Université IBM i

19 et 20 novembre 2024

IBM Innovation Studio Paris

uii2024 #ibmi
#uii2024

S24 – Tutoriel Openshift : Prise en main opérationnelle pour les débutants

19 novembre 16:00 – 17:00

Thibaud Besson
Technical Sales Power - IBM France
thibaud.besson@fr.ibm.com

IBM

common
FRANCE

Université IBM i

19 et 20 novembre 2024

IBM i
continuous innovation
continuous integration

Pourquoi cette présentation ?

- Changement de paradigme : OpenShift change le paradigme de la virtualisation : déploiement d’applications, pas de VM.
- Kubernetes a été fait pour des développeurs. Un admin système a besoin de connaître des notions nouvelles.
- L’interface est déroutante pour un admin système. Elle paraît simple, mais n’est pas intuitive si on ne connaît pas les principes sous-jacents. En ligne de commande, l’utilisateur n’est plus accompagné.
- Si tout va bien, tout va bien… Mais si problème, le debug devient très complexe en s’appuyant uniquement sur la documentation, car en général elle ne prévoit pas que tout ne se passe pas bien.

- La documentation n’accompagne pas assez
  - 💣 Pas assez orientée usage pour un débutant
  - 💣 Très dense
  - 💣 Fait des hypothèses sur les compétences du lecteur
  - 💣 Ne fait pas de priorité sur la complexité, la fréquence d’utilisation, etc.
  - 💣 Ne répète pas les informations importantes
  - 💣 Explique en général comment, mais pas assez pourquoi

Besoin d’une documentation orientée vers les débutants

La documentation OpenShift

https://docs.redhat.com/en/products

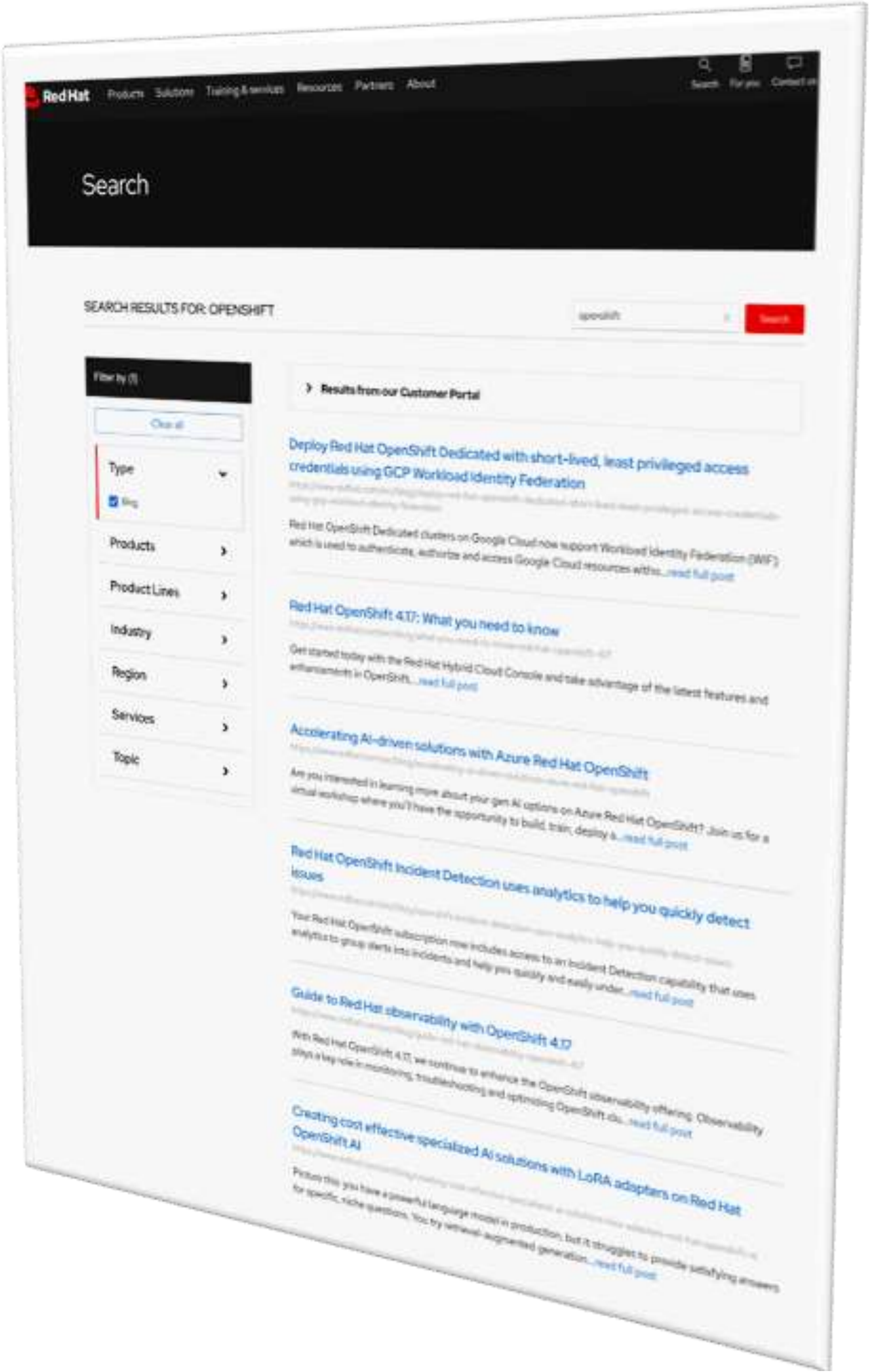
Red Hat Documentation

All product documentation

Explore and find documentation for all Red Hat products

Les blogs Red Hat

- https://www.redhat.com/en/search?search=openshift&f[0]=hybrid_type:Blog
- Plus faciles à lire
- Utile et intéressant : donne une bonne introduction à des fonctions d’OpenShift.



SEARCH RESULTS FOR: OPENSHIFT

Results from our Customer Portal

Deploy Red Hat OpenShift Dedicated with short-lived, least privileged access credentials using GCP Workload Identity Federation

Red Hat OpenShift Dedicated clusters on Google Cloud now support Workload Identity Federation (WIF) which is used to authenticate, authorize and access Google Cloud resources withi... read full post

Red Hat OpenShift 4.17: What you need to know

Get started today with the Red Hat Hybrid Cloud Console and take advantage of the latest features and enhancements in OpenShift... read full post

Accelerating AI-driven solutions with Azure Red Hat OpenShift

Are you interested in learning more about your gen AI options on Azure Red Hat OpenShift? Join us for a virtual workshop where you'll have the opportunity to build, train, deploy a... read full post

Red Hat OpenShift Incident Detection uses analytics to help you quickly detect issues

Your Red Hat OpenShift subscription now includes access to an Incident Detection capability that uses analytics to group alerts into incidents and help you quickly and easily under... read full post

Guide to Red Hat observability with OpenShift 4.17

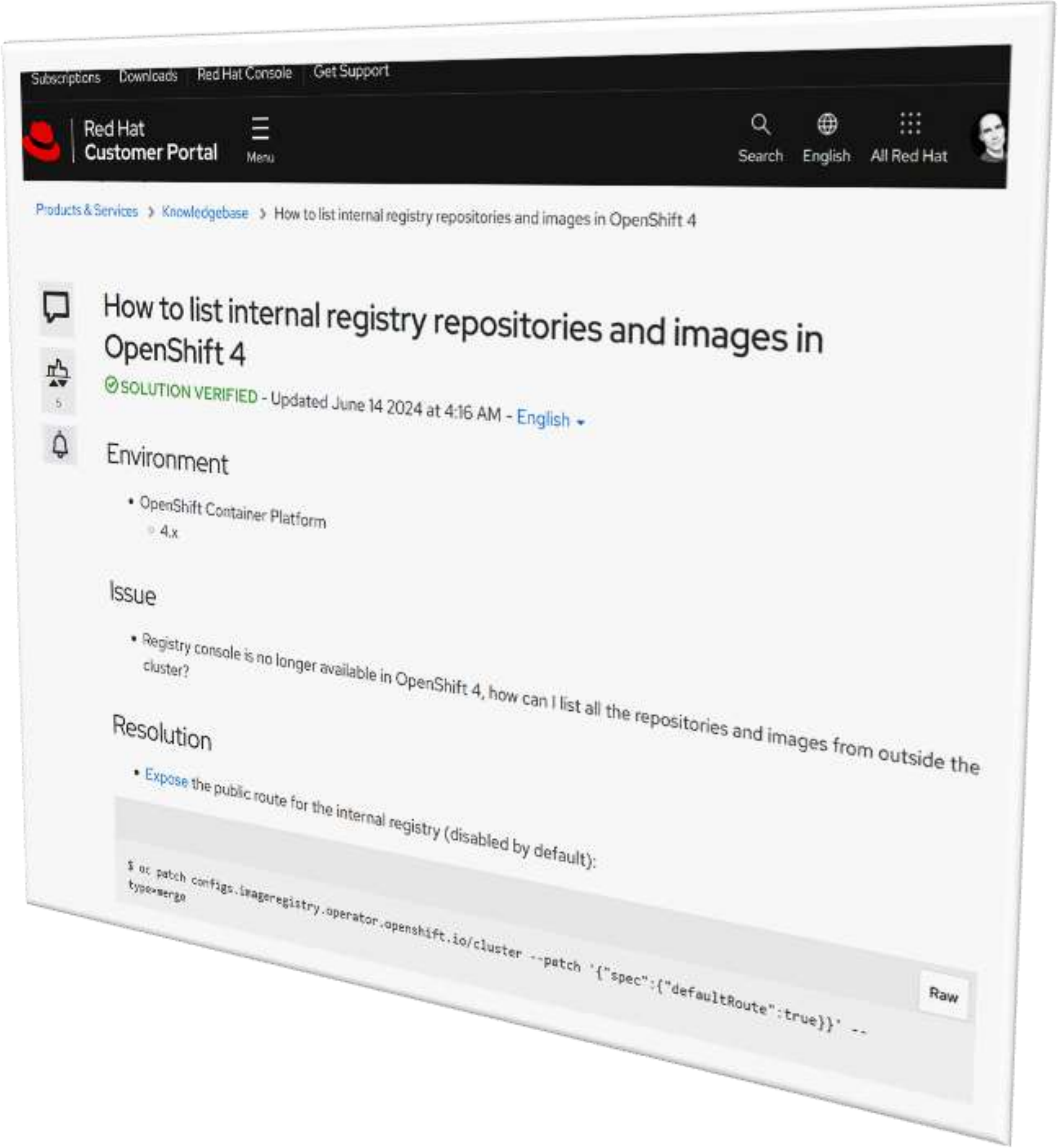
With Red Hat OpenShift 4.17, we continue to enhance the OpenShift observability offering. Observability plays a key role in monitoring, troubleshooting and optimizing OpenShift clu... read full post

Creating cost effective specialized AI solutions with LoRA adapters on Red Hat OpenShift AI

Picture this: you have a powerful language model in production, but it struggles to provide satisfying answers for specific, niche questions. You try retrieval-augmented generation... read full post

Red Hat Knowledgebase

- Nécessite un identifiant Red Hat
- Est indexé par Google
- Très utile !
- Donne des réponses :
    - Spécifiques
    - Cas pratiques
    - Difficultés rencontrées par des utilisateurs
    - Corrections d’erreurs
    - Pas (encore) dans la documentation officielle
    - Autorise les commentaires en bas de page



How to list internal registry repositories and images in OpenShift 4

SOLUTION VERIFIED - Updated June 14 2024 at 4:16 AM - English

Environment

- OpenShift Container Platform
    - 4.x

Issue

- Registry console is no longer available in OpenShift 4, how can I list all the repositories and images from outside the cluster?

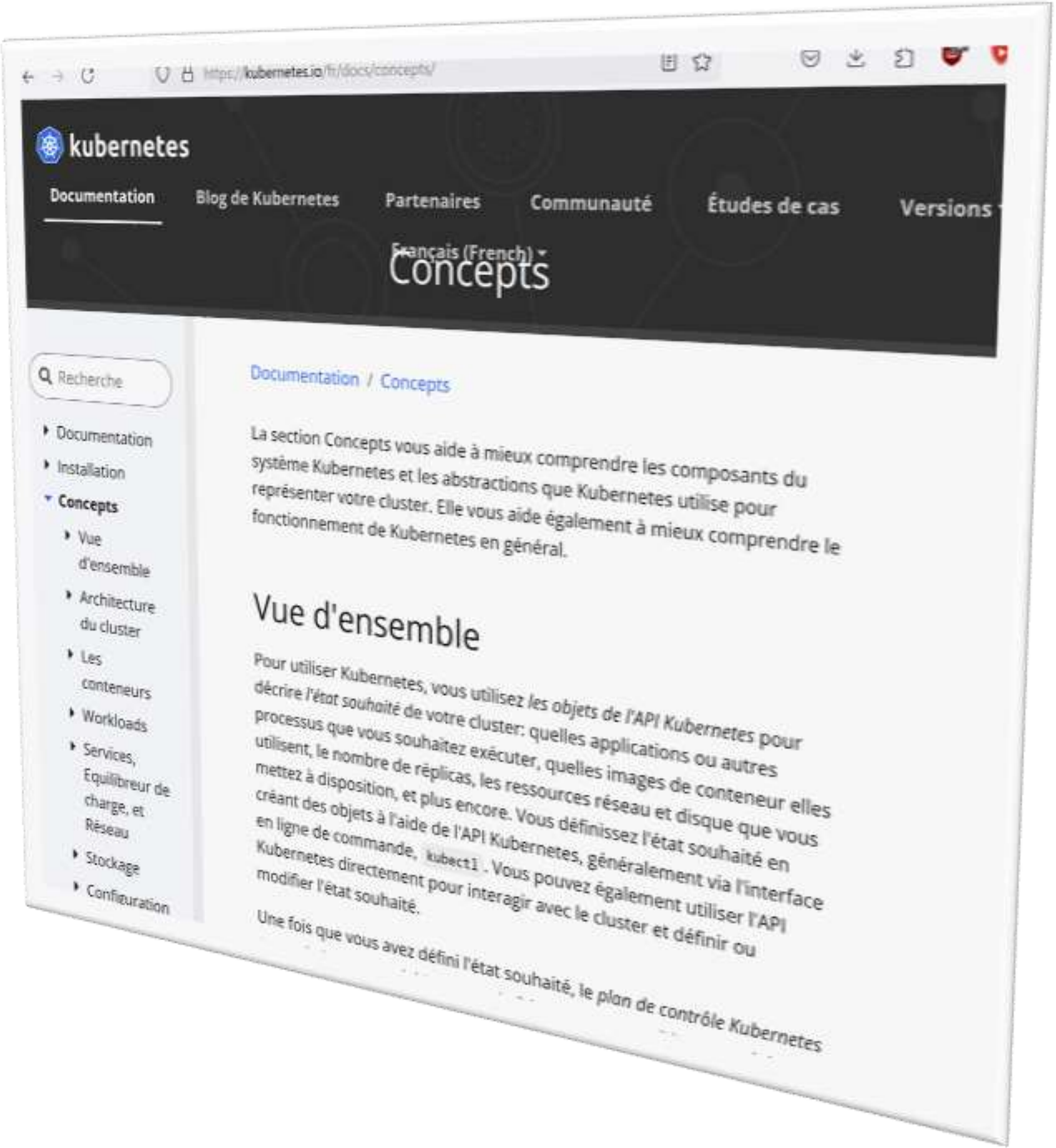
Resolution

- Expose the public route for the internal registry (disabled by default):

```
$ oc patch configs.imageregistry.operator.openshift.io/cluster --patch '{"spec":{"defaultRoute":true}}' --
type=merge
```

Documentation kubernetes

- Utile en parallèle de la documentation OpenShift
- Explique bien les concepts de base



Concepts

Documentation / Concepts

La section Concepts vous aide à mieux comprendre les composants du système Kubernetes et les abstractions que Kubernetes utilise pour représenter votre cluster. Elle vous aide également à mieux comprendre le fonctionnement de Kubernetes en général.

Vue d'ensemble

Pour utiliser Kubernetes, vous utilisez *les objets de l'API Kubernetes* pour décrire *l'état souhaité* de votre cluster: quelles applications ou autres processus que vous souhaitez exécuter, quelles images de conteneur elles utilisent, le nombre de réplicas, les ressources réseau et disque que vous mettez à disposition, et plus encore. Vous définissez l'état souhaité en créant des objets à l'aide de l'API Kubernetes, généralement via l'interface en ligne de commande, `kubectl` . Vous pouvez également utiliser l'API Kubernetes directement pour interagir avec le cluster et définir ou modifier l'état souhaité.

Une fois que vous avez défini l'état souhaité, le *plan de contrôle Kubernetes*

Valeur d’OpenShift

Kubernetes, c’est difficile !

Openshift rend Kubernetes plus simple, plus fiable, plus sûr

INSTALLER

- Templating
- Validating
- OS setup

DÉPLOYER

- Identity & security access
- App monitoring & alerts
- Storage & persistence
- Egress, ingress, & integration
- Host container images
- Build/Deploy methodology

SÉCURISER

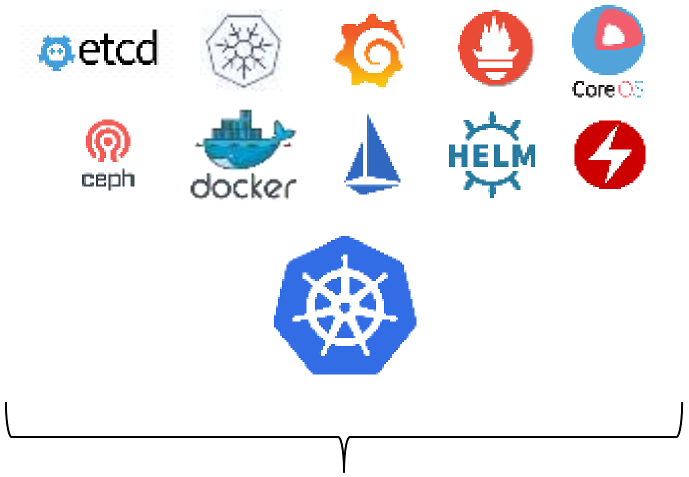
- Platform monitoring & alerts
- Metering & chargeback
- Platform security hardening
- Image hardening
- Security certifications
- Network policy
- Disaster recovery
- Resource segmentation

OPÉRER

- OS upgrade & patch
- Platform upgrade & patch
- Image upgrade & patch
- App upgrade & patch
- Security patches
- Continuous security scanning
- Multi-environment rollout
- Enterprise container registry
- Cluster & app elasticity
- Monitor, alert, remediate (Prometheus)
- Log aggregation

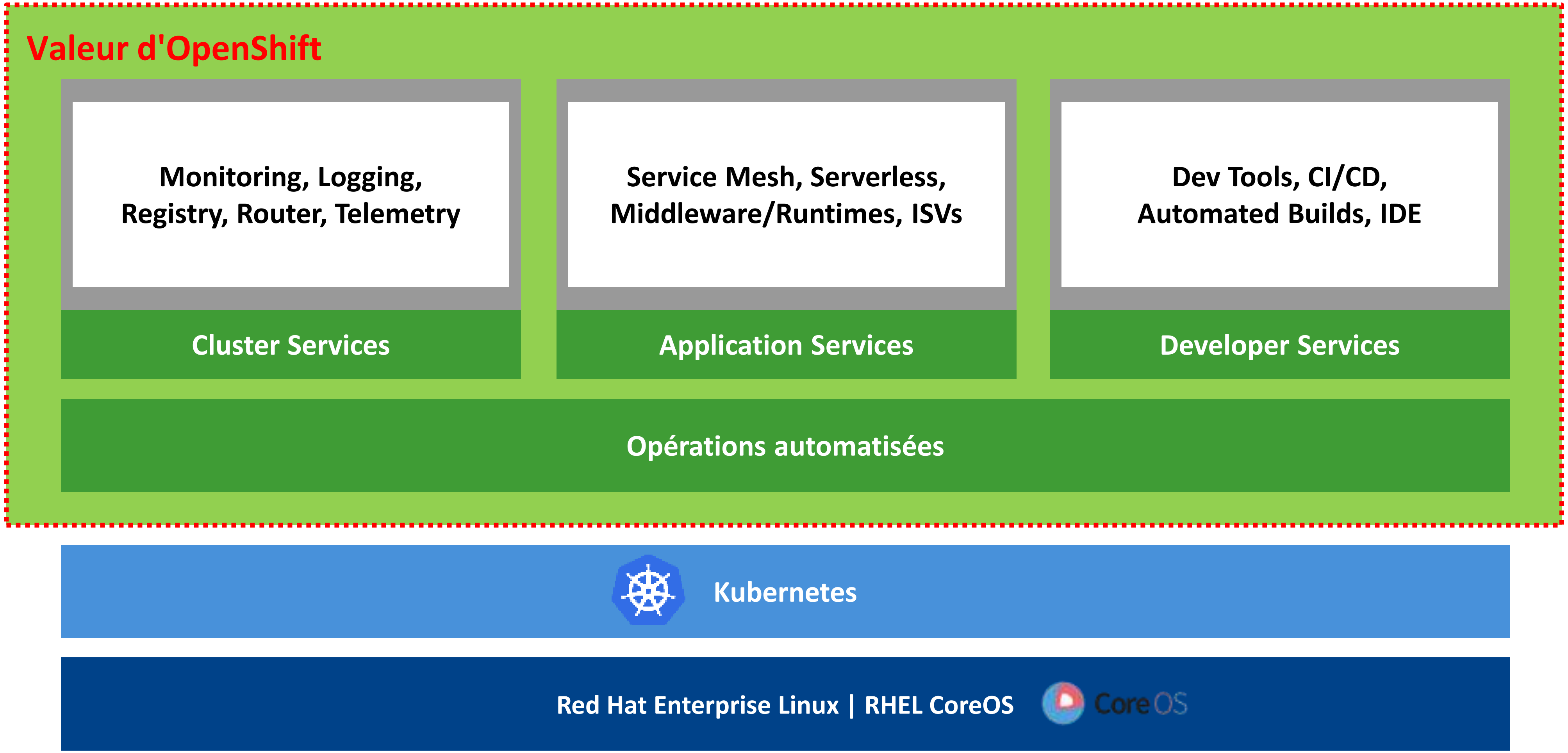
OpenShift est un Kubernetes d’entreprise fiable et sécurisé

- Des centaines de correctifs de défauts et de performances
- + de 200 intégrations validées
- Écosystème de conteneurs certifiés
- Gestion du cycle de vie d’OpenShift sur 9 ans
- Red Hat est l’un des principaux contributeurs Kubernetes depuis le premier jour



Version d’OpenShift

OpenShift : Aperçu des fonctions



Valeur d'OpenShift

Monitoring, Logging, Registry, Router, Telemetry

Cluster Services

Service Mesh, Serverless, Middleware/Runtimes, ISVs

Application Services

Dev Tools, CI/CD, Automated Builds, IDE

Developer Services

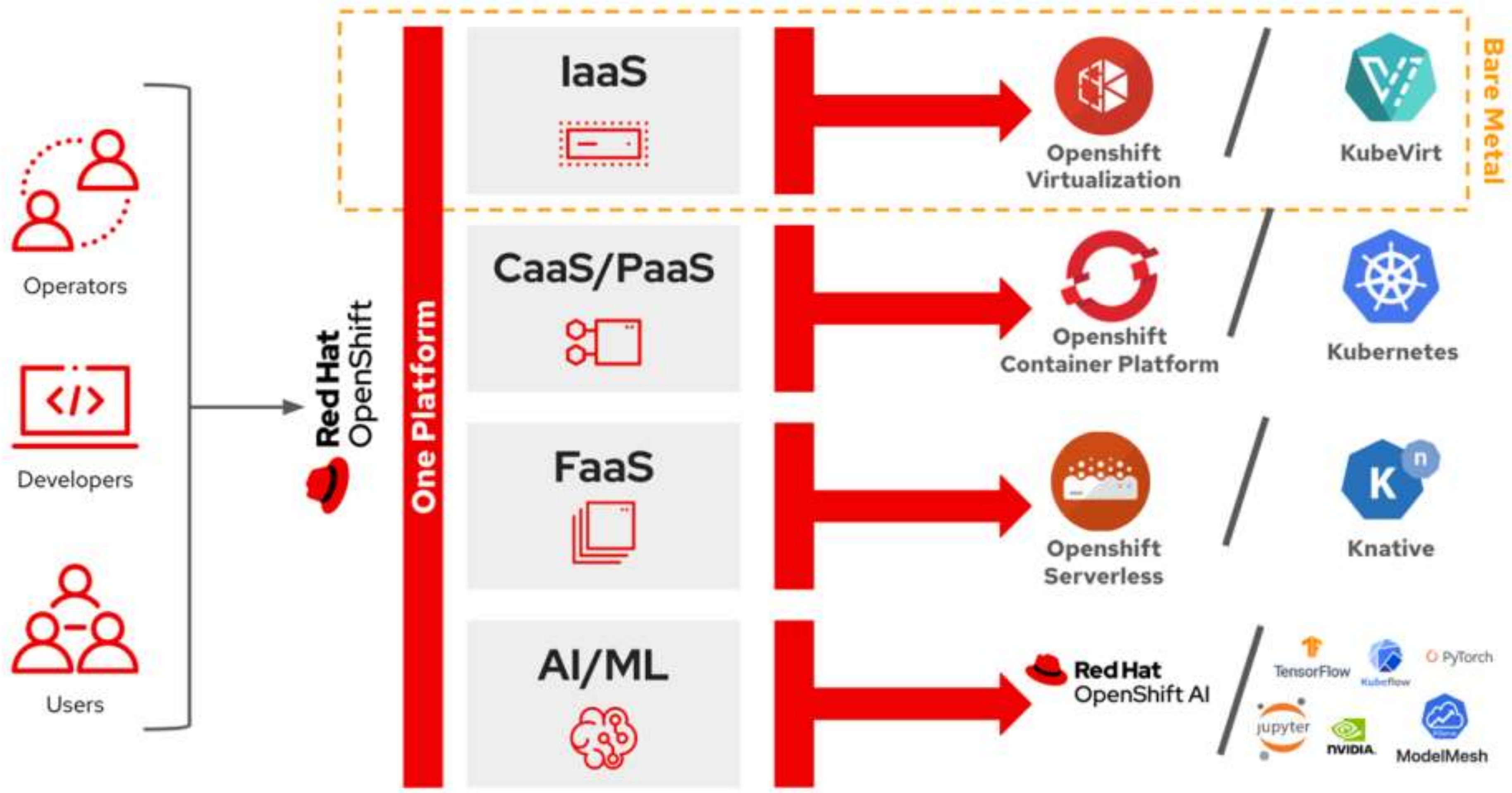
Opérations automatisées

Kubernetes

Red Hat Enterprise Linux | RHEL CoreOS

Best IT Ops Experience CaaS ⟷ PaaS ⟷ FaaS Best Developer Experience

OpenShift : Une plateforme unique pour différents usages



Université IBM i

19 et 20 novembre 2024

IBM i
continuous innovation
continuous integration

Concepts indispensables pour démarrer

Container ?

un conteneur Linux® est un processus ou un ensemble de processus isolés du reste du système linux. La technologie de containerisation utilise le noyau Linux et ses fonctions

- les control groups ou Cgroups
- les espaces de noms namespaces

pour séparer des processus afin qu'ils s'exécutent de manière indépendante.

