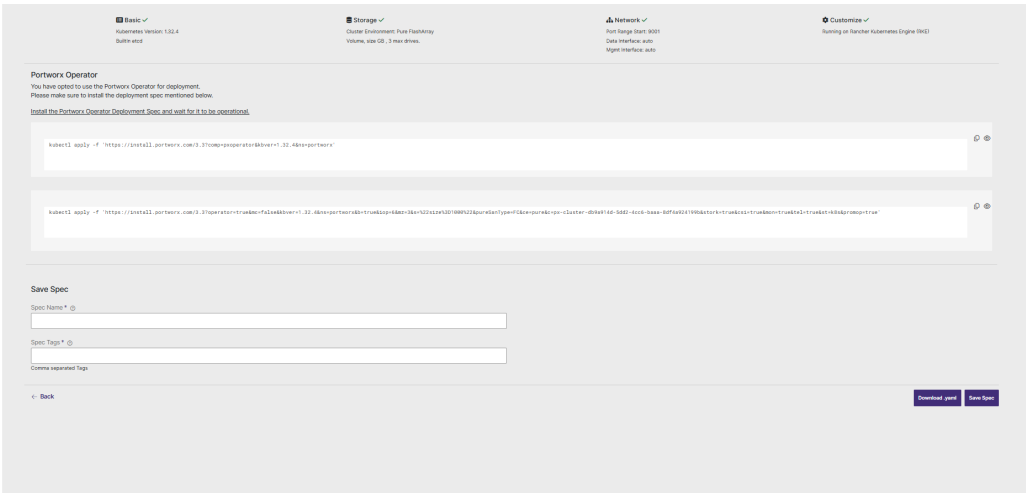


Copyright © 2025 SUSE

7. Select *Rancher Kubernetes Engine (RKE)* in the *Customize* section.



8. When the kubectl commands appear in the UI, they are ready to be applied, and you may also save them for later reference.



Install the Portworx Operator and the Portworx Cluster

1. Log in to the SUSE Virtualization management node command-line interface (CLI) with root access.

2. Run the first kubectl command saved from the previous section to install the Portworx Operator.
   For example:

Copyright © 2025 SUSE

```
Shell
kubectl apply -f
'https://install.portworx.com/3.3?comp=pxoperator&kbver=1.3.0&ns=portworx'
```

You should see the following as the operator is deployed:

```
None
namespace/portworx created
serviceaccount/portworx-operator created
clusterrole.rbac.authorization.k8s.io/portworx-operator created
clusterrolebinding.rbac.authorization.k8s.io/portworx-operator created
deployment.apps/portworx-operator created
```

3. If you are using SUSE Virtualization v1.6.0 or above, add a Stork Snapshot StorageClass, otherwise skip to the next step.

    a. Create a file, named *stork-snapshot-sc.yaml*, with the following contents:

```
None
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: stork-snapshot-sc
  annotations:
    cdi.harvesterhci.io/storageProfileVolumeModeAccessModes: |
      {"Block":["ReadWriteOnce"]}
provisioner: stork-snapshot
allowVolumeExpansion: true
reclaimPolicy: Delete
volumeBindingMode: Immediate
```

    b. Apply the StorageClass using.

```
Shell
kubectl apply -f stork-snapshot-sc.yaml
```

    c. You should see:

Copyright © 2025 SUSE

```
None
storageclass.storage.k8s.io/stork-snapshot-sc created
```

4. Run the second kubectl command from the previous section to install the Portworx cluster.
   For example:

```
None
kubectl apply -f
'https://install.portworx.com/3.3?operator=true&mc=false&kbver=1.3.0&ns=portwor
x&b=true&iop=6&mz=3&s=%22size%3D1000%22&pureSanType=FC&ce=pure&dmthin=true&c=px
-cluster-82af5d7d-f249-4f45-9e80-d1590e7bfce4&stork=true&csi=true&mon=true&tel=
true&st=k8s&promop=true'
```

   This should result in output similar to:

```
None
storagecluster.core.libopenstorage.org/px-cluster-82af5d7d-f249-4f45-9e80-d1590
e7bfce4 created
```

Add the Portworx StorageClass

1. Ensure you are still logged in to the SUSE Virtualization management CLI as root.
2. Create a file, named *portworx-storageclass.yaml*, with the following contents:

```
None
allowVolumeExpansion: true
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  annotations:
    storageclass.kubernetes.io/is-default-class: "true"
  name: data-disk-storage
parameters:
  cdi.kubevirt.io/storage.contentType: kubevirt
  nodiscard: "true"
  repl: "2"
provisioner: pxd.portworx.com
reclaimPolicy: Delete
volumeBindingMode: Immediate
```


Copyright © 2025 SUSE

3. Apply the StorageClass using:

```
Shell
kubectl apply -f portworx-storageclass.yaml
```

You should see:

```
None
storageclass.storage.k8s.io/data-disk-storage created
```

You may see a “request is invalid” error like the one below:

```
None
The request is invalid : default storage class harvester-longhorn already
exists, please reset it first before set data-disk-storage as default
```