Cette indépendance reflète l'objectif des conteneurs : exécuter plusieurs processus et applications séparément les uns des autres afin d'optimiser l'utilisation de votre infrastructure tout en bénéficiant du même niveau de sécurité que celui des systèmes distincts.

Tous les fichiers nécessaires à leur exécution sont fournis par une **image** distincte, ce qui signifie que les conteneurs Linux sont portables et fonctionnent de **la même manière** dans les environnements de développement, de test et de production. Ainsi, ils sont bien plus rapides à utiliser que les pipelines de développement qui s'appuient sur la réplication d'environnements de test traditionnels.

- User namespaces allow per-namespace mappings of user and group IDs. In the context of containers, this means that **users and groups may have privileges for certain operations inside the container without having those privileges outside the container.** (In other words, a process's set of capabilities for operations inside a user namespace can be quite different from its set of capabilities in the host system.) One of the specific goals of user namespaces is to allow a process to have root privileges for operations inside the container, while at the same time being a normal unprivileged process on the wider system hosting the container.
- By using cgroups, system administrators gain fine-grained control over allocating, prioritizing, denying, managing, and monitoring system resources.

CoreOS – le système d’exploitation dédié aux containers

- CoreOS est un système d’exploitation focalisé sur l’hébergement de containers, Kubernetes et OpenShift
- Deux versions de CoreOS :
  - Le projet upstream Fedora CoreOS open source et libre d’usage, installable indépendemment de OpenShift.
  - Red Hat Enterprise Linux CoreOS (RHCOS) est le produit supporté par Red Hat en tant que composant de OpenShift Container Platform (OCP).
- Socle monolithique, minimal, mis à jour automatiquement de manière atomique

Ce qui va vous surprendre :

- Système de fichier en layers
- Configuration par ignition file
- Modification atomique de la configuration

On ne peut pas le traiter comme un système d’exploitation ordinaire.

https://www.redhat.com/en/blog/red-hat-enterprise-linux-coreos-customization
https://developers.redhat.com/blog/2020/03/10/how-to-run-containerized-workloads-securely-and-at-scale-with-fedora-coreos#fedora_coreos

CoreOS pour OpenShift

- Installer OpenShift implique l’installation de CoreOS sur les nodes du cluster
- CoreOS n’est pas installé séparément au préalable, il fait partie de l’installation d’OpenShift
- Les principes fondamentaux de CoreOS apparaissent pendant l’installation et l’utilisation d’OpenShift
- Configuration initiale pendant l’installation de OpenShift par ignition file
- Création d’un utilisateur par défaut `core`
- Login par clef SSH, pas de mot de passe
- Configuration de CoreOS à travers MachineConfig de OpenShift, ne pas modifier directement CoreOS

CoreOS est géré de manière atomique ?

- an atomically-managed system that applies all changes (upgrades, new packages, etc.) in a single atomic operation layered on top of the base file system. This practice produces systems that are more predictable and reliable.

- https://www.redhat.com/en/blog/red-hat-enterprise-linux-coreos-customization

CRI-O : container runtime d’OpenShift

Kubernetes supporte 3 runtimes pour les containers :

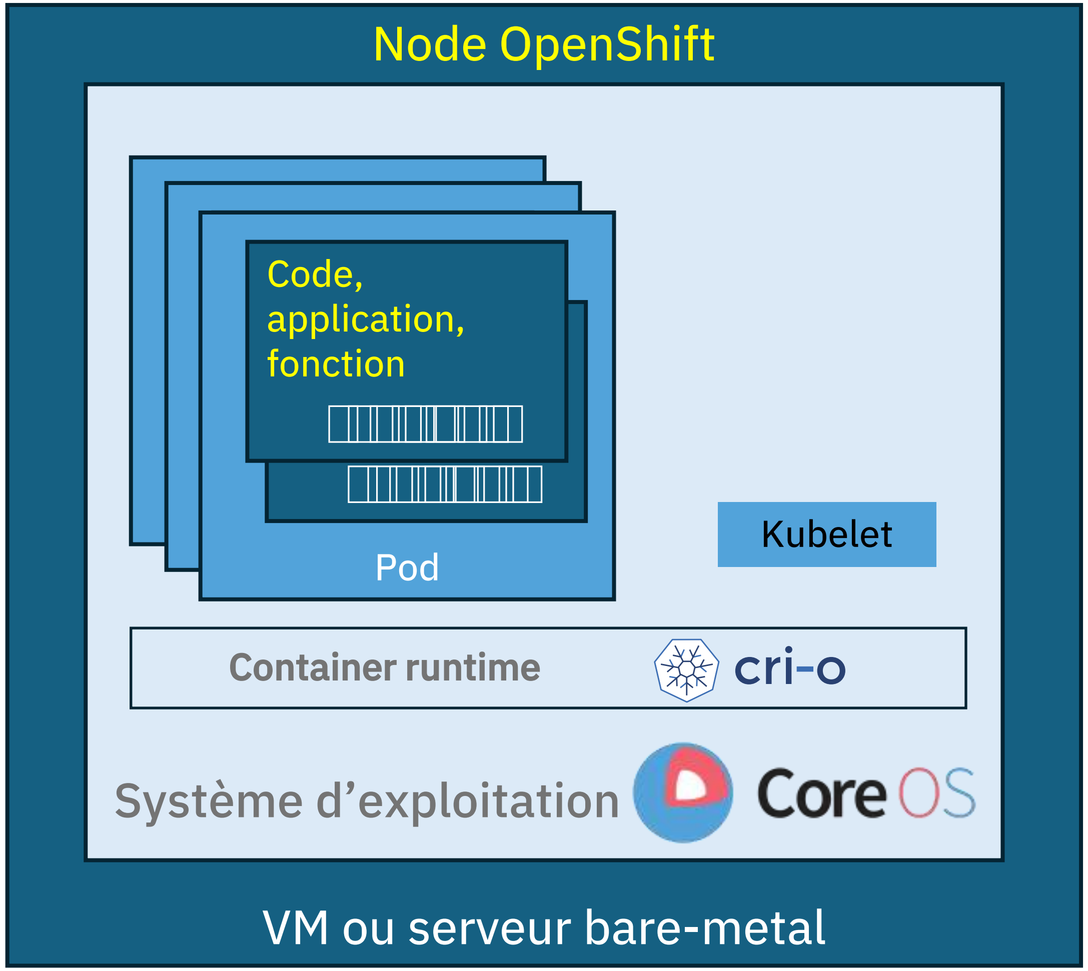
- containerd
- cri-o
- docker

- Red Hat a choisi CRI-O comme runtime dans CoreOS.
- `crictl` est la commande pour interagir avec les containers dans CoreOS des nodes du cluster.

```
crictl pods
critctl ps
critctl images
crictl exec -i -t 1f73f2d81bf98 ls
```

https://kubernetes.io/docs/tasks/debug/debug-cluster/crictl/

Les unités de base d’OpenShift



Vus depuis l’infrastructure :

- Des containers,
- dans des pods,
- dans des systèmes d’exploitation CoreOS,
- dans des VMs ou des machines physiques

Kubelet ?

Dans CoreOS des nodes du cluster OpenShift (et dans kubernetes), un composant n'est pas dans un pod : le Kubelet.

Le Kubelet fait le lien entre le Control Plane du cluster et le node qui l'héberge.

C'est un service géré par `systemd`, chargé de :

- Gérer le node
- Gérer les pods
- Gérer les ressources (CPU, mémoire, stockage, etc)
- Garantir la santé du cluster

- Important pour le debug ! `journalctl -u kubelet`

Université IBM i

19 et 20 novembre 2024

IBM i

continuous innovation
continuous integration

IBM

Architectures de déploiement d'OpenShift

Architecture classique d’un cluster OpenShift

Les nodes sont les serveurs physiques ou les VM portant les instances CoreOS

- ~~Master~~ Control plane nodes
- Worker nodes

Les pods se regroupent en deux types :

- Control plane
- Data plane

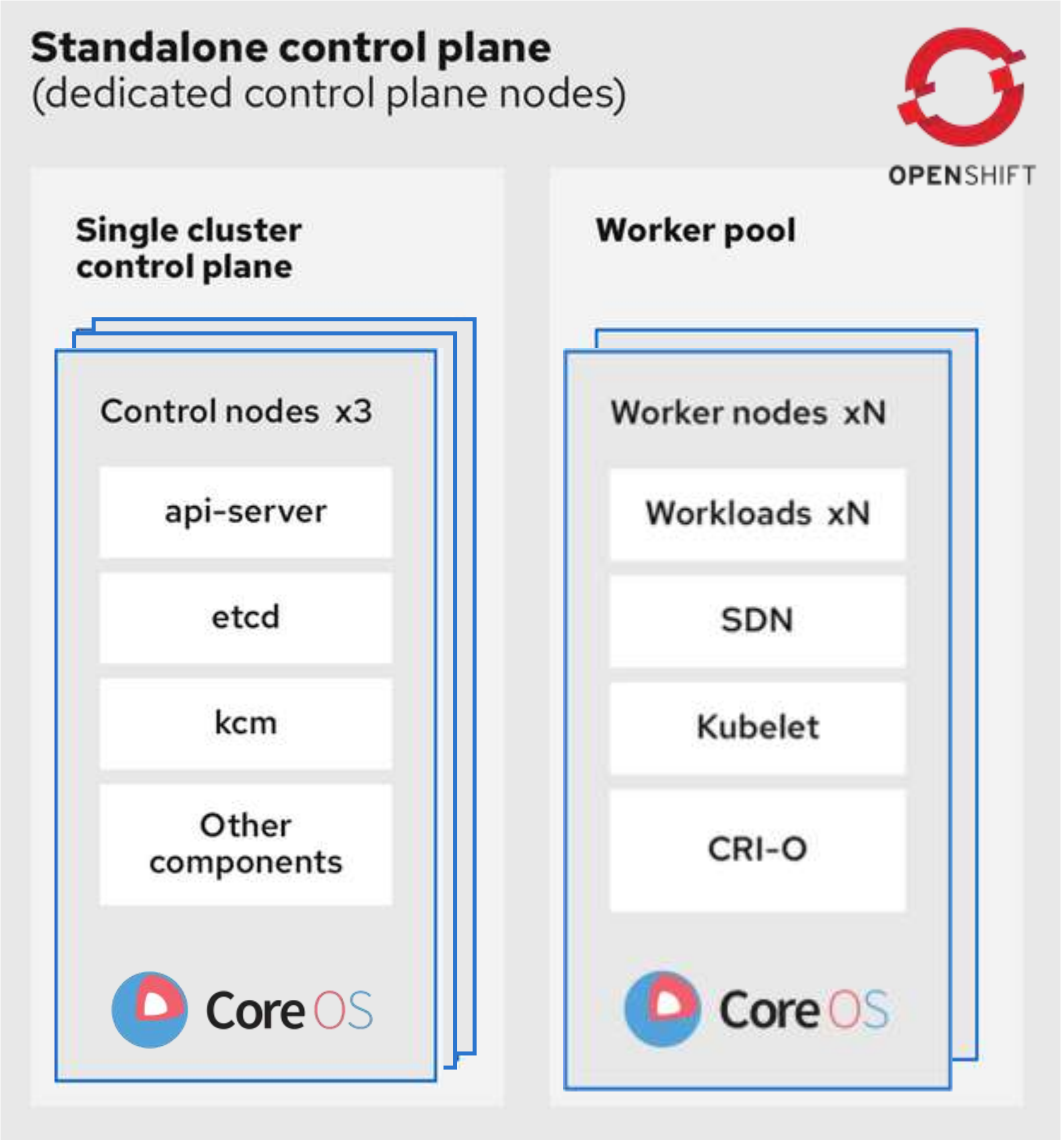
Le control plane et le data plane peuvent être :

- physiquement séparés sur plusieurs machines,
- Virtuellement séparés dans plusieurs VMs,
- regroupés sur un seul environnement, physique dans un serveur ou virtuel dans une VM.

Le control plane permettant au cluster d’opérer sont déployées dans CoreOS sous forme de containers dans des pods

- kubernetes controller manager (KCM)
- API server
- etcd

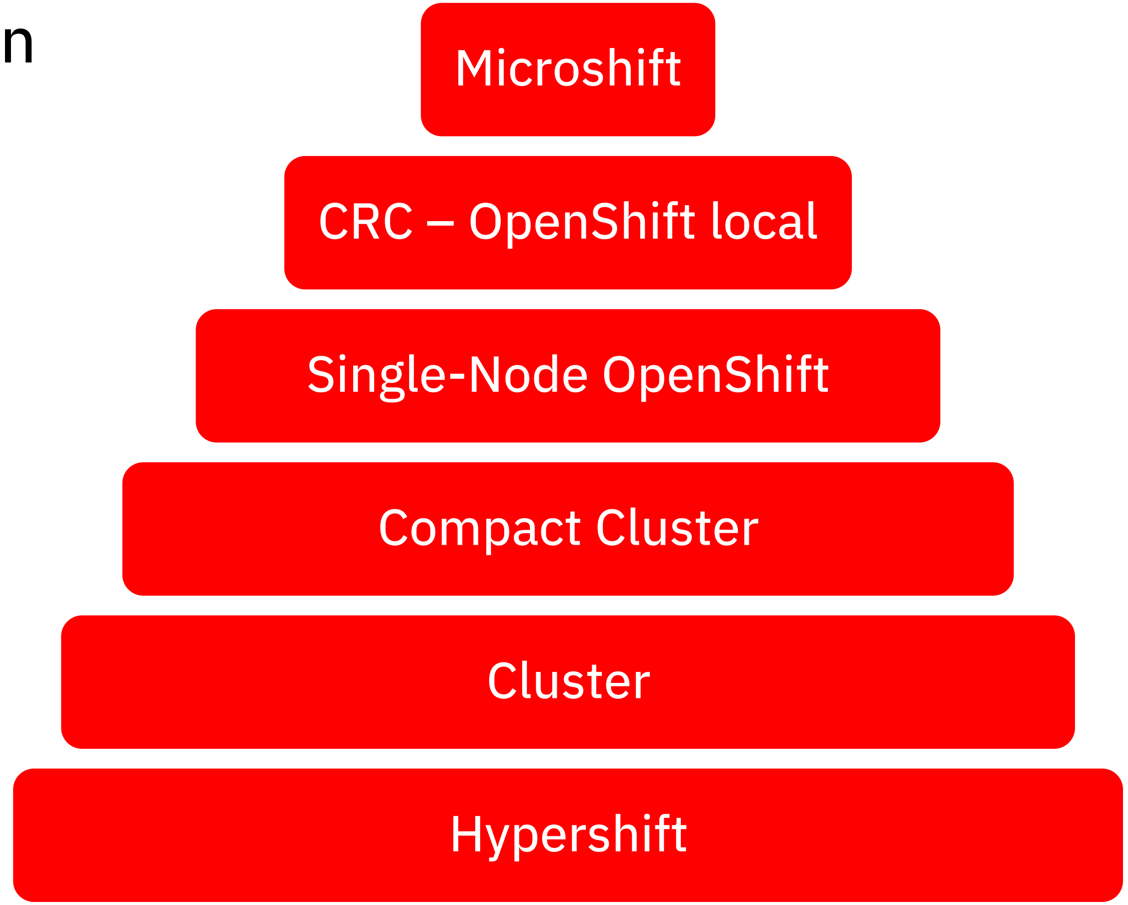
Sauf pour le Kubelet qui n’est pas un pod mais un agent déployé sous forme de service dans CoreOS des worker nodes.



Architectures du cluster OpenShift

Plusieurs architectures sont possibles selon

- Les ressources matérielles disponibles : CPU, mémoire, stockage
- Les besoins de résilience : dev, prod
- L'environnement qui accueille OpenShift : on premise, cloud, edge



Microshift

CRC – OpenShift local

Single-Node OpenShift

Compact Cluster

Cluster

Hypershift

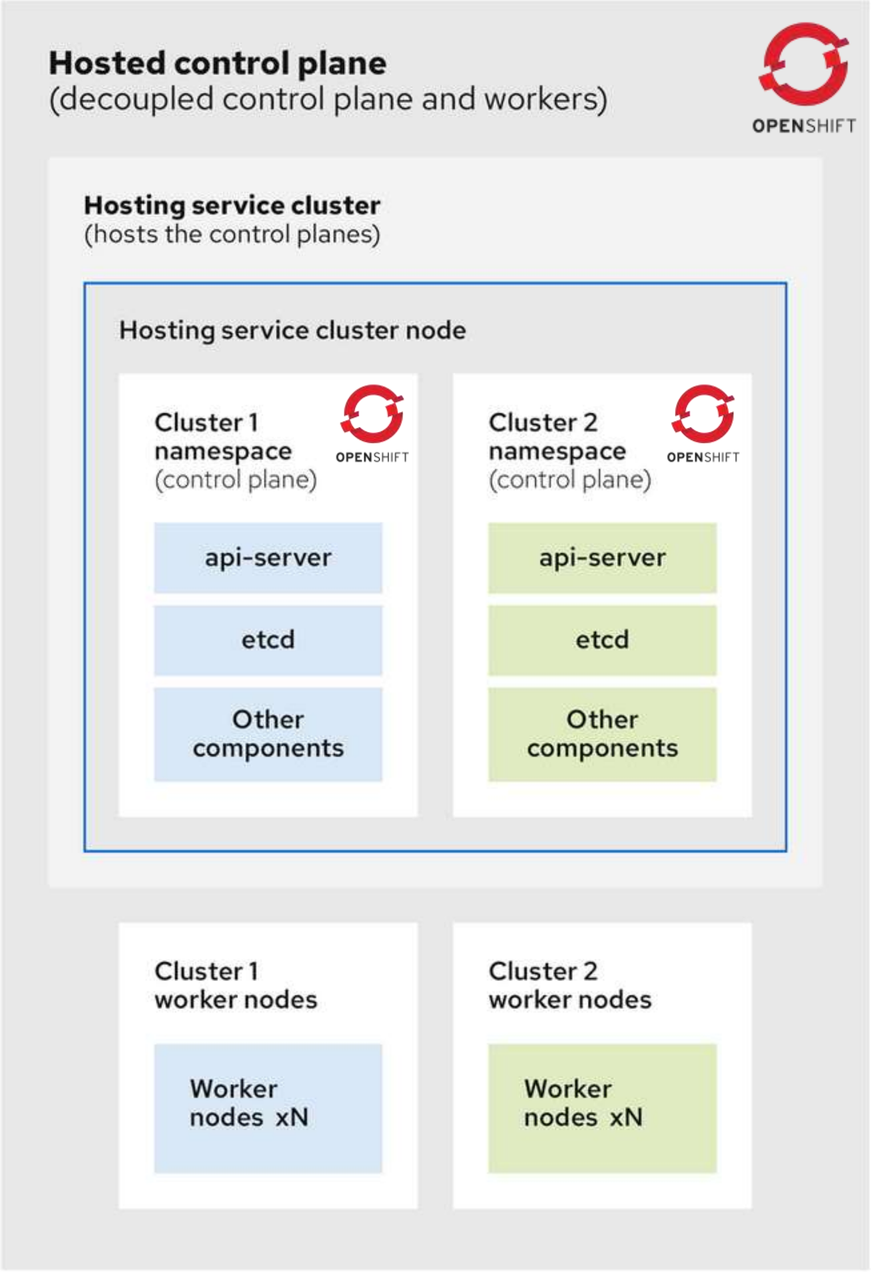
Différentes architectures selon les besoins

- **MicroShift** : Pour le Edge computing. Un cluster Kubernetes capable de fonctionner depuis une seule machine comprenant seulement deux cœurs de processeurs et 2 Go de RAM. https://docs.redhat.com/en/documentation/red_hat_build_of_microshift/4.17
- **OpenShift Local** alias CRC Code-Ready Containers : Pour les développeurs. Utiliser OpenShift sur un ordinateur de bureau sous Linux, macOS, ou Windows. https://docs.redhat.com/en/documentation/red_hat_openshift_local/2.43
- **Single Node OpenShift** SNO : tout OpenShift dans une seule machine (virtuelle ou pas). Le plus compact des déploiement OpenShift, sans redondance.

Sans redondance

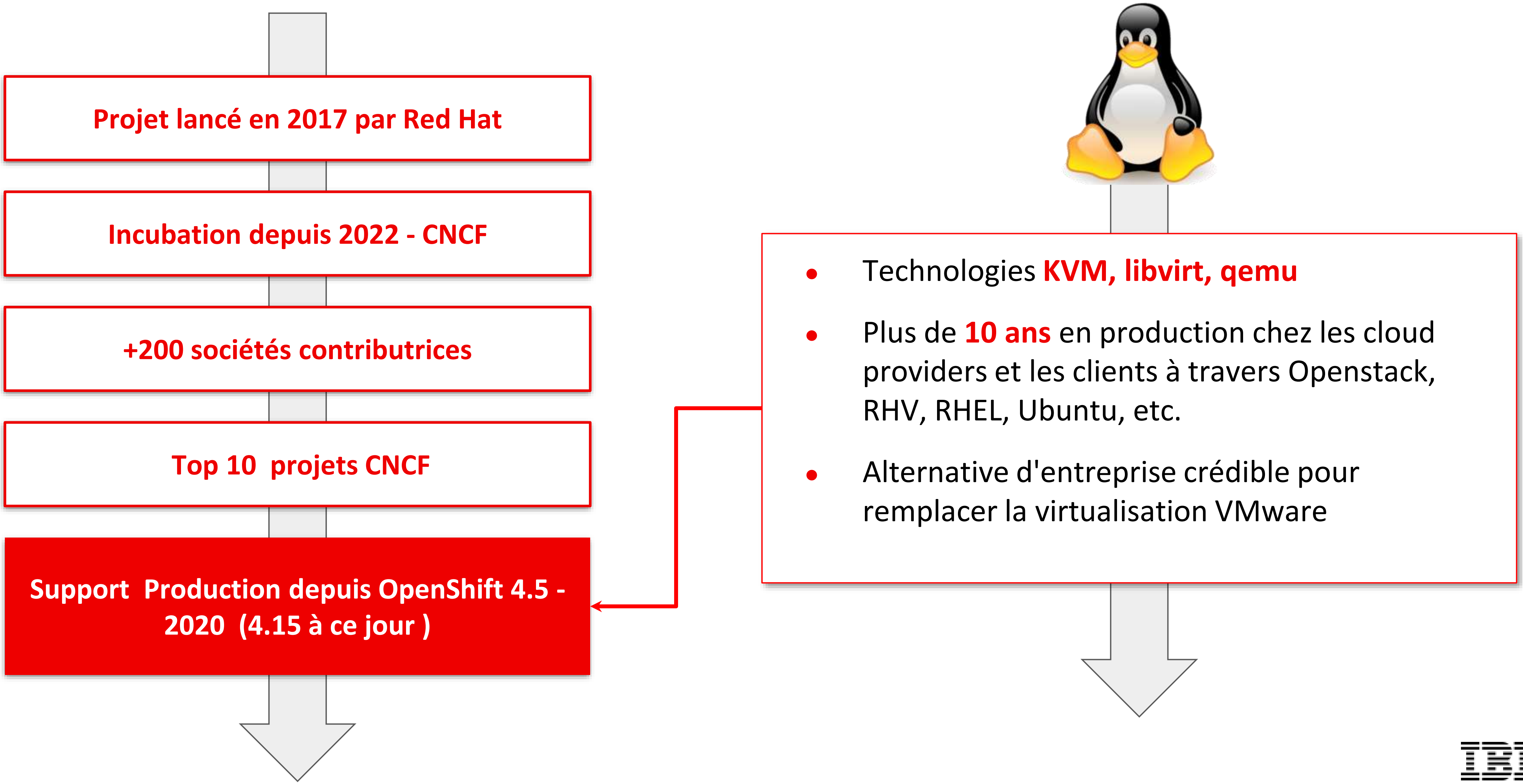
- **Compact Cluster** : cluster redondé mais control plane et data plane sur les mêmes 3 machines.
- **Cluster standard** : 3 masters, <n> workers.
- **HyperShift** alias **Hosted control plane** : OpenShift dans OpenShift. Le control plane est déployé par un cluster OpenShift.
- **Multi-Architecture Cluster** alias MAC : Un unique control plane déploie des applications sur des workers de différentes architectures matérielles x86, Power, Z.

Avec redondance



KubeVirt ⇨ OpenShift Virtualization

CLOUD NATIVE COMPUTING FOUNDATION



Projet lancé en 2017 par Red Hat

Incubation depuis 2022 - CNCF

+200 sociétés contributrices

Top 10 projets CNCF

Support Production depuis OpenShift 4.5 - 2020 (4.15 à ce jour)

- Technologies **KVM, libvirt, qemu**
- Plus de **10 ans** en production chez les cloud providers et les clients à travers Openstack, RHV, RHEL, Ubuntu, etc.
- Alternative d'entreprise crédible pour remplacer la virtualisation VMware

IBM

Université IBM i

19 et 20 novembre
2024

IBM i
continuous innovation
continuous integration

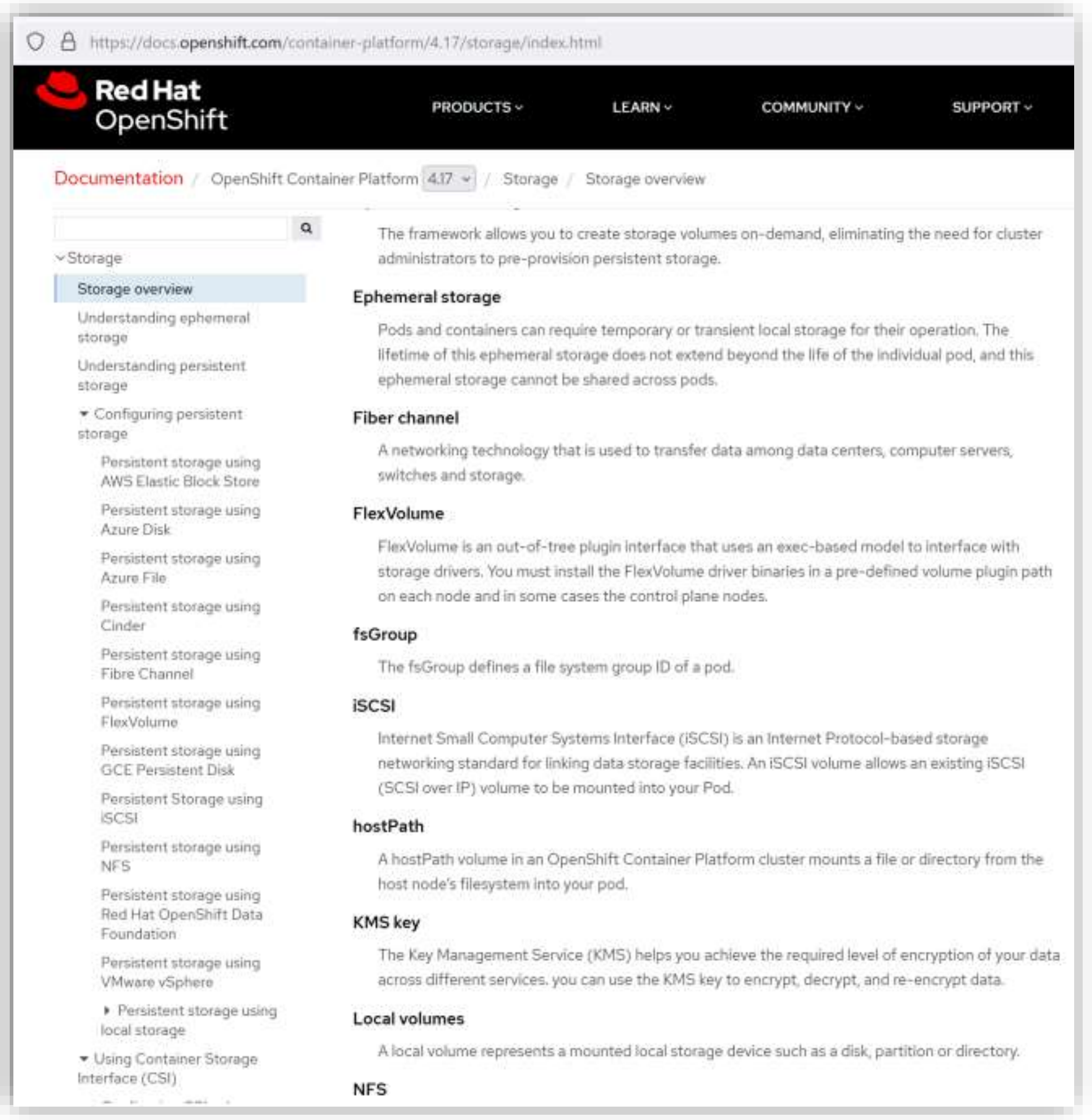
Généralités sur le stockage dans OpenShift

Stockage pour Openshift

Pendant longtemps le stockage a été négligé dans Kubernetes : Kubernetes a été conçu à l’origine pour des charge de travail « stateless », qui n’ont pas besoin de stockage persistant.

Avec l’arrivée des applications « stateful » par ex. bases de données dans Kubernetes, le stockage persistant et partagé sur nodes du cluster devient indispensable.

Les possibilités sont très nombreuses selon la source, le type du stockage, l’environnement du déploiement, etc.



https://docs.openshift.com/container-platform/4.17/storage/index.html

Documentation / OpenShift Container Platform 4.17 / Storage / Storage overview

- Storage
  - Storage overview
  - Understanding ephemeral storage
  - Understanding persistent storage
  - Configuring persistent storage
    - Persistent storage using AWS Elastic Block Store
    - Persistent storage using Azure Disk
    - Persistent storage using Azure File
    - Persistent storage using Cinder
    - Persistent storage using Fibre Channel
    - Persistent storage using FlexVolume
    - Persistent storage using GCE Persistent Disk
    - Persistent Storage using iSCSI
    - Persistent storage using NFS
    - Persistent storage using Red Hat OpenShift Data Foundation
    - Persistent storage using VMware vSphere
    - Persistent storage using local storage
  - Using Container Storage Interface (CSI)

The framework allows you to create storage volumes on-demand, eliminating the need for cluster administrators to pre-provision persistent storage.

Ephemeral storage

Pods and containers can require temporary or transient local storage for their operation. The lifetime of this ephemeral storage does not extend beyond the life of the individual pod, and this ephemeral storage cannot be shared across pods.

Fiber channel

A networking technology that is used to transfer data among data centers, computer servers, switches and storage.

FlexVolume

FlexVolume is an out-of-tree plugin interface that uses an exec-based model to interface with storage drivers. You must install the FlexVolume driver binaries in a pre-defined volume plugin path on each node and in some cases the control plane nodes.

fsGroup

The fsGroup defines a file system group ID of a pod.

iSCSI

Internet Small Computer Systems Interface (iSCSI) is an Internet Protocol-based storage networking standard for linking data storage facilities. An iSCSI volume allows an existing iSCSI (SCSI over IP) volume to be mounted into your Pod.

hostPath

A hostPath volume in an OpenShift Container Platform cluster mounts a file or directory from the host node's filesystem into your pod.

KMS key

The Key Management Service (KMS) helps you achieve the required level of encryption of your data across different services. you can use the KMS key to encrypt, decrypt, and re-encrypt data.

Local volumes

A local volume represents a mounted local storage device such as a disk, partition or directory.

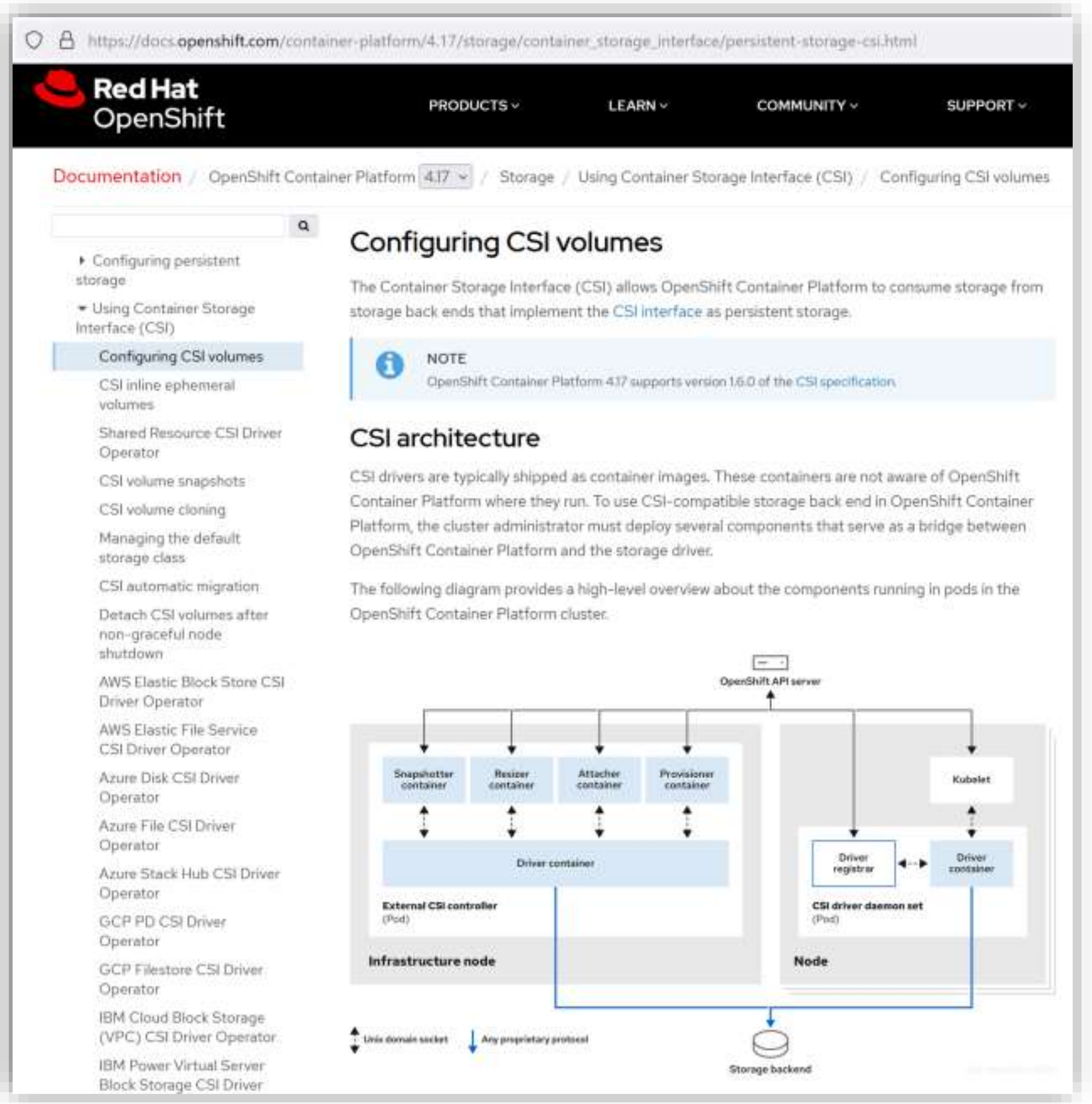
NFS

CSI storage

le Container Storage Interface (CSI) est un standard qui permet aux applications dans OpenShift de consommer du stockage sur du stockage externe, donc des baies de stockage. Il rend accessible de l'espace de stockage bloc et fichier dans Kubernetes / OpenShift.

Le Driver CSI est le composant conçu par le fournisseur de stockage pour s'interfacer avec Kubernetes.

A driver is a ***software component*** *that acts as a translator between a computer's hardware and its operating system. Drivers provide an* ***abstraction layer*** *that allows programmers to write software without needing to know the intricate details of specific hardware.*



https://docs.openshift.com/container-platform/4.17/storage/container_storage_interface/persistent-storage-csi.html

Red Hat OpenShift

PRODUCTS LEARN COMMUNITY SUPPORT

Documentation / OpenShift Container Platform 4.17 / Storage / Using Container Storage Interface (CSI) / Configuring CSI volumes

- Configuring persistent storage
- Using Container Storage Interface (CSI)
  - Configuring CSI volumes
  - CSI inline ephemeral volumes
  - Shared Resource CSI Driver Operator
  - CSI volume snapshots
  - CSI volume cloning
  - Managing the default storage class
  - CSI automatic migration
  - Detach CSI volumes after non-graceful node shutdown
  - AWS Elastic Block Store CSI Driver Operator
  - AWS Elastic File Service CSI Driver Operator
  - Azure Disk CSI Driver Operator
  - Azure File CSI Driver Operator
  - Azure Stack Hub CSI Driver Operator
  - GCP PD CSI Driver Operator
  - GCP Filestore CSI Driver Operator
  - IBM Cloud Block Storage (VPC) CSI Driver Operator
  - IBM Power Virtual Server Block Storage CSI Driver

Configuring CSI volumes

The Container Storage Interface (CSI) allows OpenShift Container Platform to consume storage from storage back ends that implement the CSI interface as persistent storage.

NOTE
OpenShift Container Platform 4.17 supports version 1.6.0 of the CSI specification.

CSI architecture

CSI drivers are typically shipped as container images. These containers are not aware of OpenShift Container Platform where they run. To use CSI-compatible storage back end in OpenShift Container Platform, the cluster administrator must deploy several components that serve as a bridge between OpenShift Container Platform and the storage driver.

The following diagram provides a high-level overview about the components running in pods in the OpenShift Container Platform cluster.

CSI drivers that are installed with OpenShift Container Platform supported by OpenShift Container Platform:

If your CSI driver is not listed in the following table, you must follow the installation instructions provided by your **CSI storage vendor** to use their supported CSI features.

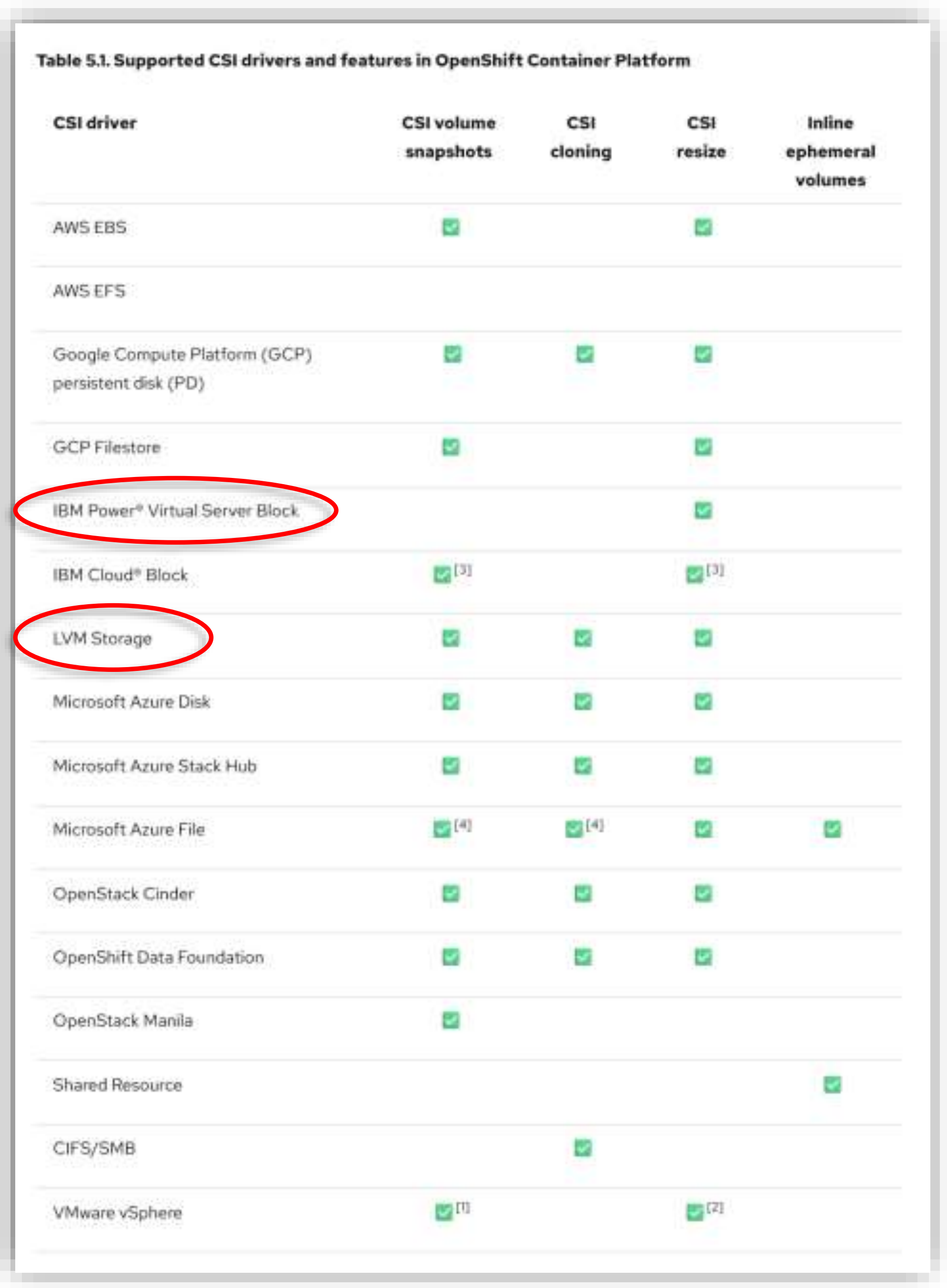


Table 5.1. Supported CSI drivers and features in OpenShift Container Platform

CSI driver	CSI volume snapshots	CSI cloning	CSI resize	Inline ephemeral volumes
AWS EBS	✓		✓	
AWS EFS				
Google Compute Platform (GCP) persistent disk (PD)	✓	✓	✓	
GCP Filestore	✓		✓	
IBM Power® Virtual Server Block			✓	
IBM Cloud® Block	✓ [3]		✓ [3]	
LVM Storage	✓	✓	✓	
Microsoft Azure Disk	✓	✓	✓	
Microsoft Azure Stack Hub	✓	✓	✓	
Microsoft Azure File	✓ [4]	✓ [4]	✓	✓
OpenStack Cinder	✓	✓	✓	
OpenShift Data Foundation	✓	✓	✓	
OpenStack Manila	✓			
Shared Resource				✓
CIFS/SMB		✓		
VMware vSphere	✓ [1]		✓ [2]	

Dynamic provisioning

Dynamic provisioning of persistent storage depends on the capabilities of the CSI driver and underlying storage back end. The provider of the CSI driver should document how to create a storage class in OpenShift Container Platform and the parameters available for configuration.
The created storage class can be configured to enable dynamic provisioning.

IBM block storage CSI driver

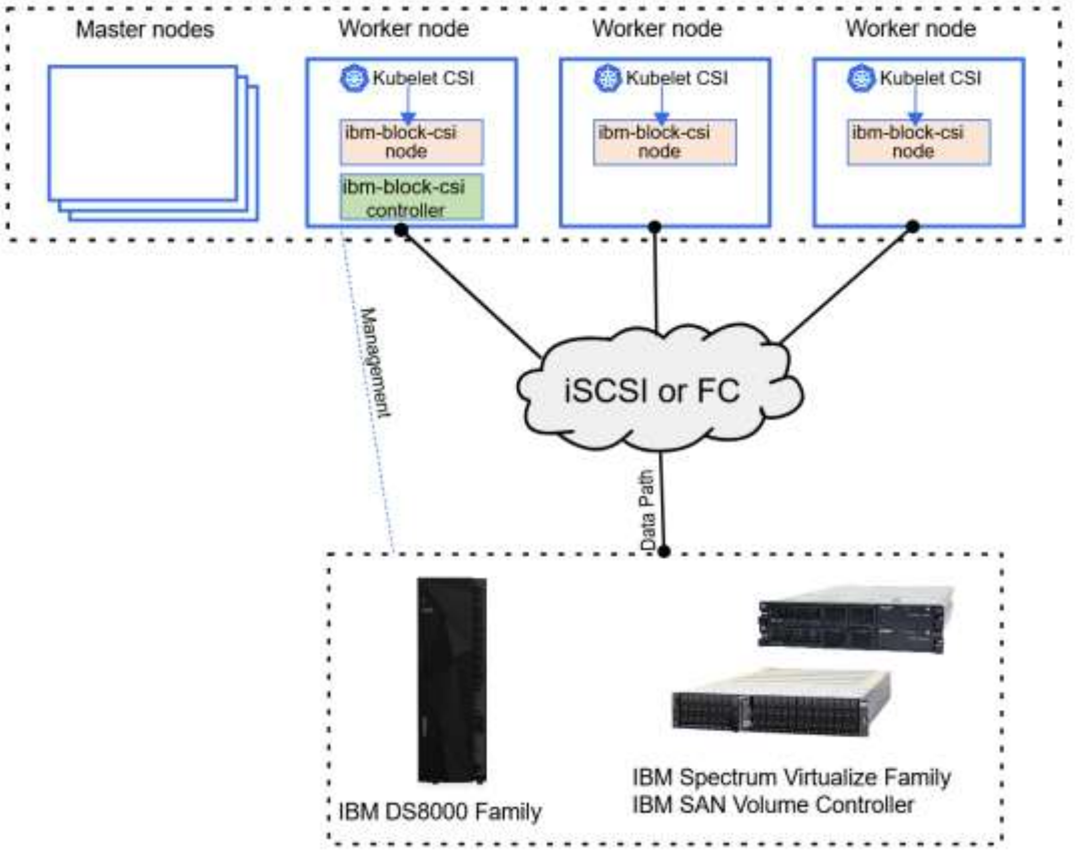
IBM® block storage CSI driver is leveraged by Kubernetes persistent volumes (PVs) to dynamically provision for block storage used with stateful containers.

IBM block storage CSI driver is based on an open-source IBM project (CSI driver), included as a part of IBM storage orchestration for containers. IBM storage orchestration for containers enables enterprises to implement a modern container-driven hybrid multicloud environment that can reduce IT costs and enhance business agility, while continuing to derive value from existing systems.

By leveraging CSI (Container Storage Interface) drivers for IBM storage systems, Kubernetes persistent volumes (PVs) can be dynamically provisioned for block or file storage to be used with stateful containers, such as database applications (IBM Db2®, MongoDB, PostgreSQL, etc.) running in Red Hat® OpenShift® Container Platform and/or Kubernetes clusters. Storage provisioning can be fully automated with additional support of cluster orchestration systems to automatically deploy, scale, and manage containerized applications.