If this happens, unset the default harvester-longhorn storage class using the following command, then re-apply the StorageClass:

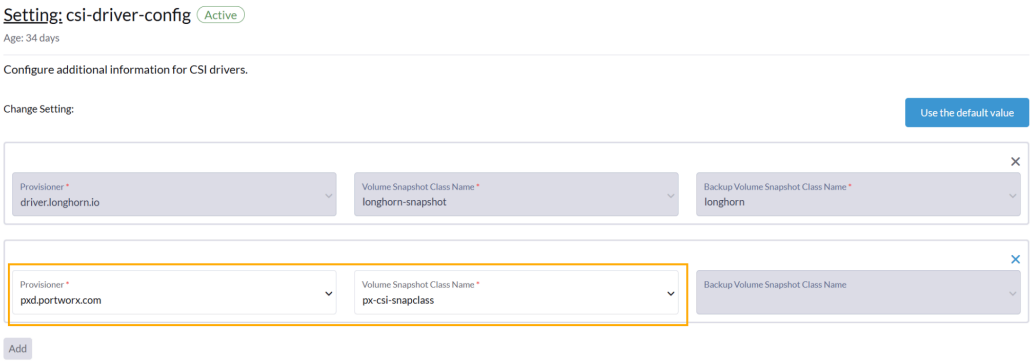
```
Shell
kubectl patch storageclass harvester-longhorn -p '{"metadata":
{"annotations":{"storageclass.kubernetes.io/is-default-class":"false"}}}'
```

Update the CSI Driver Configuration

1. Log in to the SUSE Virtualization UI.
2. Navigate to *Advanced* → *Settings*.
3. Find and select *csi-driver-config*.
4. Select ⋮ → *Edit Setting* to access the configuration options.
5. Set the *Provisioner* to the third-party CSI driver, ‘pxd.portworx.com’.
6. Configure the *Volume Snapshot Class Name* to ‘px-csi-snapclass’.
   This setting points to the name of the VolumeSnapshotClass used for creating

Copyright © 2025 SUSE

volume snapshots or VM snapshots.



Copyright © 2025 SUSE

Validation

Ensure that Portworx Enterprise is correctly integrated with SUSE Virtualization using the Pure Storage FlashArray by provisioning a virtual machine (VM) that uses a StorageClass backed by Portworx and Pure Storage.

Create a VM

1. Log in to the SUSE Virtualization UI.
2. Navigate to *Virtual Machines* and click *Create*.
   You may choose to create one or multiple VM instances.
3. Select a *namespace* for your VM.

   Note: Only the *harvester-public* namespace is visible to all users by default.
4. Provide a name for your VM.
5. (Optional) Select a *VM template* (iso-image, raw-image, windows-iso-image) to speed up provisioning.
6. On the *Basics* tab, configure the following settings:
   a. CPU: 2
      In practice, you may allocate up to 254 vCPUs, but best practice is not to exceed physical threads.
   b. Memory: 4 GB
   c. SSHKey: Select an existing key or upload a new one (required if you need SSH access).
   d. (Optional) Overcommit configuration: You may set CPU, memory, and storage overcommit ratios to allow scheduling of additional VMs even when physical resources are fully utilized.
   e. (Optional) Other advanced settings: These include run strategy, OS type, cloud-init data, etc.
7. On Volumes, select or add disks.
   a. On the Volumes tab, you will see one writable root disk created by default.
   b. Click *Add Disk* → choose *cd-rom* as the disk type.

Copyright © 2025 SUSE

c. Select the ISO image you uploaded and set its *boot order* to *1*.

d. Choose the StorageClass named data-disk-storage, which is provisioned by Portworx.

8. For the root disk, select the StorageClass (e.g., data-disk-storage) and specify the desired size.

9. The management network is added by default. You may remove it if using VLAN networks or, optionally, you can configure additional networks (VLAN, bridge, masquerade).

10. Review all settings and click *Create* to deploy the VM.

NOTE: Ensure that the data-disk-storage StorageClass was created during the installation process and is marked as the default or selected explicitly.

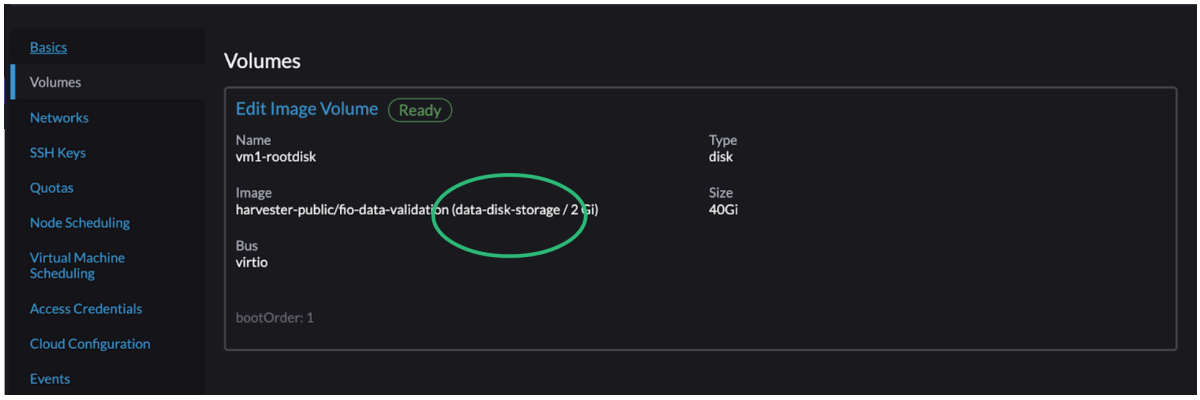
Verify the VM Storage

You can use the SUSE Virtualization UI or CLI to verify that the VM is backed by Portworx and Pure Storage.