IBM storage orchestration for containers includes the following driver types for storage provisioning:

- The IBM block storage CSI driver, for block storage,
- The IBM Storage® Scale CSI driver, for file storage.



https://www.ibm.com/docs/en/stg-block-csi-driver/1.12.0?topic=overview
https://github.com/ibm/ibm-block-csi-driver

IBM i
continuous innovation
continuous integration

Généralités
sur l’installation
d’ OpenShift

Plusieurs méthodes

Les méthodes de déploiement d'OpenShift sont nombreuses, adaptées à la plateforme qui l'accueille :

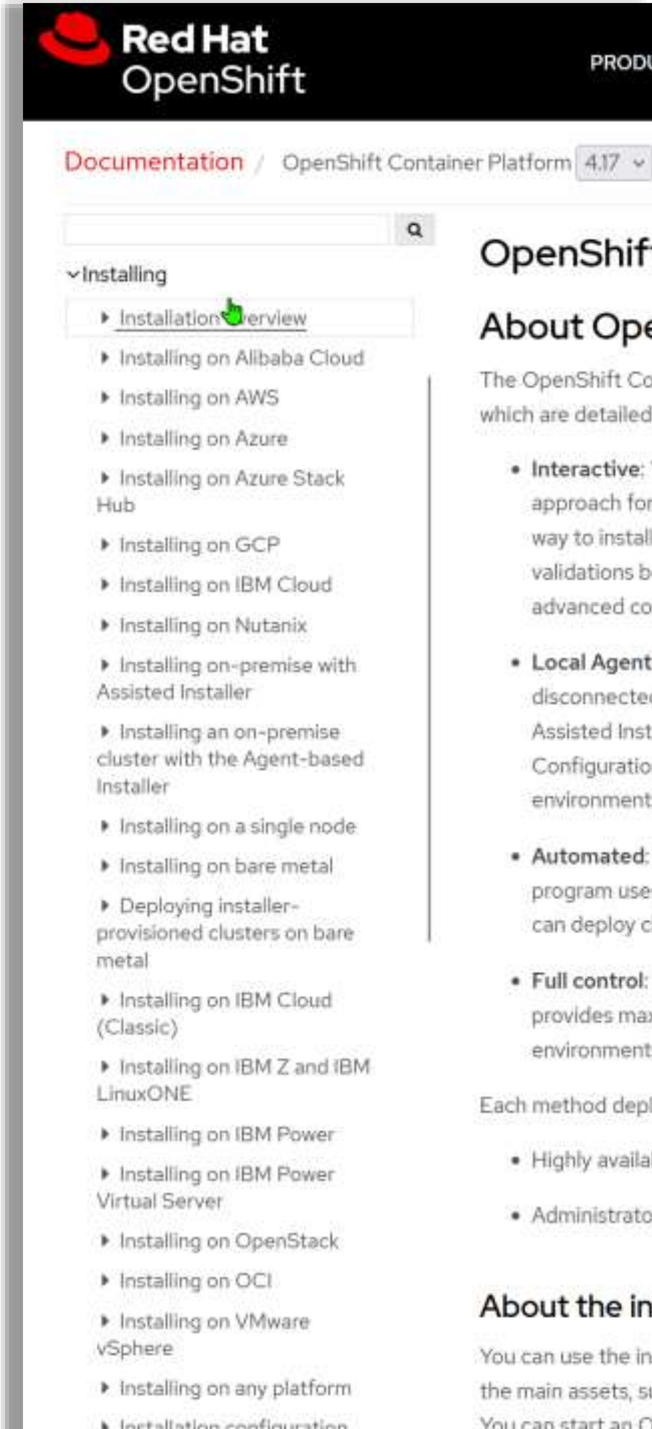
- Bare-metal
- VM donc Hyperviseur
- Cloud par l'automatisation proposée par le fournisseur
- Cloud privé vSphere, OpenStack, etc.
- x86, Power, IBM Z
- Connecté à internet ou déconnecté

Elles ont évolué :

- Totalement manuelle ☹
- Helper node ☺
- Web-based Assisted Installer : mode connecté ☺
- Local Agent-based : mode déconnecté ☺

❤ Avant tout, il faut choisir la bonne méthode pour son cas !

Assisted Installer et Agent-based sont les plus récentes et les plus faciles



Documentation / OpenShift Container Platform 4.17

Installing

- Installation overview
- Installing on Alibaba Cloud
- Installing on AWS
- Installing on Azure
- Installing on Azure Stack Hub
- Installing on GCP
- Installing on IBM Cloud
- Installing on Nutanix
- Installing on-premise with Assisted Installer
- Installing an on-premise cluster with the Agent-based Installer
- Installing on a single node
- Installing on bare metal
- Deploying installer-provisioned clusters on bare metal
- Installing on IBM Cloud (Classic)
- Installing on IBM Z and IBM LinuxONE
- Installing on IBM Power
- Installing on IBM Power Virtual Server
- Installing on OpenStack
- Installing on OCI
- Installing on VMware vSphere
- Installing on any platform
- Installation configuration

OpenShif

About Ope

The OpenShift Co
which are detailed

- **Interactive:**
approach for
way to instal
validations b
advanced co
- **Local Agent**
disconnecte
Assisted Inst
Configuratio
environment
- **Automated:**
program use
can deploy c
- **Full control:**
provides ma
environment

Each method depl

- Highly availa
- Administrato

About the in

You can use the in
the main assets, s
You can start an O

Glossaire

https://docs.openshift.com/container-platform/4.17/installing/overview/index.html

The OpenShift Container Platform installation program

A program that provisions the infrastructure and deploys a cluster.

Ignition config files

A file that the Ignition tool uses to configure Red Hat Enterprise Linux CoreOS (RHCOS) during operating system initialization. The installation program generates different Ignition configuration files to initialize bootstrap, control plane, and worker nodes.

Bootstrap node

A temporary machine that runs a minimal Kubernetes configuration required to deploy the OpenShift Container Platform control plane.

Control plane

A container orchestration layer that exposes the API and interfaces to define, deploy, and manage the lifecycle of containers. Also known as control plane machines.

Compute node

Nodes that are responsible for executing workloads for cluster users. Also known as worker nodes.

Kubernetes manifests

Specifications of a Kubernetes API object in a JSON or YAML format. A configuration file can include deployments, config maps, secrets, daemonsets, and so on.

Kubelet

A primary node agent that runs on each node in the cluster to ensure that containers are running in a pod.

Load balancers

A load balancer serves as the single point of contact for clients. Load balancers for the API distribute incoming traffic across control plane nodes.

Operators

The preferred method of packaging, deploying, and managing a Kubernetes application in an OpenShift Container Platform cluster. An operator takes human operational knowledge and encodes it into software that is easily packaged and shared with customers.

Machine Config Operator

An Operator that manages and applies configurations and updates of the base operating system and container runtime, including everything between the kernel and kubelet, for the nodes in the cluster.

UPI / IPI ?   Disconnected ?

User Provisioned Infrastructure (UPI):

The UPI installation is highly customizable and tunable. The infrastructure is not configured within the installation of Red Hat OpenShift, and the cluster heavily relies on the proper configuration of the following infrastructure services:

- DHCP
- DNS
- Proxy
- Router
- NAT
- Firewall
- web hosting
- LDAP (Active Directory or equivalent)
- TFTP/SFTP server
- NFS (NAS or equivalent tuned Linux server)

NOTE: Given the customization and flexibility of an UPI installation, this methodology would be the most representative for an **on-premises enterprise deployment**.

Installer Provisioned Infrastructure (IPI):

The installation program deploys and configures the infrastructure that the cluster runs on.

The IPI installation provides a turn-key solution and includes all the necessary infrastructure services within the Red Hat OpenShift cluster.

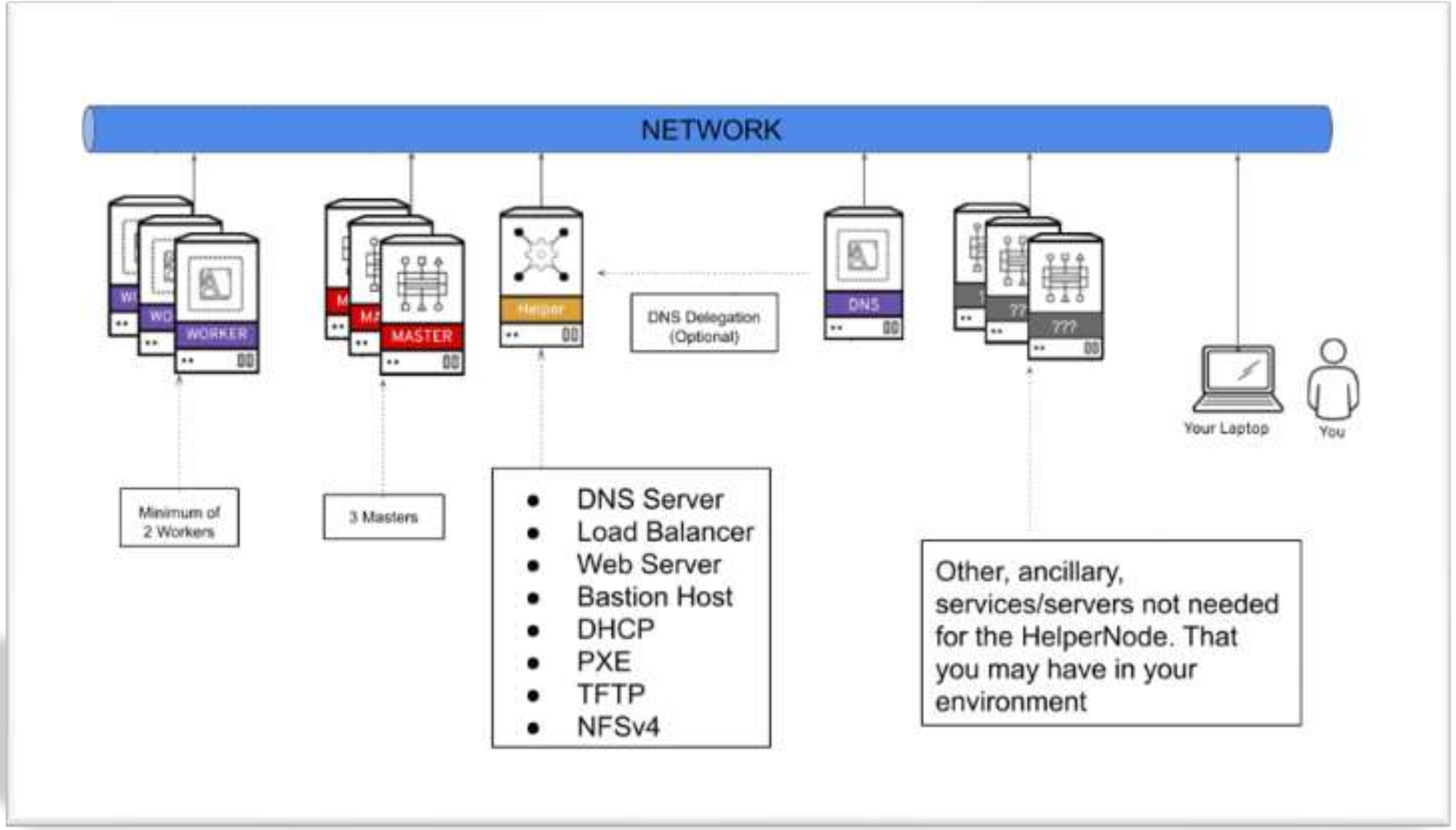
Significant planning must be done before deployment to ensure your team calculated and sized correctly the capacity, size of the deployment, and number of control planes.

Disconnected installation

In some situations, parts of a data center might not have access to the internet, even through proxy servers. You can still install the OpenShift Container Platform in these environments, but you must download the required software and images and make them available to the disconnected environment.

Installation UPI par helper-node

- Adapté pour un déploiement en VM ou bare metal de plusieurs nodes & un contrôle avancé
- Playbook Ansible qui installe les prérequis pour démarrer une installation OpenShift locale (on prem) UPI:
  - ✓ DHCP
  - ✓ DNS
  - ✓ PXE
  - ✓ Load balancer
  - ✓ TFTP
  - ✓ NFS
- https://github.com/redhat-cop/ocp4-helpernode



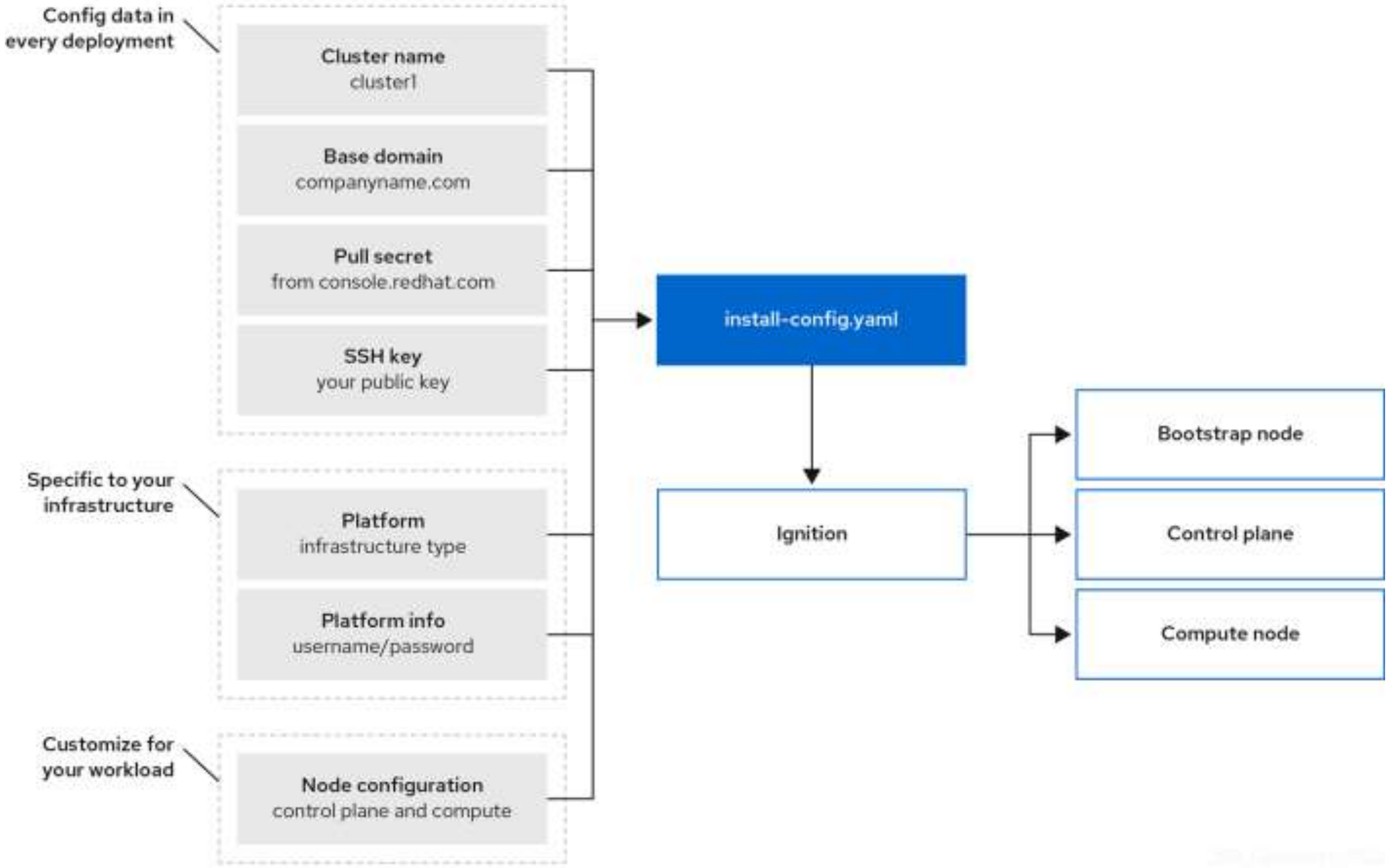
- DNS Server
- Load Balancer
- Web Server
- Bastion Host
- DHCP
- PXE
- TFTP
- NFSv4

Other, ancillary, services/servers not needed for the HelperNode. That you may have in your environment

Aperçu du processus d’installation

Lisez cette page très instructive :

https://docs.openshift.com/container-platform/4.17/installing/overview/index.html



Config data in every deployment

Cluster name
cluster1

Base domain
companyname.com

Pull secret
from console.redhat.com

SSH key
your public key

Specific to your infrastructure

Platform
infrastructure type

Platform info
username/password

Customize for your workload

Node configuration
control plane and compute

install-config.yaml

Ignition

Bootstrap node

Control plane

Compute node

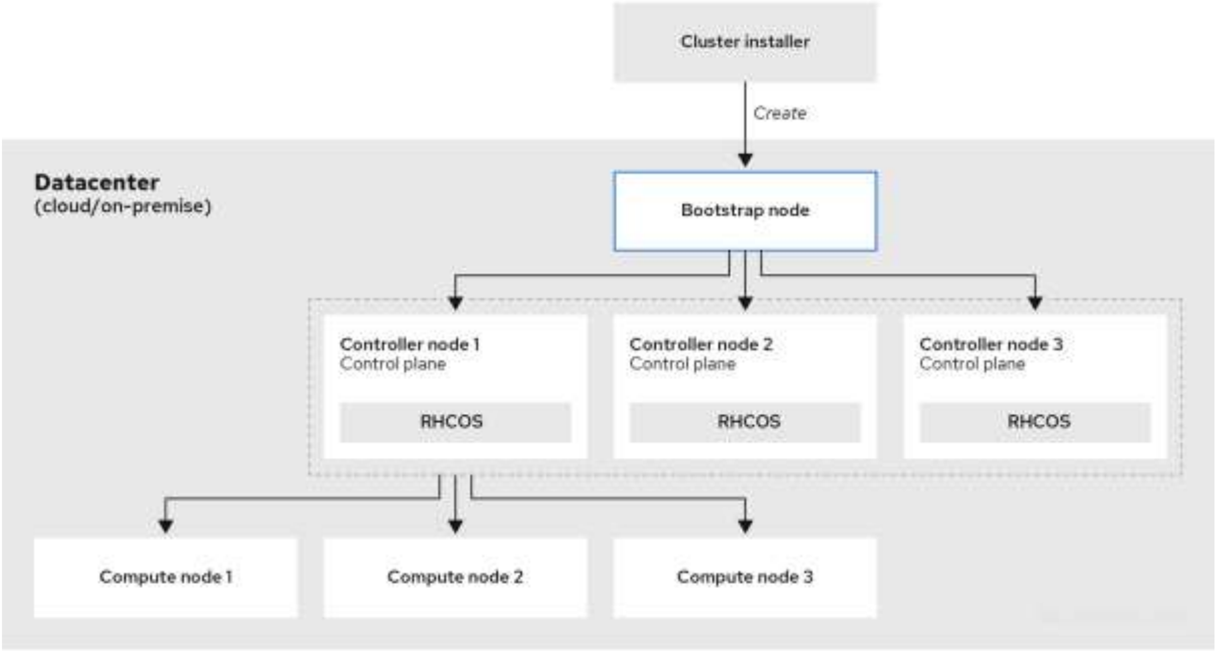
Détails du processus d’installation

Ce processus est plus ou moins visible selon la méthode d’installation. En mode UPI, helper-node, il est bien visible à travers les reboots successifs. Avec le helper-node, la machine de Boostrap est le helper node.

When a cluster is provisioned, each machine in the cluster requires information about the cluster. OpenShift Container Platform uses a temporary bootstrap machine during initial configuration to provide the required information to the permanent control plane.

The temporary bootstrap machine boots by using an Ignition config file that describes how to create the cluster. The bootstrap machine creates the control plane machines that make up the control plane.

The control plane machines then create the compute machines, which are also known as worker machines.



https://docs.openshift.com/container-platform/4.17/installing/overview/index.html

Université IBM i

19 et 20 novembre 2024

IBM i
continuous innovation
continuous integration

Installation

d’un SNO

Single Node OpenShift

Prérequis Single node OpenShift

Single-node OpenShift basic installation

- 8 CPU cores : en fait des threads de processeur.
  - ✓ En x86 hyperthreading, 8 vCPU, 2 threads par cœurs, donc 4 cœurs x86.
  - ✓ Sur Power : 8 threads par cœur, donc 1 vCPU
  - ✓ 16 GB RAM
- 100 GB storage

Single-node OpenShift + multicluster engine

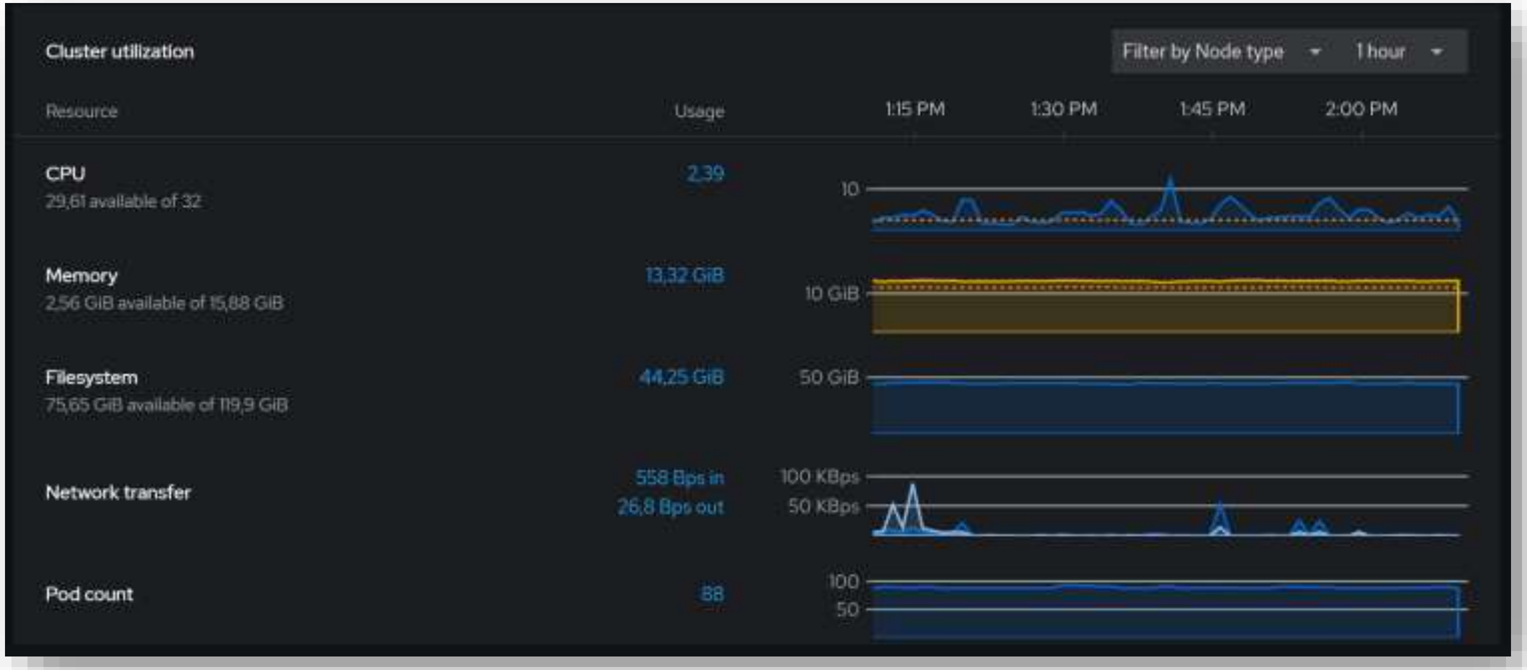
- Additional 8 CPU cores
- Additional 32 GB RAM

Consommation mesurée sur SNO 4.17 sur POWER9 :

- 28,77 CPU available of 32 (4 vCPU)
- Memory 2,5 GiB available of 15,88 GiB
- Filesystem 75 GiB available of 119,9 GiB

Recommendation stockage :

Configurer 2 disques, un pour openshift, l’autre pour un driver de stockage et les workloads à déployer.



Cluster utilization

Filter by Node type ▾ 1 hour ▾

Resource	Usage
CPU 29,61 available of 32	2,39
Memory 2,56 GiB available of 15,88 GiB	13,32 GiB
Filesystem 75,65 GiB available of 119,9 GiB	44,25 GiB
Network transfer	558 Bps in 26,8 Bps out
Pod count	88

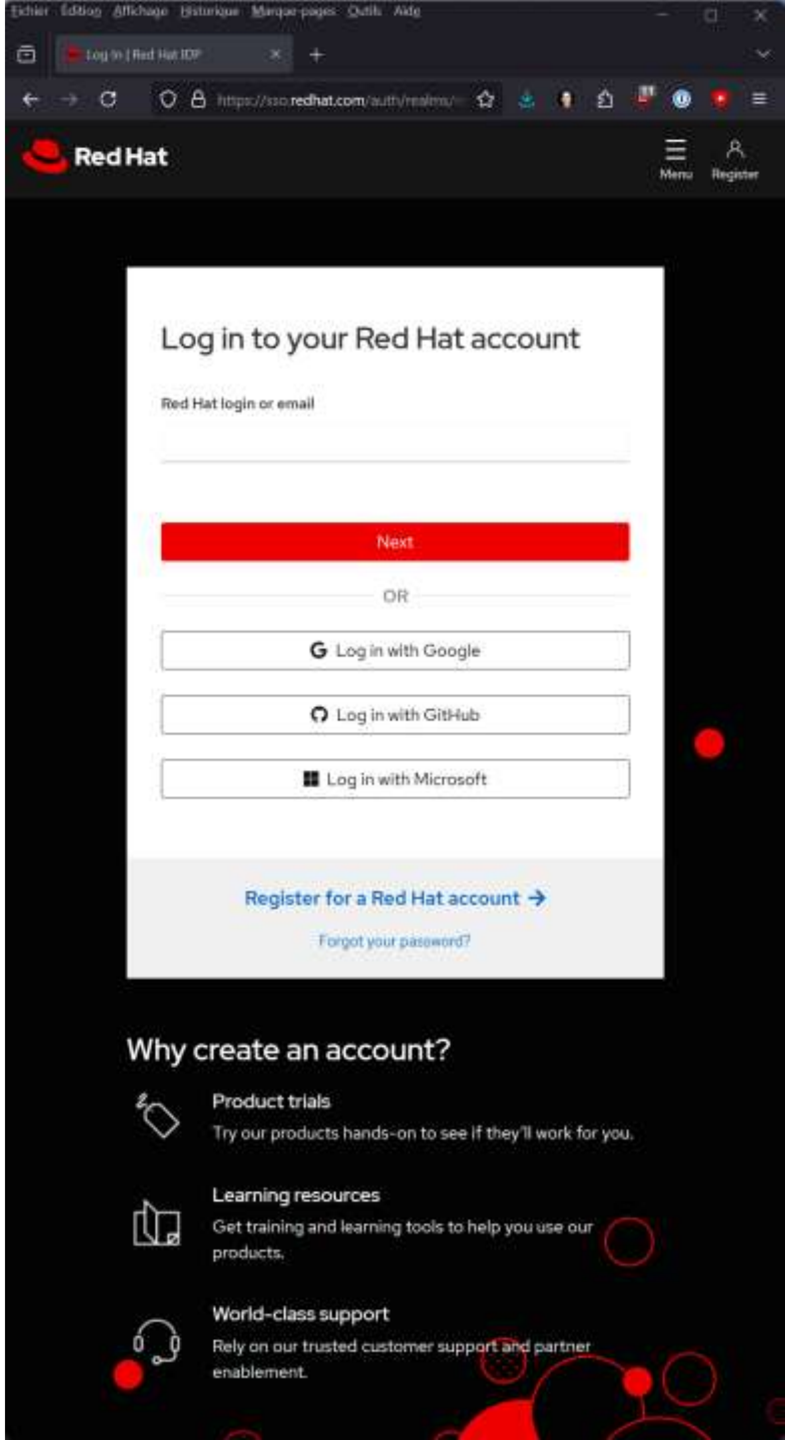
Créez un compte Red Hat

Vous avez besoin de ce compte pour

- Versions d’essai des produits
- Éducation
- Support

Et donc aussi pour installer OpenShift, quelle que soit la méthode :

- Accéder à l’assisted Install
- En UPI, car il vous faut un « pull secret »


Fichier Édition Affichage Historique Marque-pages Outils Aide
Log in | Red Hat IDP
https://sso.redhat.com/auth/realms/
Red Hat
Menu
Register

Log in to your Red Hat account

Red Hat login or email

Next

OR

Log in with Google

Log in with GitHub

Log in with Microsoft

Register for a Red Hat account →

Forgot your password?

Why create an account?

Product trials
Try our products hands-on to see if they'll work for you.

Learning resources
Get training and learning tools to help you use our products.

World-class support
Rely on our trusted customer support and partner enablement.

3 Options pour installer un SNO

1. Utiliser un helper-node : une VM séparée avec les services nécessaires à une installation « traditionnelle » on premise. On peut faire plus simple…

2. Créer une ISO et booter dessus avec le programme d’installation et un DHCP minimal, ou passer dans l’ISO l’adresse IP voulue. On peut faire plus simple !

3. Utiliser l’installation assistée par https://console.redhat.com/openshift
   - Le plus simple
   - Ne nécessite pas de VM « bastion » pour l’installation et l’usage
   - Le Bastion joue le rôle de load balancer, mais ici Single Node, donc pas d’équilibrage, donc pas besoin de load balancer
   - ISO réutilisable et complète, ou minimale

Installation assistée – Assisted Installer

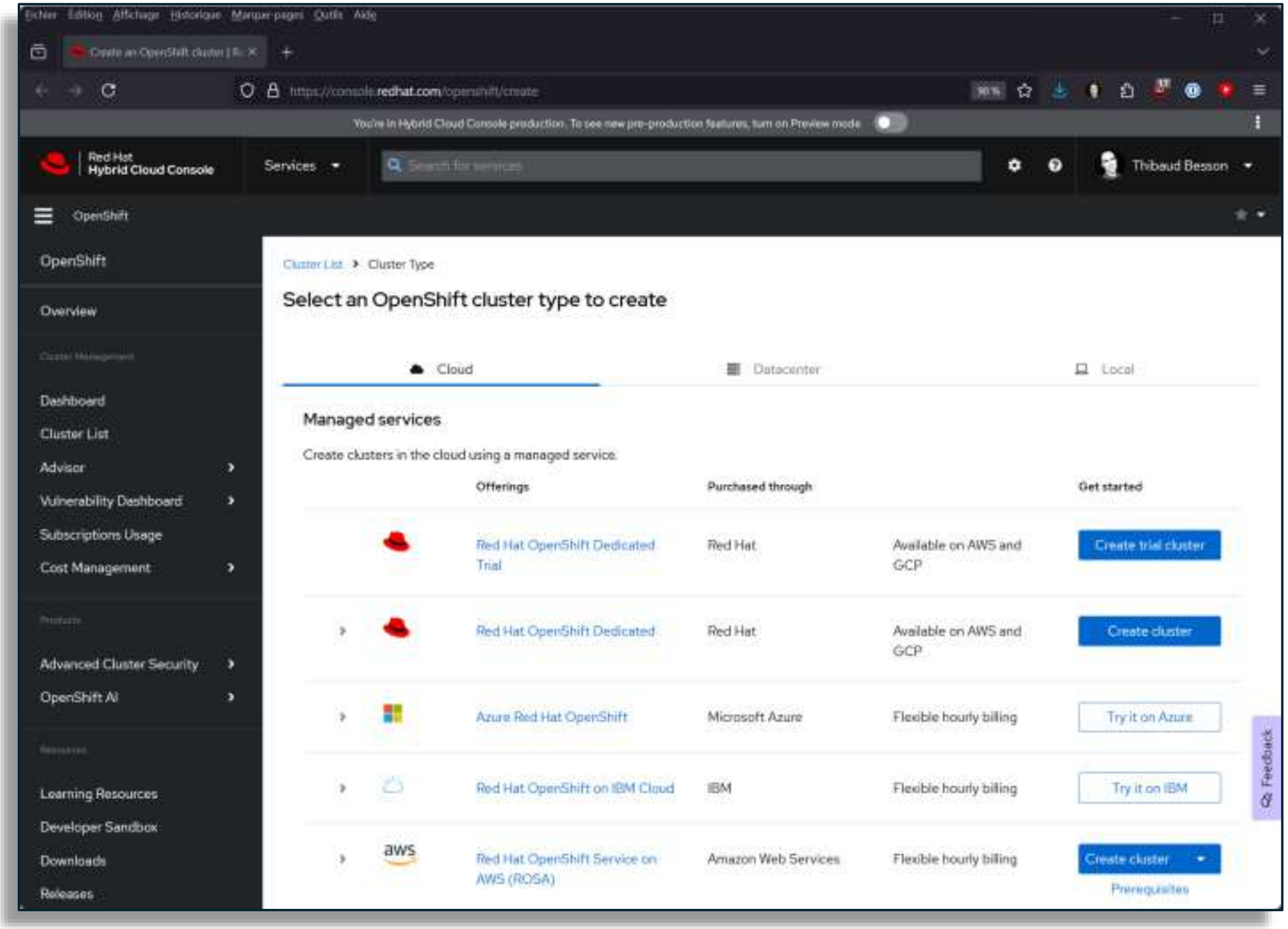
- Tout le processus est géré par

l'OpenShift Cluster Manager Hybrid Cloud Console

https://console.redhat.com/openshift/overview

Assisted Installer provide the following advantages:

- A web interface to perform cluster installation without having to create the installation configuration file.
- Boostrap machine is no longer required, the bootstrapping process takes place on a random node of the cluster.
- A simplified deployment model that does not require in-depth knowledge of OpenShift.
- Flexible API.
- Deploying Single Node OpenShift (SNO).
- Installing OpenShift Virtualization and OpenShift Data Foundation (formerly OpenShift Container Storage) from the web interface.


Fichier Edition Affichage Historique Marque-pages Outils Aide
Create an OpenShift cluster | R
https://console.redhat.com/openshift/create
90 %
You're in Hybrid Cloud Console production. To see new pre-production features, turn on Preview mode
Red Hat
Hybrid Cloud Console
Services
Search for services
Thibaud Besson
OpenShift
OpenShift
Overview
Dashboard
Cluster List
Advisor
Vulnerability Dashboard
Subscriptions Usage
Cost Management
Advanced Cluster Security
OpenShift AI
Learning Resources
Developer Sandbox
Downloads
Releases
Cluster List › Cluster Type
Cloud
Datacenter
Local
Feedback

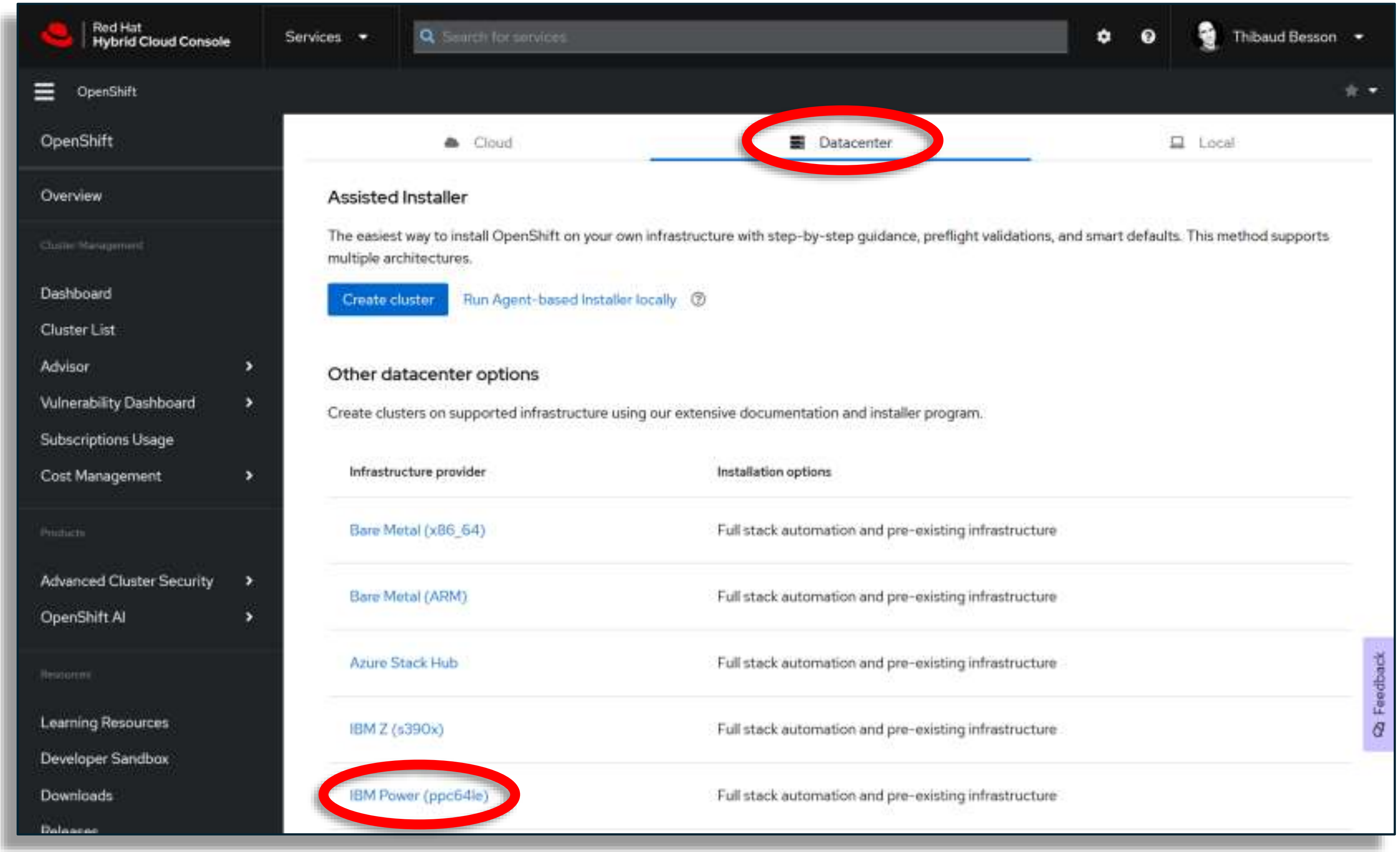
Select an OpenShift cluster type to create

Managed services

Create clusters in the cloud using a managed service.

Offerings	Purchased through		Get started
Red Hat OpenShift Dedicated Trial	Red Hat	Available on AWS and GCP	Create trial cluster
Red Hat OpenShift Dedicated	Red Hat	Available on AWS and GCP	Create cluster
Azure Red Hat OpenShift	Microsoft Azure	Flexible hourly billing	Try it on Azure
Red Hat OpenShift on IBM Cloud	IBM	Flexible hourly billing	Try it on IBM
Red Hat OpenShift Service on AWS (ROSA)	Amazon Web Services	Flexible hourly billing	Create cluster Prerequisites

https://console.redhat.com/openshift/create



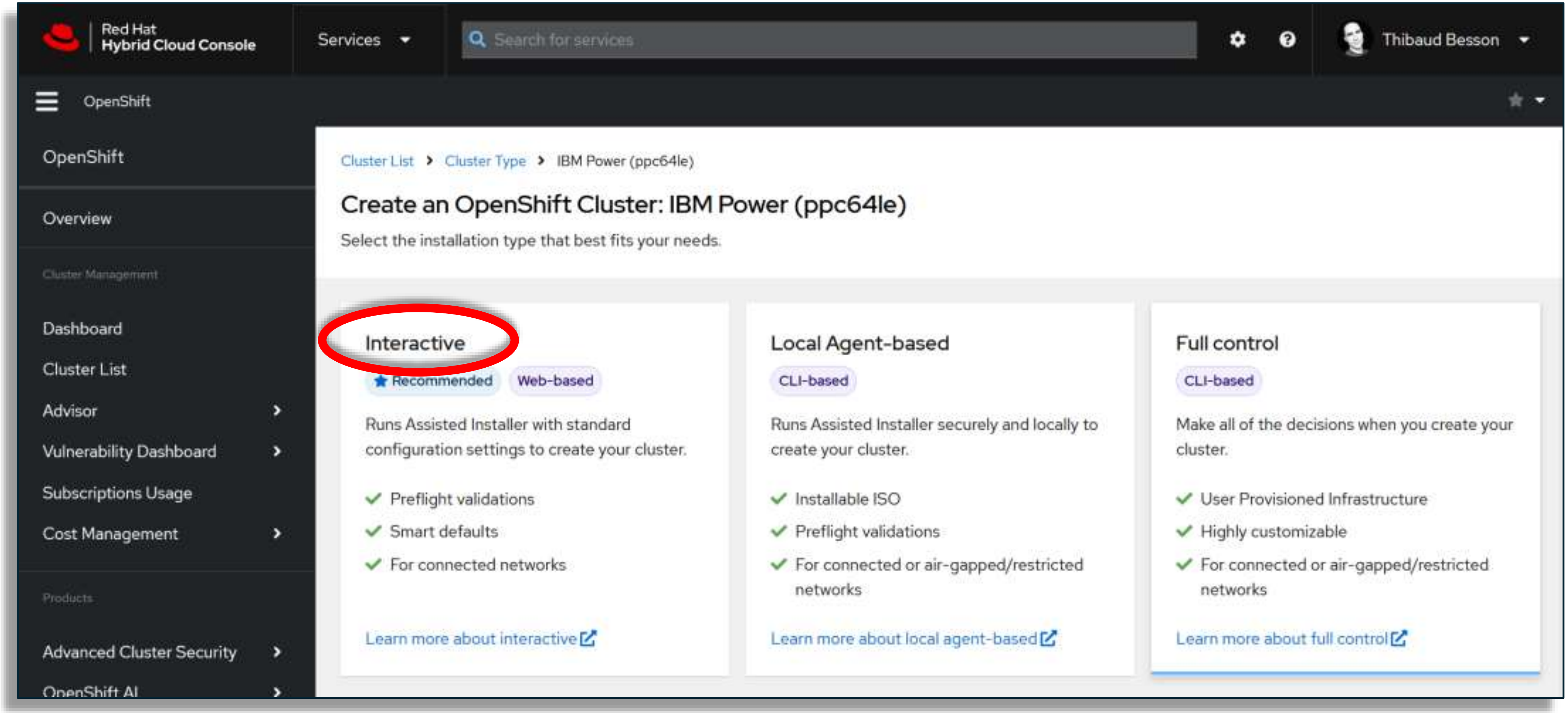
Assisted Installer

The easiest way to install OpenShift on your own infrastructure with step-by-step guidance, preflight validations, and smart defaults. This method supports multiple architectures.

Other datacenter options

Create clusters on supported infrastructure using our extensive documentation and installer program.

Infrastructure provider	Installation options
Bare Metal (x86_64)	Full stack automation and pre-existing infrastructure
Bare Metal (ARM)	Full stack automation and pre-existing infrastructure
Azure Stack Hub	Full stack automation and pre-existing infrastructure
IBM Z (s390x)	Full stack automation and pre-existing infrastructure
IBM Power (ppc64le)	Full stack automation and pre-existing infrastructure



Cluster List › Cluster Type › IBM Power (ppc64le)

Create an OpenShift Cluster: IBM Power (ppc64le)

Select the installation type that best fits your needs.

Interactive

★ Recommended | Web-based

Runs Assisted Installer with standard configuration settings to create your cluster.

- ✓ Preflight validations
- ✓ Smart defaults
- ✓ For connected networks

Learn more about interactive

Local Agent-based

CLI-based

Runs Assisted Installer securely and locally to create your cluster.

- ✓ Installable ISO
- ✓ Preflight validations
- ✓ For connected or air-gapped/restricted networks

Learn more about local agent-based

Full control

CLI-based

Make all of the decisions when you create your cluster.

- ✓ User Provisioned Infrastructure
- ✓ Highly customizable
- ✓ For connected or air-gapped/restricted networks

Learn more about full control

Renseigner les détails du cluster

Dans l’interface web :

✓ Cluster name

✓ Base domain

✓ Version

✓ CPU architecture

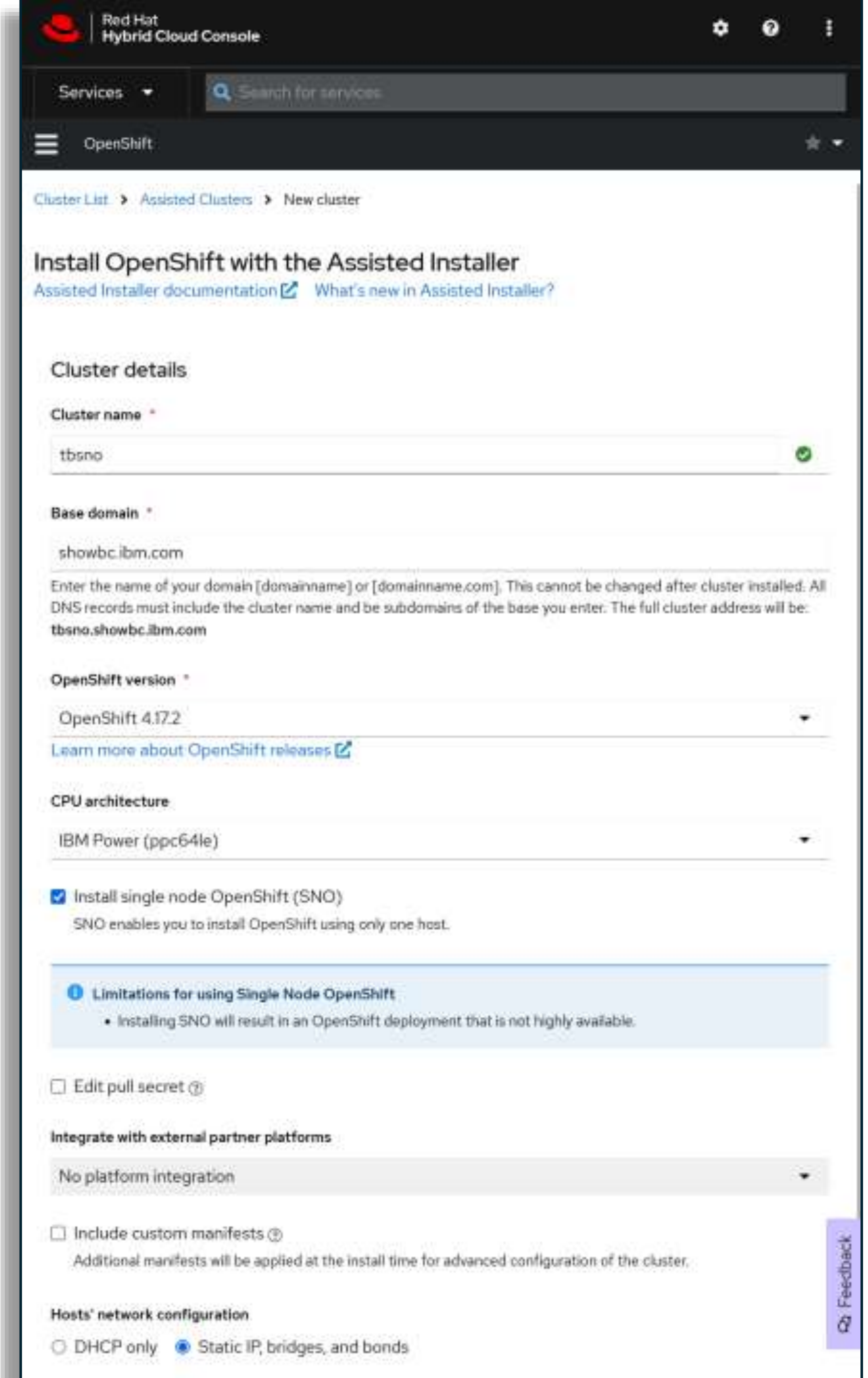
✓ SNO

✓ Static IP

Dans l’infrastructure IT :

✓ DNS setup

- ➢ A records
- ➢ PTR records



Cluster List › Assisted Clusters › New cluster

Install OpenShift with the Assisted Installer

Assisted Installer documentation What's new in Assisted Installer?

Cluster details

Cluster name *

tbsno

Base domain *

showbc.ibm.com

Enter the name of your domain [domainname] or [domainname.com]. This cannot be changed after cluster installed. All DNS records must include the cluster name and be subdomains of the base you enter. The full cluster address will be: **tbsno.showbc.ibm.com**

OpenShift version *

OpenShift 4.17.2

Learn more about OpenShift releases

CPU architecture

IBM Power (ppc64le)

☑ Install single node OpenShift (SNO)

SNO enables you to install OpenShift using only one host.

Limitations for using Single Node OpenShift

- Installing SNO will result in an OpenShift deployment that is not highly available.

☐ Edit pull secret

Integrate with external partner platforms

No platform integration

☐ Include custom manifests

Additional manifests will be applied at the install time for advanced configuration of the cluster.