Verify with the SUSE Virtualization UI

1. In the SUSE Virtualization UI, navigate to *Virtual Machines* → *<Your VM>* → *Volumes*.

2. Confirm that the attached disk is the 'data-disk-storage' StorageClass.



Copyright © 2025 SUSE

Verify with the SUSE Virtualization CLI

1. Log in to the SUSE Virtualization management CLI as root.
2. List the PersistentVolumeClaims (PVCs) in the VM's namespace.
    a. Confirm that the *STORAGECLASS* column shows 'data-disk-storage'.
    b. This indicates that the disk was provisioned using the Portworx storage class.

```
Shell
kubectl get pvc -n <namespace>

NAME          STATUS  VOLUME                                     CAPACITY
ACCESS MODES  STORAGECLASS       AGE
vm1-rootdisk  Bound   pvc-8f3c8b2c-6e1e-4c6a-a83e-9d6c34ac3b21  40Gi      RWX
data-disk-storage  2m
```

3. Confirm that the volume is Portworx-managed by describing the relevant PersistentVolume (PV).
    a. Check the *provisioned-by* annotation is 'pxd.portworx.com'.
    b. Verify that the *Driver* is 'pxd.portworx.com'.

```
Shell
kubectl describe pv pvc-8f3c8b2c-6e1e-4c6a-a83e-9d6c34ac3b21

Name:              pvc-8f3c8b2c-6e1e-4c6a-a83e-9d6c34ac3b21
StorageClass:  data-disk-storage
Annotations:
      pv.kubernetes.io/provisioned-by: pxd.portworx.com
Status: Bound
Reclaim Policy:    Delete
VolumeMode:      Block
Source:
    Type:              CSI (a Container Storage Interface (CSI) volume source)
    Driver:            pxd.portworx.com
    FSType:
    VolumeHandle:      424582431912071110
```


Copyright © 2025 SUSE

```
ReadOnly:          false
VolumeAttributes:      attached=ATTACH_STATE_EXTERNAL
                       error=
                       parent=
                       readonly=false
                       secure=false
                       shared=false
                       shared_mode=BLOCK
                       sharedv4=false
                       state=VOLUME_STATE_DETACHED
                       type=px-raw-volume
```


Copyright © 2025 SUSE

Conclusion

Organizations are modernizing their applications and infrastructure, transitioning from traditional, virtual machine deployments to modern, cloud-native technologies, such as containers and Kubernetes. SUSE Virtualization provides a flexible platform that enables container and virtual machine workloads to run side-by-side, on the same, unified infrastructure. Portworx Enterprise by Pure Storage brings resilient, scalable, and efficient storage services for mission-critical workloads, supporting enterprise availability, mobility, and data protection needs. Together, SUSE Virtualization and Portworx by Pure Storage present enterprises with a robust, flexible infrastructure platform to support their mission-critical, virtualized and cloud-native workloads.

This guide provides an overview of this unified, infrastructure platform along with detailed steps for configuring and validating the integration of SUSE Virtualization with Portworx Enterprise and a Pure Storage FlashArray to deliver a unified, infrastructure platform.

Continue your learning journey by exploring the following technical resources:

- SUSE Virtualization (Harvester) documentation
- Manage Portworx RWX Block Volumes on SUSE Virtualization
- Pure Storage FlashArray

To learn more about modernizing and optimizing your IT infrastructure landscape, contact SUSE at isv-cosell@suse.com.

Copyright © 2025 SUSE

Frequently Asked Questions

- Why is third-party storage support important for SUSE Virtualization?
    - Broad support for third-party storage provides customers with flexibility to leverage existing storage investments and to choose storage solutions that best meet business needs. Additionally, this helps customers avoid vendor lock-in and enables them to optimize storage infrastructure costs.
- Does Portworx require Pure Storage FlashArray?
    - No. Portworx is a Kubernetes-integrated storage solution that can work with a variety of hardware platforms, including third-party arrays and local SSD storage. However, Portworx does integrate with Pure Storage FlashArray to provide storage automation and resilience for containerized workloads with SUSE Virtualization.
- How can I expand the storage consumed by workloads within Portworx after installation?
    - Portworx enables automatic volume growth using AutopilotRule custom resources.
- What CPU architectures does Portworx support with SUSE Virtualization?
    - Portworx can run on Kubernetes nodes running x86 Linux hosts.

Copyright © 2025 SUSE