Hosts' network configuration

○ DHCP only ● Static IP, bridges, and bonds

DNS records à renseigner dans l’infra

- Kubernetes API
- Ingress route : The OpenShift Container Platform application wildcard
- Les IP des machines physiques ou virtuelles du control plane et du data plane
- La resolution inverse DNS est requise pour l’API Kubernetes, les machines du control plane et du data plane

Configuration DNS - Détails

Une fois OpenShift déployé, la résolution de nom (par DNS ou autre) est requise pour les composants suivants :

Component	Record	Description
Kubernetes API	api.<cluster_name>.<base_domain>.	A DNS A/AAAA or CNAME record, and a DNS PTR record, to identify the API load balancer. These records must be resolvable by **both clients external to the cluster and from all the nodes within the cluster**.
	api-int.<cluster_name>.<base_domain>.	A DNS A/AAAA or CNAME record, and a DNS PTR record, to internally identify the API load balancer. These records must be resolvable from all the nodes within the cluster.
		The API server must be able to **resolve the worker nodes by the hostnames that are recorded in Kubernetes**. If the API server cannot resolve the node names, then proxied API calls can fail, and you cannot retrieve logs from pods.
Ingress routes : Routes to applications deployed in cluster	*.apps.<cluster_name>.<base_domain>.	A **wildcard DNS A/AAAA or CNAME record that refers to the application ingress load balancer**. The application ingress load balancer targets the machines that run the Ingress Controller pods. The Ingress Controller pods run on the compute machines by default. These records must be resolvable by both clients external to the cluster and from all the nodes within the cluster. For example, **console-openshift-console.apps.<cluster_name>.<base_domain>** is used as a wildcard route to the OpenShift Container Platform console.
Control plane machines	<master><n>.<cluster_name>.<base_domain>.	DNS A/AAAA or CNAME records and DNS PTR records to identify each machine for the control plane nodes. These records must be **resolvable by the nodes within the cluster.**
Compute machines	<worker><n>.<cluster_name>.<base_domain>.	DNS A/AAAA or CNAME records and DNS PTR records to identify each machine for the worker nodes. These records must be **resolvable by the nodes within the cluster.**

DNS Forward Zone

```
$TTL 1W
@   IN  SOA  ns1.example.com.  root (
            2019070700  ; serial
            3H      ; refresh (3 hours)
            30M     ; retry (30 minutes)
            2W      ; expiry (2 weeks)
            1W )    ; minimum (1 week)
    IN  NS  ns1.example.com.
    IN  MX 10  smtp.example.com.
;
;
ns1.example.com.                  IN  A  192.168.1.1
smtp.example.com.                 IN  A  192.168.1.5
;
helper.example.com.               IN  A  192.168.1.5
api.ocp4.example.com.             IN  A  192.168.1.5
api-int.ocp4.example.com.         IN  A  192.168.1.5
*.apps.ocp4.example.com.          IN  A  192.168.1.5
;
control-plane0.ocp4.example.com.  IN  A  192.168.1.97
control-plane1.ocp4.example.com.  IN  A  192.168.1.98
control-plane2.ocp4.example.com.  IN  A  192.168.1.99
;
worker0.ocp4.example.com.         IN  A  192.168.1.11
worker1.ocp4.example.com.         IN  A  192.168.1.7
;
;EOF
```

Bastion :

- helper node,
- API,
- ingress routes des applications

DNS Reverse Zone

```
$$TTL 1W
@ IN SOA ns1.example.com. root (
         2019070700 ; serial
         3H         ; refresh (3 hours)
         30M        ; retry (30 minutes)
         2W         ; expiry (2 weeks)
         1W )       ; minimum (1 week)
  IN NS ns1.example.com.
;
;
 5.1.168.192.in-addr.arpa.  IN  PTR   api.ocp4.example.com.
 5.1.168.192.in-addr.arpa.  IN  PTR   api-int.ocp4.example.com.
;
97.1.168.192.in-addr.arpa.  IN  PTR   control-plane0.ocp4.example.com.
98.1.168.192.in-addr.arpa.  IN  PTR   control-plane1.ocp4.example.com.
99.1.168.192.in-addr.arpa.  IN  PTR   control-plane2.ocp4.example.com.
;
11.1.168.192.in-addr.arpa.  IN  PTR   worker0.ocp4.example.com.
 7.1.168.192.in-addr.arpa.  IN  PTR   worker1.ocp4.example.com.
;
;EOF
```

Pas d’accès à votre DNS ? Solutions :

Écrire en dur les adresses IP <-> Hostnames dans /etc/hosts

1. Linux client : /etc/hosts
2. Linux client : dnsmask
3. Windows client : C:\Windows\System32\drivers\etc\hosts
4. Windows DNS setup par subnet : DNS zone delegation

Inconvénient du /etc/hosts : il ne permet pas le wildcard `*.apps``.ocp4.example.com`

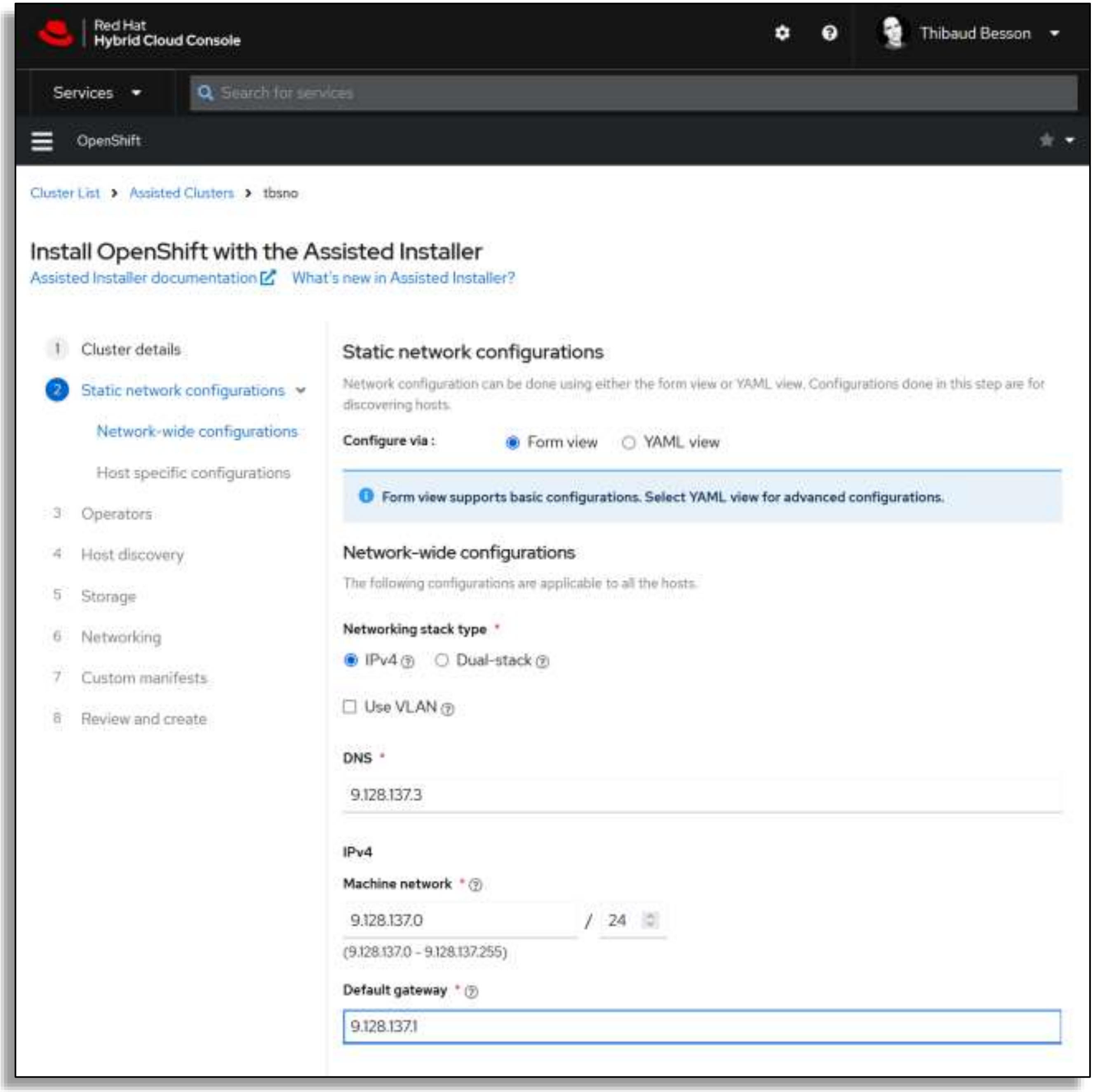
Il faut renseigner les hostnames de chaque nouvelle application déployée dans OpenShift !

Option pour windows : pointer sur le DNS (bastion si il existe, ou celui du réseau de OpenShift) par une « DNS zone delegation » :

- using the PowerShell command Add-DnsClientNrptRule
- Add-DnsClientNrptRule -Namespace ".mycluster.com" -NameServers "10.0.0.1"

Configuration réseau

- IP
- Gateway
- DNS
- Masque
- Adresse MAC



Install OpenShift with the Assisted Installer

Assisted Installer documentation   What's new in Assisted Installer?

Static network configurations

Network configuration can be done using either the form view or YAML view. Configurations done in this step are for discovering hosts.

Configure via : ● Form view ○ YAML view

Form view supports basic configurations. Select YAML view for advanced configurations.

Network-wide configurations

The following configurations are applicable to all the hosts.

Networking stack type *

● IPv4 ○ Dual-stack

□ Use VLAN

DNS *

9.128.137.3

IPv4

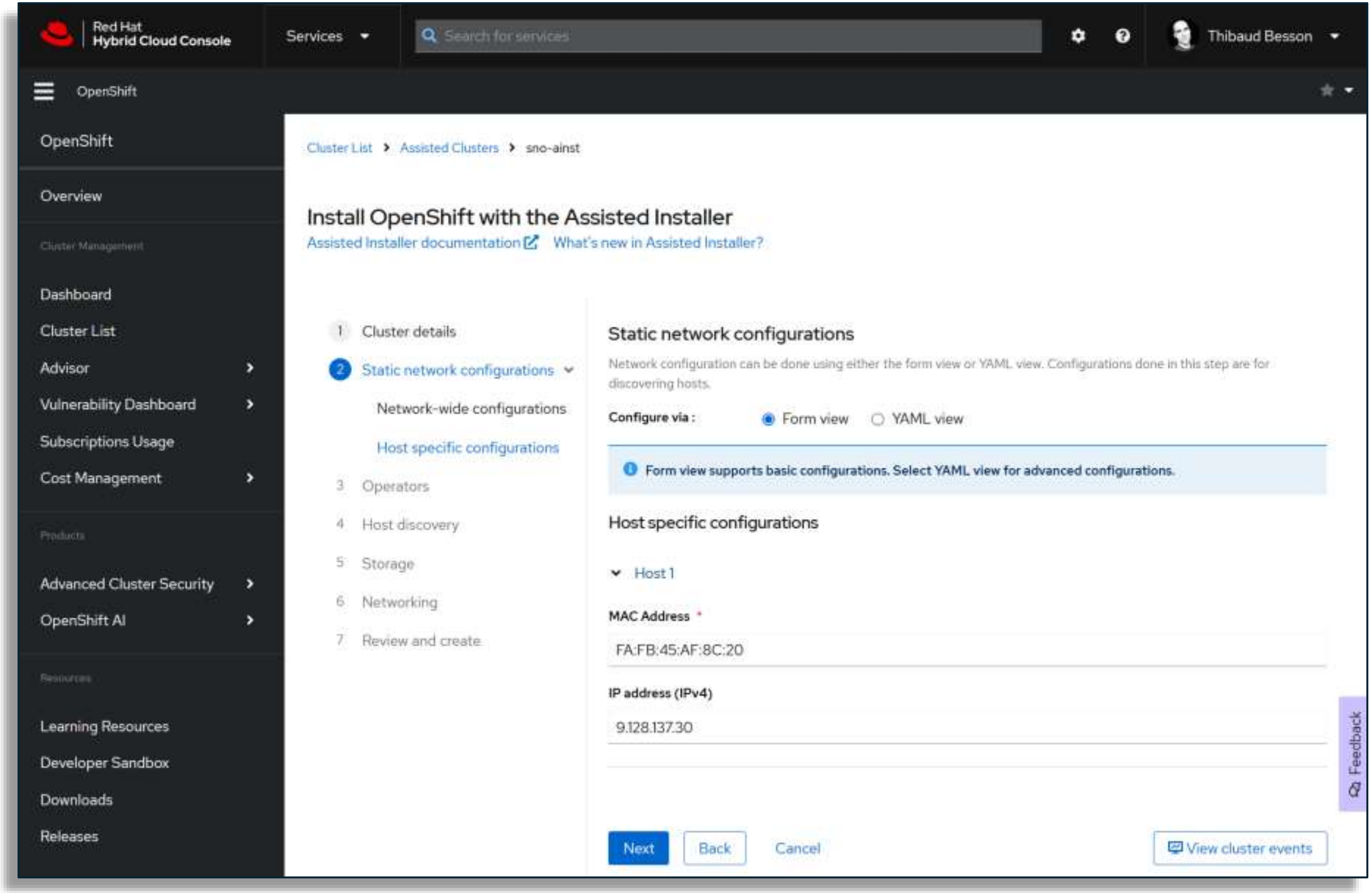
Machine network *

9.128.137.0 / 24

(9.128.137.0 - 9.128.137.255)

Default gateway *

9.128.137.1



Install OpenShift with the Assisted Installer

Assisted Installer documentation What's new in Assisted Installer?

1 Cluster details
2 Static network configurations
Network-wide configurations
Host specific configurations
3 Operators
4 Host discovery
5 Storage
6 Networking
7 Review and create

Static network configurations

Network configuration can be done using either the form view or YAML view. Configurations done in this step are for discovering hosts.

Configure via : ● Form view ○ YAML view

Form view supports basic configurations. Select YAML view for advanced configurations.

Host specific configurations

Host 1

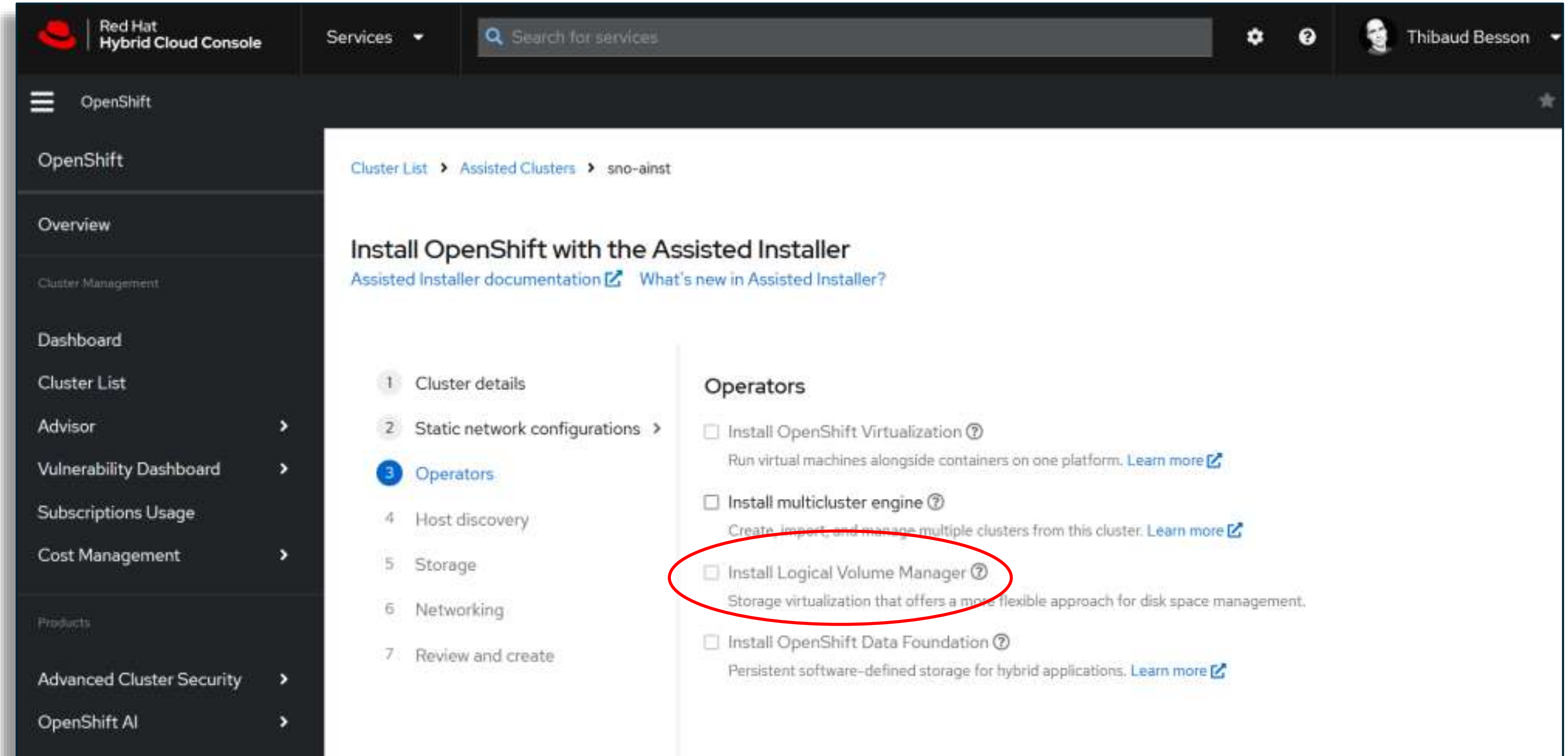
MAC Address *

FA:FB:45:AF:8C:20

IP address (IPv4)

9.128.137.30

Installation optionnelle d’opérateurs



Cluster List › Assisted Clusters › sno-ainst

Install OpenShift with the Assisted Installer

Assisted Installer documentation   What's new in Assisted Installer?

1 Cluster details
2 Static network configurations
3 Operators
4 Host discovery
5 Storage
6 Networking
7 Review and create

Operators

☐ Install OpenShift Virtualization
Run virtual machines alongside containers on one platform. Learn more

☐ Install multicluster engine
Create, import, and manage multiple clusters from this cluster. Learn more

☐ Install Logical Volume Manager
Storage virtualization that offers a more flexible approach for disk space management.

☐ Install OpenShift Data Foundation
Persistent software-defined storage for hybrid applications. Learn more

Qu’est-ce qu’un opérateur ?

Les opérateurs Kubernetes sont des contrôleurs spécifiques aux applications qui étendent l'API Kubernetes pour créer, configurer et gérer des instances d'applications complexes. Ils codent les connaissances opérationnelles et automatisent des tâches telles que :

- Déploiement d'applications
- Effectuer et restaurer des sauvegardes
- Gestion des mises à niveau
- Mise à l'échelle
- Basculement
- Opérations personnalisées spécifiques aux applications

Exemple d'opérateur

Key features in common with CloudNativePG

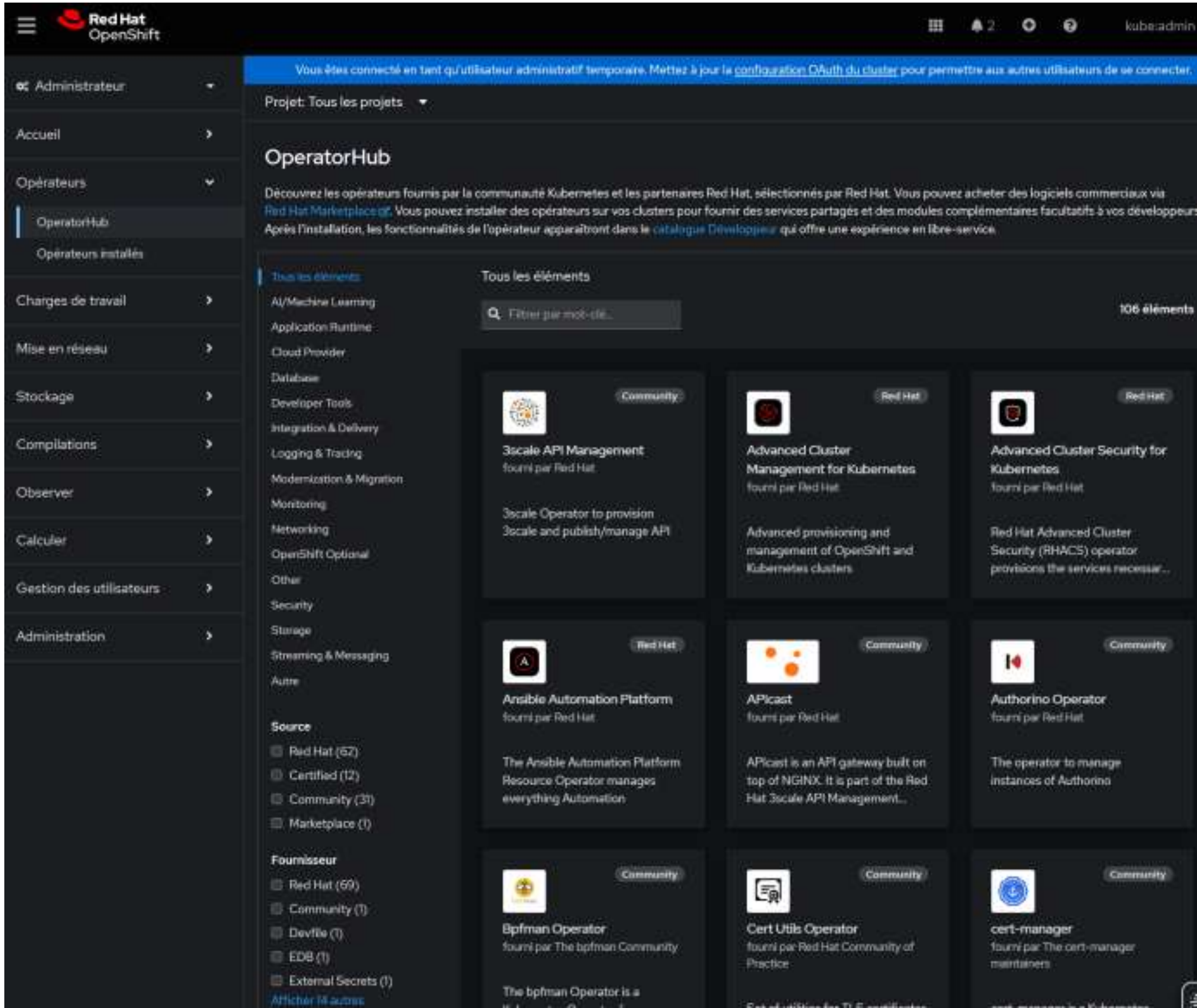
- Kubernetes API integration for high availability
- Self-healing through failover and automated recreation of replicas
- Capacity management with scale up/down capabilities
- Planned switchovers for scheduled maintenance
- Read-only and read-write Kubernetes services definitions
- Rolling updates for Postgres minor versions and operator upgrades
- Continuous backup and point-in-time recovery
- Connection Pooling with PgBouncer
- Integrated metrics exporter out of the box
- PostgreSQL replication across multiple Kubernetes clusters
- Separate volume for WAL files

Features unique to EDB Postgres of Kubernetes

- Long Term Support
- Support on IBM Power and z/Linux through partnership with IBM
- Oracle compatibility through EDB Postgres Advanced Sever
- Transparent Data Encryption (TDE) through EDB Postgres Advanced Server
- Cold backup support with Kasten and Velero/OADP
- Generic adapter for third-party Kubernetes backup tools

Catalogue d’opérateurs

- Permet de déployer une application dans Openshift simplement et selon les bonnes pratiques.



Vous êtes connecté en tant qu'utilisateur administratif temporaire. Mettez à jour la configuration OAuth du cluster pour permettre aux autres utilisateurs de se connecter.

OperatorHub

Découvrez les opérateurs fournis par la communauté Kubernetes et les partenaires Red Hat, sélectionnés par Red Hat. Vous pouvez acheter des logiciels commerciaux via Red Hat Marketplace. Vous pouvez installer des opérateurs sur vos clusters pour fournir des services partagés et des modules complémentaires facultatifs à vos développeurs Après l'installation, les fonctionnalités de l'opérateur apparaîtront dans le catalogue Développeur qui offre une expérience en libre-service.

3scale API Management
fourni par Red Hat
3scale Operator to provision 3scale and publish/manage API

Advanced Cluster Management for Kubernetes
fourni par Red Hat
Advanced provisioning and management of OpenShift and Kubernetes clusters

Advanced Cluster Security for Kubernetes
fourni par Red Hat
Red Hat Advanced Cluster Security (RHACS) operator provisions the services necessar...

Ansible Automation Platform
fourni par Red Hat
The Ansible Automation Platform Resource Operator manages everything Automation

APIcast
fourni par Red Hat
APIcast is an API gateway built on top of NGINX. It is part of the Red Hat 3scale API Management...

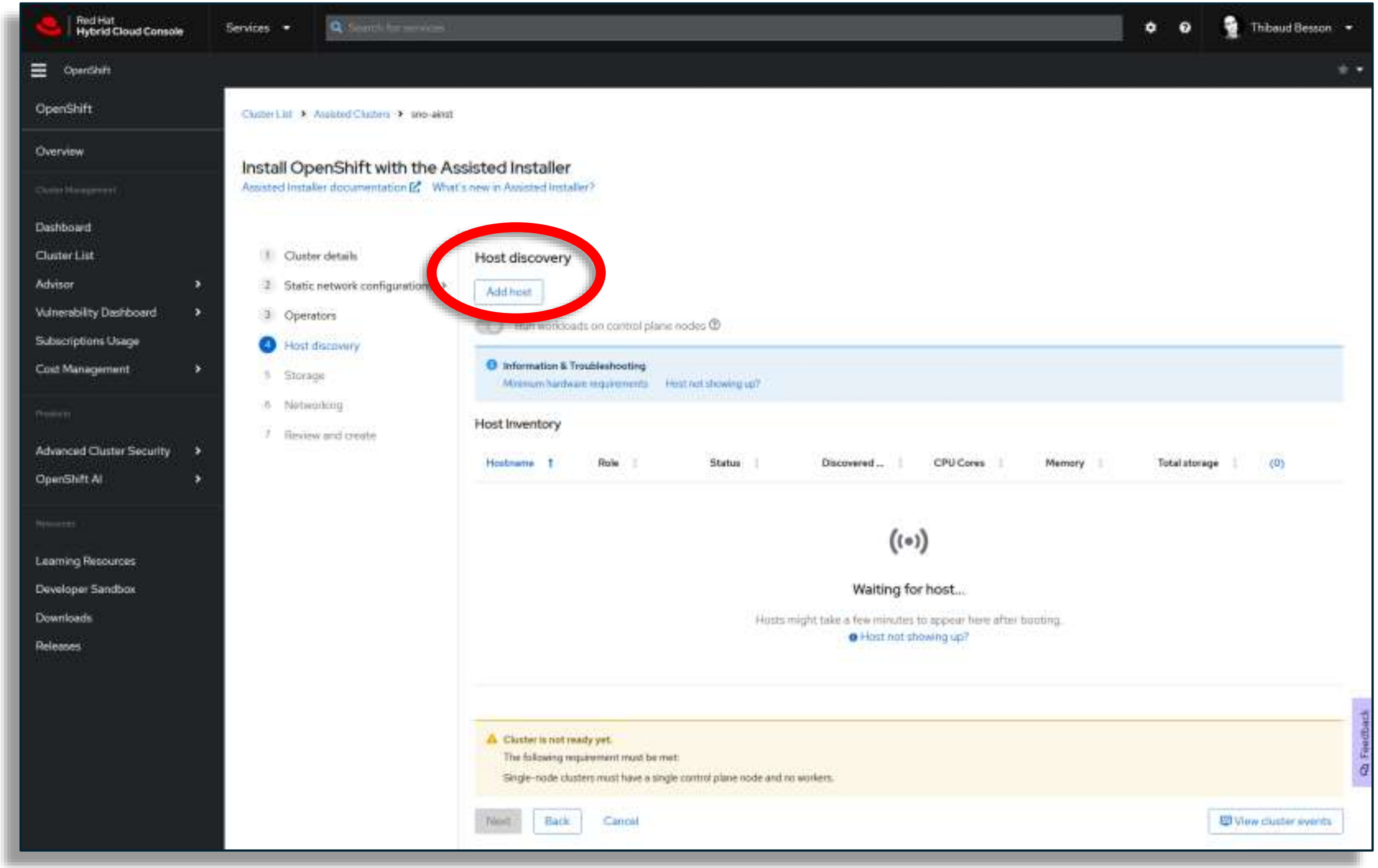
Authorino Operator
fourni par Red Hat
The operator to manage instances of Authorino

Bpfman Operator
fourni par The bpfman Community
The bpfman Operator is a

Cert Utils Operator
fourni par Red Hat Community of Practice

cert-manager
fourni par The cert-manager maintainers

Ajouter le host : démarrer l’installation



Cluster List › Assisted Clusters › sno-aiinst

Install OpenShift with the Assisted Installer

Assisted Installer documentation | What's new in Assisted Installer?

1 Cluster details
2 Static network configuration
3 Operators
4 Host discovery
5 Storage
6 Networking
7 Review and create

Host discovery

Add host

Run workloads on control plane nodes

Information & Troubleshooting
Minimum hardware requirements | Host not showing up?

Host Inventory

Hostname	Role	Status	Discovered ...	CPU Cores	Memory	Total storage	(0)

Waiting for host...

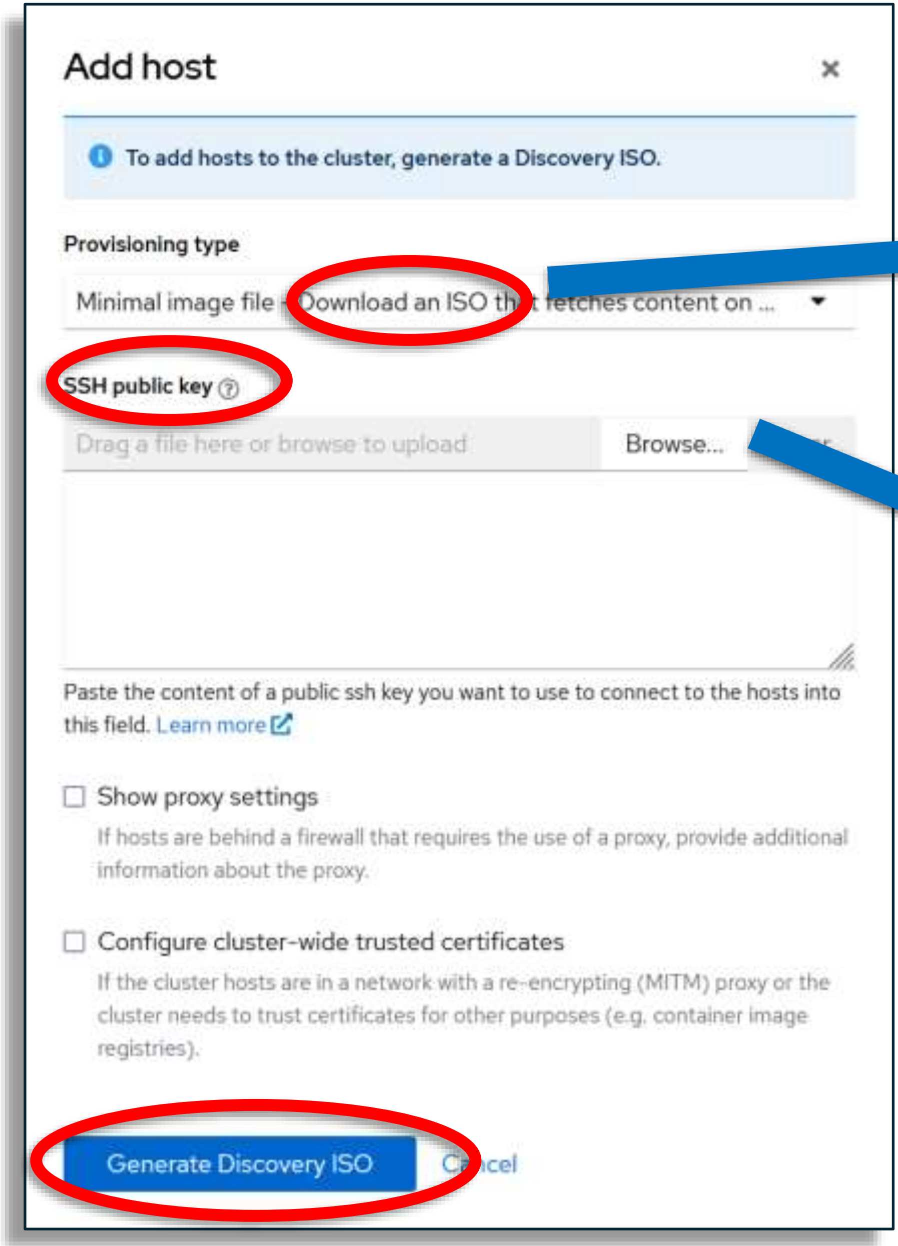
Hosts might take a few minutes to appear here after booting.

Host not showing up?

Cluster is not ready yet.
The following requirement must be met:
Single-node clusters must have a single control plane node and no workers.

Next | Back | Cancel | View cluster events

Ajouter le host : démarrer l’installation



Add host

×

To add hosts to the cluster, generate a Discovery ISO.

Provisioning type

Minimal image file - Download an ISO that fetches content on ...

SSH public key

Drag a file here or browse to upload | Browse...

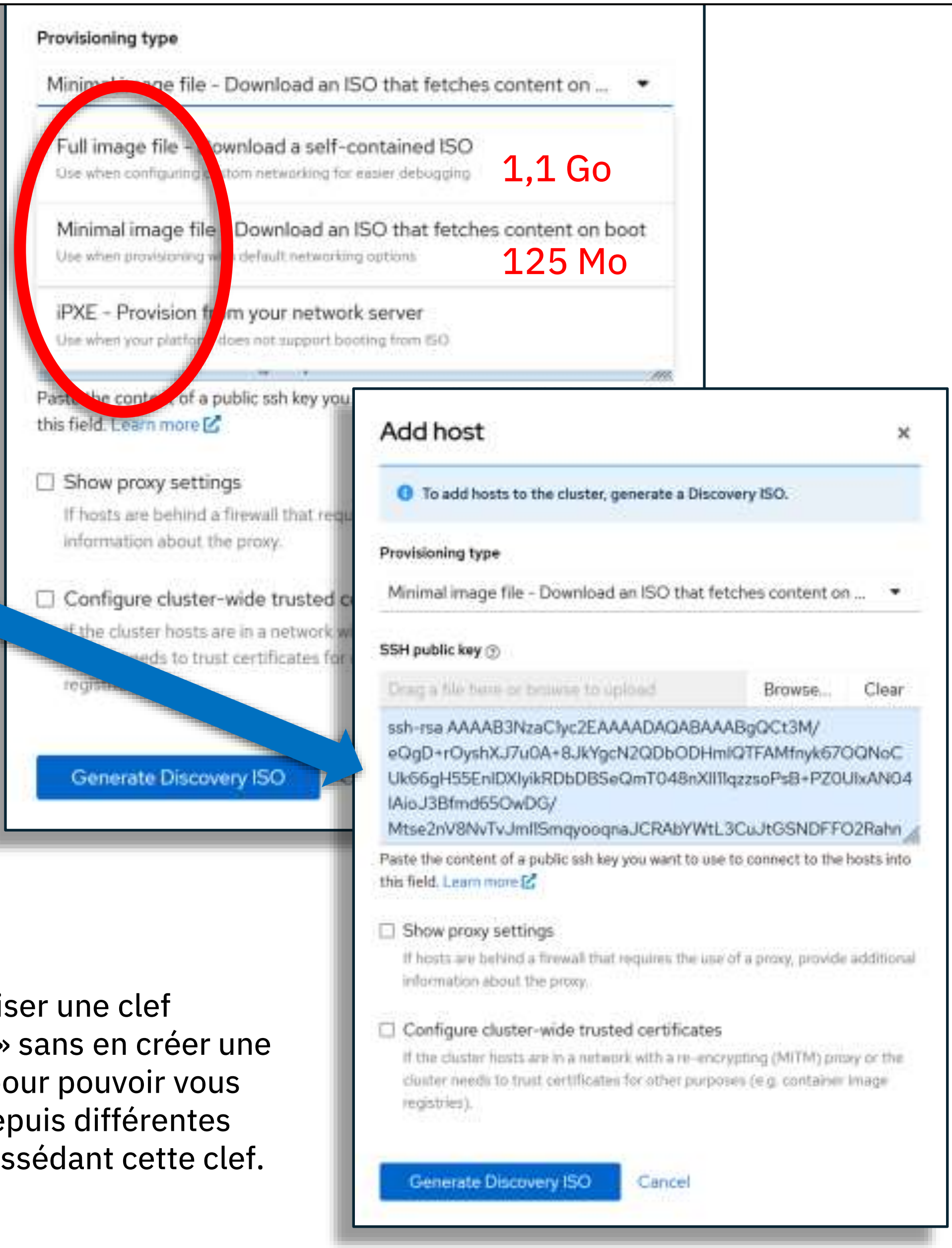
Paste the content of a public ssh key you want to use to connect to the hosts into this field. Learn more

□ Show proxy settings

If hosts are behind a firewall that requires the use of a proxy, provide additional information about the proxy.

□ Configure cluster-wide trusted certificates

If the cluster hosts are in a network with a re-encrypting (MITM) proxy or the cluster needs to trust certificates for other purposes (e.g. container image registries).



Provisioning type

Minimal image file - Download an ISO that fetches content on ...

- Full image file - Download a self-contained ISO
  Use when configuring custom networking for easier debugging
- Minimal image file - Download an ISO that fetches content on boot
  Use when provisioning with default networking options
- iPXE - Provision from your network server
  Use when your platform does not support booting from ISO

1,1 Go

125 Mo

Paste the content of a public ssh key you
this field. Learn more

□ Show proxy settings

If hosts are behind a firewall that requ
information about the proxy.

□ Configure cluster-wide trusted c

If the cluster hosts are in a network w
needs to trust certificates fo
regis

Add host

×

To add hosts to the cluster, generate a Discovery ISO.

Provisioning type

Minimal image file - Download an ISO that fetches content on ...

SSH public key

Drag a file here or browse to upload | Browse... | Clear

```
ssh-rsa AAAAB3NzaClyc2EAAAADAQABAAABgQCt3M/
eQgD+rOyshXJ7u0A+8JkYgcN2QDbODHmlQTFAMfnyk67OQNoC
Uk66gH55EnIDXlyikRDbDBSeQmT048nXlI1lqzzsoPsB+PZ0UlxAN04
lAioJ3Bfmd65OwDG/
Mtse2nV8NvTvJmlISmqyooqnaJCRAbYWtL3CuJtGSNDFFO2Rahn
```

Paste the content of a public ssh key you want to use to connect to the hosts into this field. Learn more

□ Show proxy settings

If hosts are behind a firewall that requires the use of a proxy, provide additional information about the proxy.

□ Configure cluster-wide trusted certificates

If the cluster hosts are in a network with a re-encrypting (MITM) proxy or the cluster needs to trust certificates for other purposes (e.g. container image registries).

Pas de connexion à CoreOS par login / mot de passe !

Conseil : utiliser une clef « habituelle » sans en créer une spécifique, pour pouvoir vous connecter depuis différentes machines possédant cette clef.

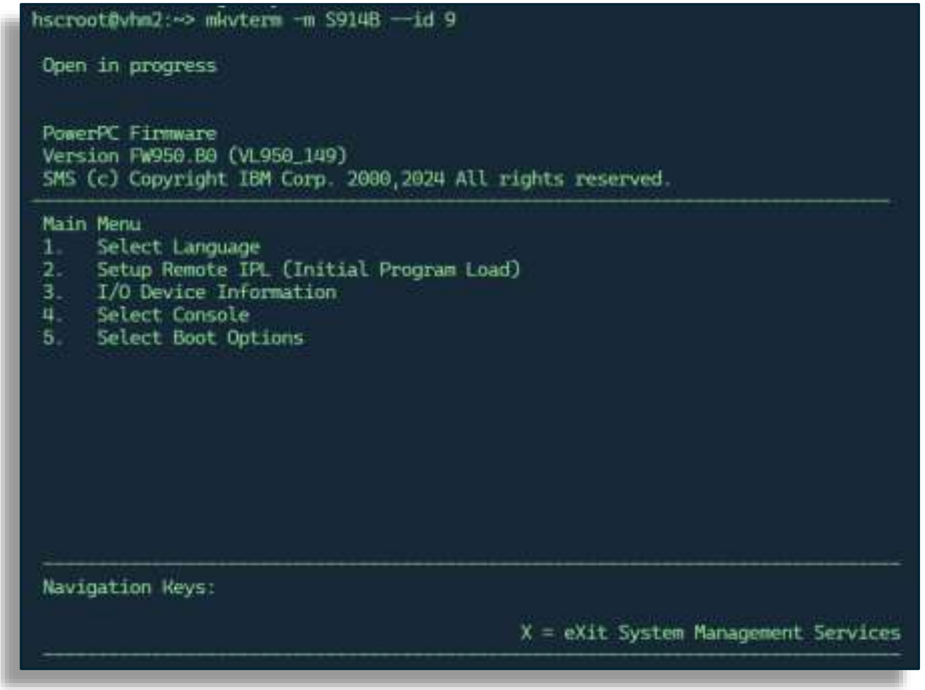
Copy ISO to VIOS repo

Copier l’ISO Dans le lecteur DVD virtuel

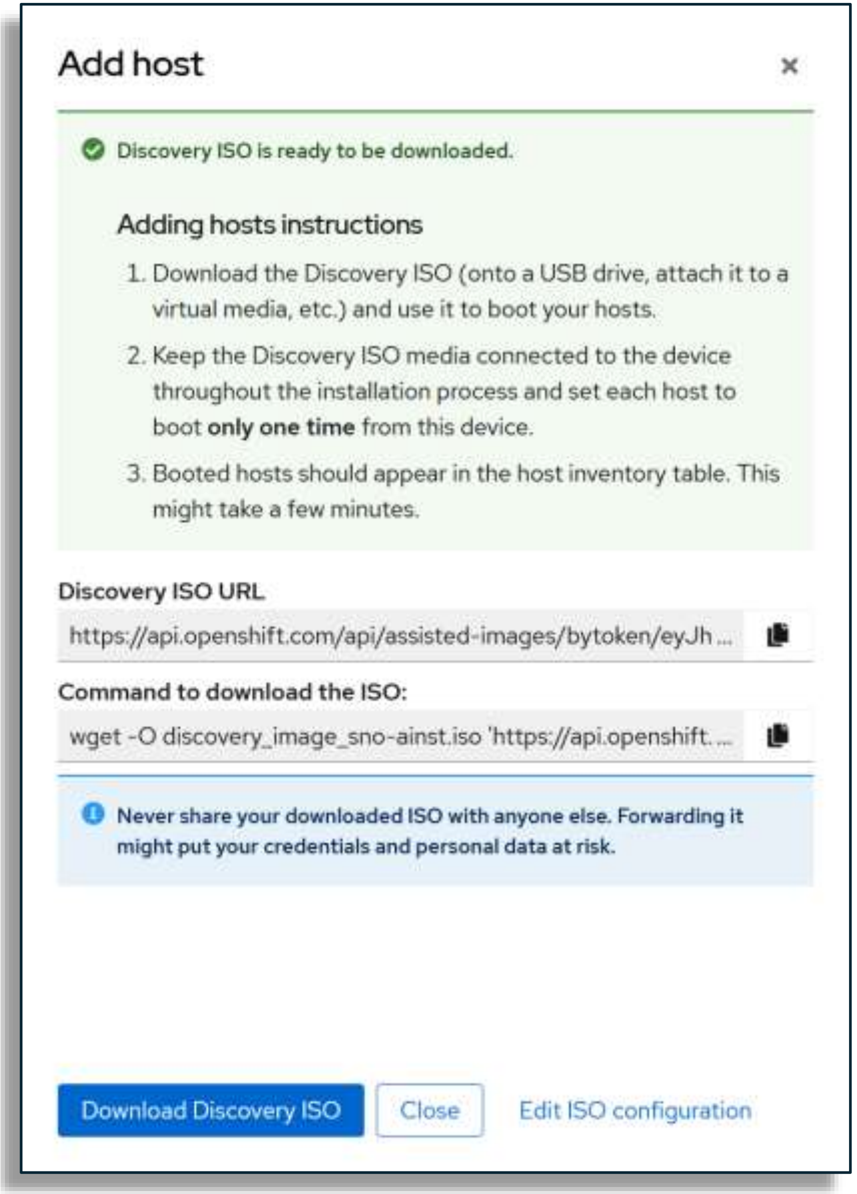
```
# scp 3125f0ee-f1ba-4b49-98f8-b76f8b18a796-discovery.iso padmin@vios:/home/padmin

$ mkvopt -name sno.iso -file /home/padmin/3125f0ee-f1ba-4b49-98f8-b76f8b18a796-discovery.iso

$ loadopt -disk sno.iso -vtd vtopt1
```

Démarrer la VM en SMS et démarrer sur l’ISO

Ouvrir un terminal depuis le terminal de la HMC



```
hscroot@vhm2:~> mkvterm -m S914B --id 9

 Open in progress


 PowerPC Firmware
 Version FW950.B0 (VL950_149)
 SMS (c) Copyright IBM Corp. 2000,2024 All rights reserved.
-------------------------------------------------------------------------------
 Main Menu
 1.   Select Language
 2.   Setup Remote IPL (Initial Program Load)
 3.   I/O Device Information
 4.   Select Console
 5.   Select Boot Options








-------------------------------------------------------------------------------
 Navigation Keys:

                                      X = eXit System Management Services
-------------------------------------------------------------------------------
```



Add host

Discovery ISO is ready to be downloaded.

Adding hosts instructions

1. Download the Discovery ISO (onto a USB drive, attach it to a virtual media, etc.) and use it to boot your hosts.
2. Keep the Discovery ISO media connected to the device throughout the installation process and set each host to boot **only one time** from this device.
3. Booted hosts should appear in the host inventory table. This might take a few minutes.

Discovery ISO URL

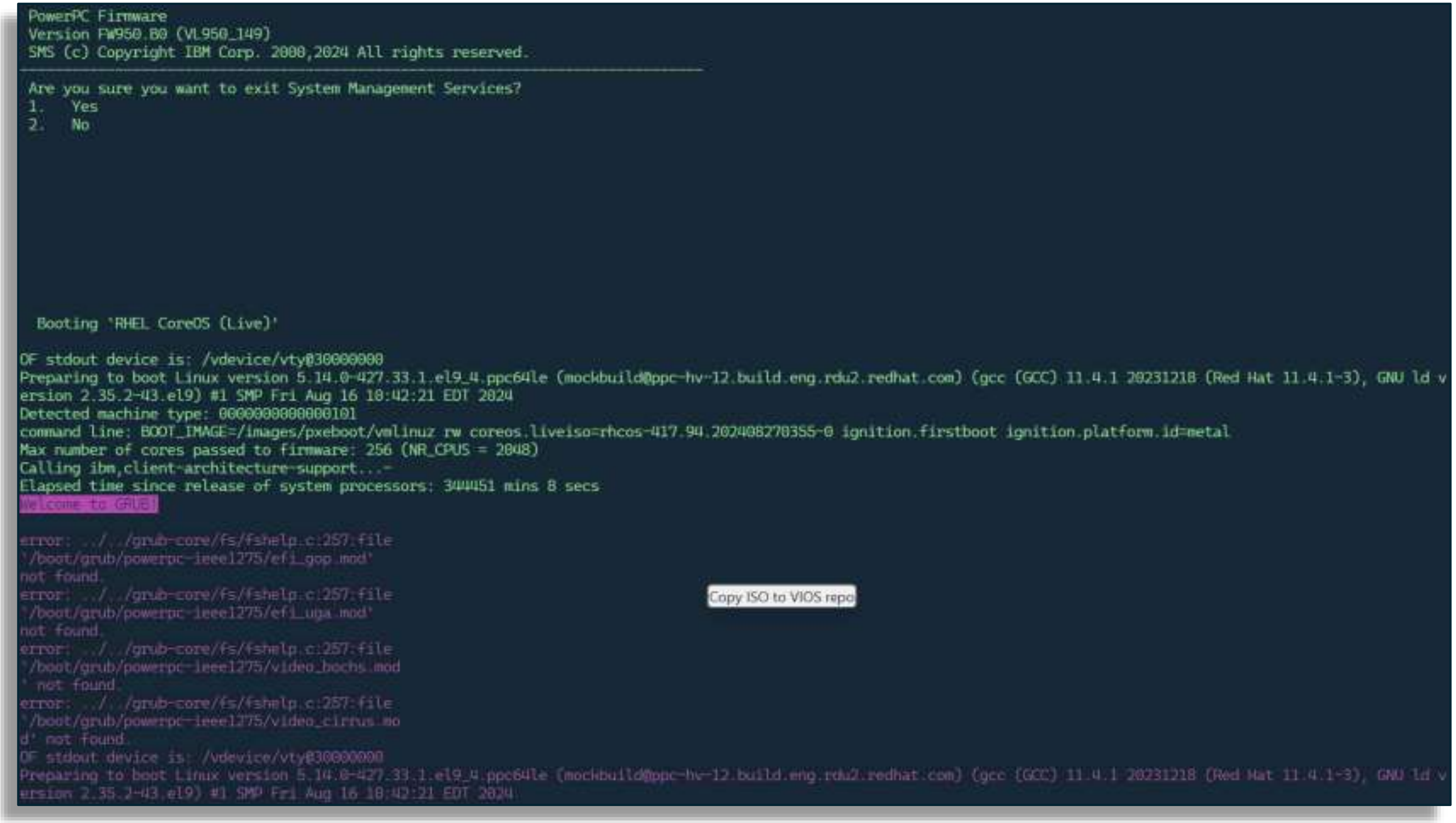
https://api.openshift.com/api/assisted-images/bytoken/eyJh ...

Command to download the ISO:

wget -O discovery_image_sno-ainst.iso 'https://api.openshift. ...

Never share your downloaded ISO with anyone else. Forwarding it might put your credentials and personal data at risk.

Début de l’installation de OpenShift Single Node



```
 PowerPC Firmware
 Version FW950.B0 (VL950_149)
 SMS (c) Copyright IBM Corp. 2000,2024 All rights reserved.
-------------------------------------------------------------------------------
 Are you sure you want to exit System Management Services?
 1.   Yes
 2.   No








  Booting 'RHEL CoreOS (Live)'

OF stdout device is: /vdevice/vty@30000000
Preparing to boot Linux version 5.14.0-427.33.1.el9_4.ppc64le (mockbuild@ppc-hv-12.build.eng.rdu2.redhat.com) (gcc (GCC) 11.4.1 20231218 (Red Hat 11.4.1-3), GNU ld v
ersion 2.35.2-43.el9) #1 SMP Fri Aug 16 10:42:21 EDT 2024
Detected machine type: 0000000000000101
command line: BOOT_IMAGE=/images/pxeboot/vmlinuz rw coreos.liveiso=rhcos-417.94.202408270355-0 ignition.firstboot ignition.platform.id=metal
Max number of cores passed to firmware: 256 (NR_CPUS = 2048)
Calling ibm,client-architecture-support...-
Elapsed time since release of system processors: 344451 mins 8 secs
Welcome to GRUB!

error: ../../grub-core/fs/fshelp.c:257:file
'/boot/grub/powerpc-ieee1275/efi_gop.mod'
not found.
error: ../../grub-core/fs/fshelp.c:257:file
'/boot/grub/powerpc-ieee1275/efi_uga.mod'
not found.
error: ../../grub-core/fs/fshelp.c:257:file
'/boot/grub/powerpc-ieee1275/video_bochs.mod
' not found.
error: ../../grub-core/fs/fshelp.c:257:file
'/boot/grub/powerpc-ieee1275/video_cirrus.mo
d' not found.
OF stdout device is: /vdevice/vty@30000000
Preparing to boot Linux version 5.14.0-427.33.1.el9_4.ppc64le (mockbuild@ppc-hv-12.build.eng.rdu2.redhat.com) (gcc (GCC) 11.4.1 20231218 (Red Hat 11.4.1-3), GNU ld v
ersion 2.35.2-43.el9) #1 SMP Fri Aug 16 10:42:21 EDT 2024
```

Université IBM i

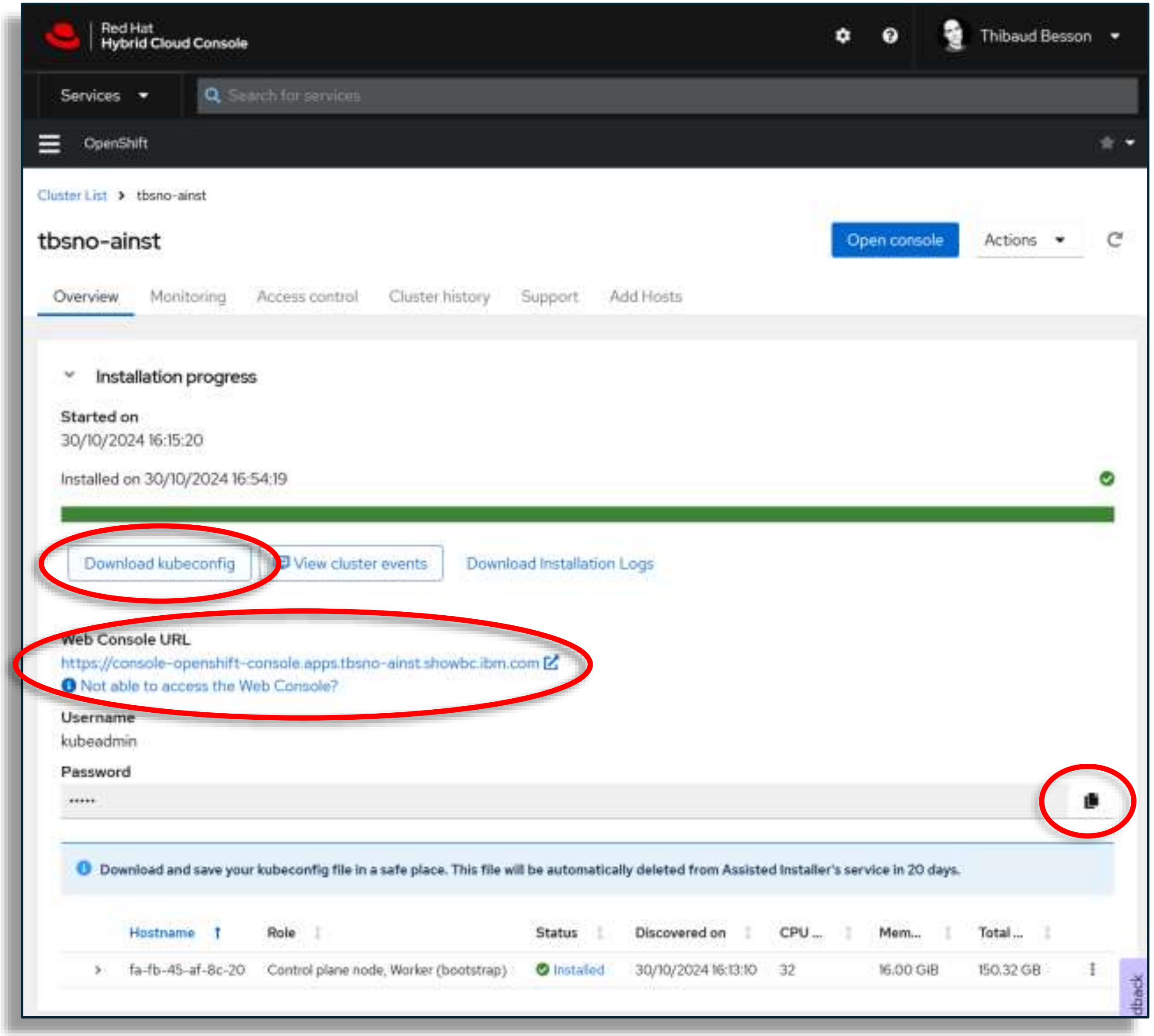
19 et 20 novembre 2024

IBM i
continuous innovation
continuous integration

Configuration Jour 0

Cluster installé – informations de connexion au cluster

- Copier `kubeconfig`
- Copier le mot de passe de l’administrateur du cluster : `kubeadmin`



tbsno-ainst

Installation progress

Started on
30/10/2024 16:15:20

Installed on 30/10/2024 16:54:19

Web Console URL
https://console-openshift-console.apps.tbsno-ainst.showbc.ibm.com

Not able to access the Web Console?

Username
kubeadmin

Password
•••••

Download and save your kubeconfig file in a safe place. This file will be automatically deleted from Assisted Installer's service in 20 days.

Hostname	Role	Status	Discovered on	CPU ...	Mem...	Total ...
fa-fb-45-af-8c-20	Control plane node, Worker (bootstrap)	Installed	30/10/2024 16:13:10	32	16.00 GiB	150.32 GB

Fichier kubeconfig ?

- Fichier YAML contenant les détails pour authentifier le cluster : adresse IP, utilisateur, certificats, etc.
- Nécessaire pour interagir en ligne de commande avec le serveur API du cluster avec le cluster par le client `'oc'`
- Des composants du cluster utilisent le fichier kubeconfig pour interagir avec le serveur API du cluster : controller manager, scheduler and kubelet. On le trouve donc dans CoreOS dans `/var/lib/kubelet/kubeconfig`

https://kubernetes.io/docs/concepts/configuration/organize-cluster-access-kubeconfig/#the-kubeconfig-environment-variable

La variable d’environnement KUBECONFIG

- The KUBECONFIG environment variable holds a list of kubeconfig files. For Linux and Mac, the list is colon-delimited. For Windows, the list is semicolon-delimited.
- The KUBECONFIG environment variable is not required. If the KUBECONFIG environment variable doesn't exist, kubectl uses the default kubeconfig file, $HOME/.kube/config.
- You can have any number of kubeconfig in the .kube directory. Each config will have a unique context name (ie, the name of the cluster).
- You can validate the Kubeconfig file by listing the contexts. You can list all the contexts using the following command. It will list the context name as the name of the cluster.

```
oc config get-contexts -o=name
```

https://docs.openshift.com/container-platform/4.17/cli_reference/openshift_cli/managing-cli-profiles.html

Utiliser la ligne de commande

- https://docs.openshift.com/container-platform/4.17/cli_reference/openshift_cli/getting-started-cli.html
- oc login -u=user1 [--server=https://(...):6443 --insecure-skip-tls-verify=true]

Configurer la résolution DNS

Rappel : la résolution de nom est necessaire pour les composants suivants du cluster OpenShift :

- L’API Kubernetes
- L’adressage des applications déployées dans OpenShift : Ingress route des applications
- Le control plane et les serveurs de compute
- La résolution DNS inverse est aussi requise pour l’API Kubernetes, le control plane et les serveurs de compute.

Configuration DNS - Détails

Component	Record	Description
Kubernetes API	api.<cluster_name>.<base_domain>.	A DNS A/AAAA or CNAME record, and a DNS PTR record, to identify the API load balancer. These records must be resolvable by both clients external to the cluster and from all the nodes within the cluster.
	api-int.<cluster_name>.<base_domain>.	A DNS A/AAAA or CNAME record, and a DNS PTR record, to internally identify the API load balancer. These records must be resolvable from all the nodes within the cluster.
		The API server must be able to resolve the worker nodes by the hostnames that are recorded in Kubernetes. If the API server cannot resolve the node names, then proxied API calls can fail, and you cannot retrieve logs from pods.
Ingress routes : Routes to applications deployed in cluster	*.apps.<cluster_name>.<base_domain>.	A wildcard DNS A/AAAA or CNAME record that refers to the application ingress load balancer. The application ingress load balancer targets the machines that run the Ingress Controller pods. The Ingress Controller pods run on the compute machines by default. These records must be resolvable by both clients external to the cluster and from all the nodes within the cluster. For example, console-openshift-console.apps.<cluster_name>.<base_domain> is used as a wildcard route to the OpenShift Container Platform console.
Control plane machines	<master><n>.<cluster_name>.<base_domain>.	DNS A/AAAA or CNAME records and DNS PTR records to identify each machine for the control plane nodes. These records must be resolvable by the nodes within the cluster.
Compute machines	<worker><n>.<cluster_name>.<base_domain>.	DNS A/AAAA or CNAME records and DNS PTR records to identify each machine for the worker nodes. These records must be resolvable by the nodes within the cluster.

Exemple de DNS Forward Zone

Bastion :
- helper node,
- API,
- ingress routes des applications

```
$TTL 1W
@   IN  SOA  ns1.example.com.  root (
            2019070700  ; serial
            3H      ; refresh (3 hours)
            30M     ; retry (30 minutes)
            2W      ; expiry (2 weeks)
            1W )    ; minimum (1 week)
    IN  NS  ns1.example.com.
    IN  MX 10  smtp.example.com.
;
;
ns1.example.com.                  IN  A  192.168.1.1
smtp.example.com.                 IN  A  192.168.1.5
;
helper.example.com.               IN  A  192.168.1.5
api.ocp4.example.com.             IN  A  192.168.1.5
api-int.ocp4.example.com.         IN  A  192.168.1.5
*.apps.ocp4.example.com.          IN  A  192.168.1.5
;
control-plane0.ocp4.example.com.  IN  A  192.168.1.97
control-plane1.ocp4.example.com.  IN  A  192.168.1.98
control-plane2.ocp4.example.com.  IN  A  192.168.1.99
;
worker0.ocp4.example.com.         IN  A  192.168.1.11
worker1.ocp4.example.com.         IN  A  192.168.1.7
;
;EOF
```

Exemple de DNS Reverse Zone

```
$$TTL 1W
@ IN SOA ns1.example.com. root (
         2019070700 ; serial
         3H         ; refresh (3 hours)
         30M        ; retry (30 minutes)
         2W         ; expiry (2 weeks)
         1W )       ; minimum (1 week)
  IN NS ns1.example.com.
;
;
 5.1.168.192.in-addr.arpa.  IN  PTR   api.ocp4.example.com.
 5.1.168.192.in-addr.arpa.  IN  PTR   api-int.ocp4.example.com.
;
97.1.168.192.in-addr.arpa.  IN  PTR   control-plane0.ocp4.example.com.
98.1.168.192.in-addr.arpa.  IN  PTR   control-plane1.ocp4.example.com.
99.1.168.192.in-addr.arpa.  IN  PTR   control-plane2.ocp4.example.com.
;
11.1.168.192.in-addr.arpa.  IN  PTR   worker0.ocp4.example.com.
 7.1.168.192.in-addr.arpa.  IN  PTR   worker1.ocp4.example.com.
;
;EOF
```

Connection à CoreOS ?

- Si tout va bien, vous n’êtes pas supposé-e vous connecter à CoreOS sur vos nodes !

“One of the interesting things about the new OpenShift is that it suggests not to use SSH directly (you can see this in sshd_config on the nodes because they have PermitRootLogin no set on them). By design, OpenShift 4 clusters are immutable and rely on Operators to apply cluster changes. In turn, this means that accessing the underlying nodes directly by SSH is not the recommended procedure. Additionally, the nodes will be tainted as accessed.”

3 moyens de se connecter à CoreOS :

- Accéder à coreOS à travers le cluster par commande oc : Si besoin (debug) , et si c’est encore possible (cluster fonctionnel) :

```
oc debug node/<node-name>
```

https://www.redhat.com/en/blog/how-oc-debug-works

- Connection par SSH : 2 possibilités théoriques
  1. Besoin du réseau et de la clef SSH publique donnée à l’installation
  2. Besoin du réseau, d’un login / password et de l’autorisation du serveur SSH pour les login avec mot de passe

- Connexion par console virtuelle : HMC vterm, virsh terminal, BMC console, etc. :

  Besoin d’un login / password, donc d’un fichier /etc/shadow dans coreOS avec le hash d’un mot de passe pour l’utilisateur « core »

Connection à CoreOS depuis le cluster

https://www.redhat.com/en/blog/how-oc-debug-works

- Accéder à coreOS à travers le cluster par commande oc :

Si besoin de debug, et si c’est encore possible (cluster fonctionnel)

```
oc debug node/<node-name>
```

La commande démarre un pod à partir d’une image téléchargée sur quay.io, appelé ‘node-name’-debug

Connection à CoreOS par SSH avec clef SSH

Si le node n’est pas connecté au cluster, pas de « oc debug ». Reste SSH.

Pour se connecter par SSH, plusieurs possibilités selon l’état de la configuration SSH de CoreOS :

État par défaut : besoin du réseau et de la clef SSH publique donnée à OpenShift pendant l’installation. Conseil : utiliser une clef « habituelle » sans en créer une spécifique, pour pouvoir vous connecter depuis différentes machines possédant cette clef.

```
$ ssh -i /path/to/privatekey core@[master-hostname]
```

Après une customisation de SSH : Besoin du réseau, d’un login / password et de l’autorisation du serveur SSH pour les login avec mot de passe

Personnaliser les nodes

Ne faites pas de modifications directes dans CoreOS ! Cela créera des problèmes dans la mise à jour ultérieure du cluster.

OpenShift Container Platform supports both cluster-wide and per-machine configuration via Ignition, which allows arbitrary partitioning and file content changes to the operating system.

There are two ways to deploy machine config changes:

- Creating machine configs that are included in manifest files to start up a cluster during openshift-install.
- Creating machine configs that are passed to running OpenShift Container Platform nodes via the Machine Config Operator.

Connection à CoreOS par SSH avec login /password

Ce n’est pas la configuration par défaut de SSH pour CoreOS de OpenShift.

Connexion « traditionnelle » de l’IT :

- Besoin du réseau,
- d’un login / password, non configuré dans CoreOS par défaut,
- de l’autorisation du serveur SSH pour les logins avec mot de passe, désactivée par défaut

Nécessité de personnaliser CoreOS avec un MachineConfig file :
https://access.redhat.com/solutions/7071828

Connexion sans SSH : par une console virtuelle

https://access.redhat.com/solutions/7010657

Créer un password hash avec mkpasswd :

```
$ mkpasswd -m SHA-512 testpasswd
```

Ou bien par OpenSSL : The “-6” flag specifies to use the SHA-512 algorithm.

```
$ openssl passwd -6 testpasswd
```

Créer un fichier machine config file avec l’utilisateur core et le mot de passe hashé :

```
apiVersion: machineconfiguration.openshift.io/v1
kind: MachineConfig
metadata:
  labels:
    machineconfiguration.openshift.io/role: master
  name: set-core-user-password
spec:
  config:
    ignition:
      version: 3.2.0
    passwd:
      users:
      - name: core
        passwordHash: $6$2E1HD6NFB7KsUEUy$Gdd.MdJhWE5V/Rl3.uR/59g05SZc9GKoPhaMSmSHM2s7jPkw8zk5saL310BKgLkYyT8O3ncbZXJQPQGiCs0dD.
```

Why is this important? Scenarios :

1. A new node is failing to join the cluster and ssh/api access is not possible but a local console (via cloud provider or bare metal BMC). The administrator would like to pull logs to triage the joining problem.
2. sshd is not enabled and the API connection to the kubelet is down (so no `oc debug node`) and the administrator needs to triage the problem and/or collect logs.

By default, Red Hat Enterprise Linux CoreOS (RHCOS) creates a user named core on the nodes in your cluster. You can use the core user to access the node through a cloud provider serial console or a bare metal baseboard controller manager (BMC). This can be helpful, for example, if a node is down and you cannot access that node by using SSH or the oc debug node command. However, by default, there is no password for this user, so you cannot log in without creating one.

You can create a password for the core user by using a machine config. The Machine Config Operator (MCO) assigns the password and injects the password into the /etc/shadow file, allowing you to log in with the core user. The MCO does not examine the password hash. As such, the MCO cannot report if there is a problem with the password.

Dans les labels, le rôle doit être master car on voit dans les machine config pools qu’il n’y a pas de nœud worker.

Configurer Login / password sur CoreOS par MCO

1.Create the machine config by running the following command:

```
$ oc create -f <file-name>.yaml
```

The nodes do not reboot and should become available in a few moments. to watch for the machine config pools to be updated:

```
$ oc get mcp
NAME   CONFIG                            UPDATED UPDATING DEGRADED MACHINECOUNT READYMACHINECOUNT UPDATEDMACHINECOUNT
DEGRADEDMACHINECOUNT AGE
master rendered-master-d686a3ffc8fde True    False    False    3 3 3 0 64m
worker rendered-worker-4605605a5b1f9 False   True     False    3 0 0 0 64m
```

Vérification

1.After the nodes return to the UPDATED=True state, start a debug session for a node:

```
$ oc debug node/<node_name>
```

2.Set `/host` as the root directory within the debug shell by running the following command:

```
sh-4.4# chroot /host
```

3.Check the contents of the /etc/shadow file

```
...
core:$6$2sE/010goDuRSxxv$o18K52wor.wIwZp:19418:0:99999:7:::
...
```

The hashed password is assigned to the core user.

Une fois connecté à CoreOS, comment passer des commandes ‘oc’ ?

- oc command fails when it is run from cluster nodes.

```
sh-4.4# oc get pod -n openshift-monitoring
error: Missing or incomplete configuration info. Please
point to an existing, complete config file
  1. Via the command-line flag --kubeconfig
  2. Via the KUBECONFIG environment variable
  3. In your home directory as ~/.kube/config
```

- Add --kubeconfig=/var/lib/kubelet/kubeconfig option to the oc command.

```
sh-4.4# oc get pod -n openshift-monitoring --
kubeconfig=/var/lib/kubelet/kubeconfig
```

Diagnostic Steps

No clusters or contexts information for oc command by default on cluster nodes :

```
sh-4.4# oc config view
apiVersion: v1
clusters: null
contexts: null
current-context: ""
kind: Config
preferences: {}
users: null
```

Université IBM i

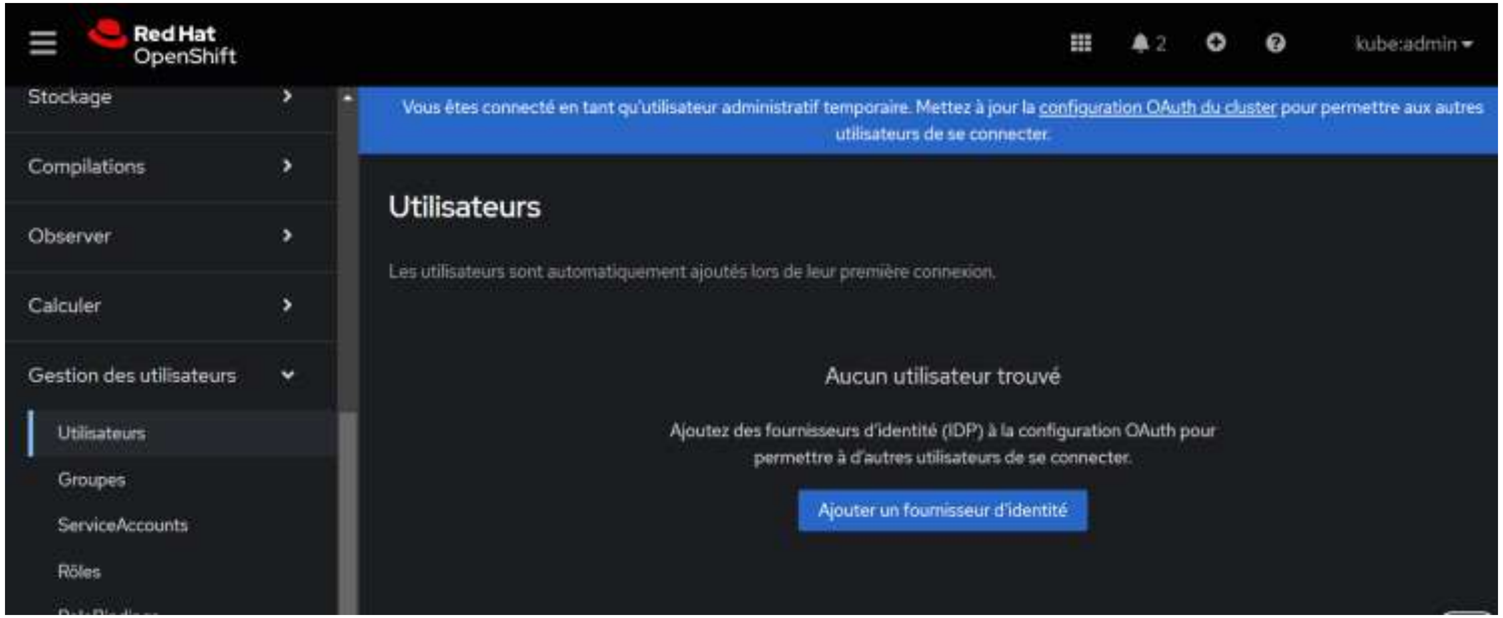
19 et 20 novembre 2024

IBM i
continuous innovation
continuous integration

Gestion des utilisateurs

Authentification et autorisation **OAuth server**

- Le control plane d’OpenShift inclut un serveur OAuth (Open Authorization)qui détermine l’identité d’un utilisateur à partir d’un fournisseur d’identité, puis qui génère un jeton d’accès (access token).
- OAuth est un protocole d’autorisation permettant de profiler une connexion sécurisée, en utilisant des jetons d’encodage sans état pour sécuriser les sessions des utilisateurs sur une application web. Il permet à un utilisateur d'autoriser une application tierce à accéder à ses données sans partager son mot de passe.
- OAuth travaille avec des fournisseurs d’identité, qui gèrent les authentifications. OAuth gère l’autorisation des permissions par l’utilisateur, ainsi que les serveurs du fournisseur d’identité.



Vous êtes connecté en tant qu'utilisateur administratif temporaire. Mettez à jour la configuration OAuth du cluster pour permettre aux autres utilisateurs de se connecter.

Utilisateurs

Les utilisateurs sont automatiquement ajoutés lors de leur première connexion.

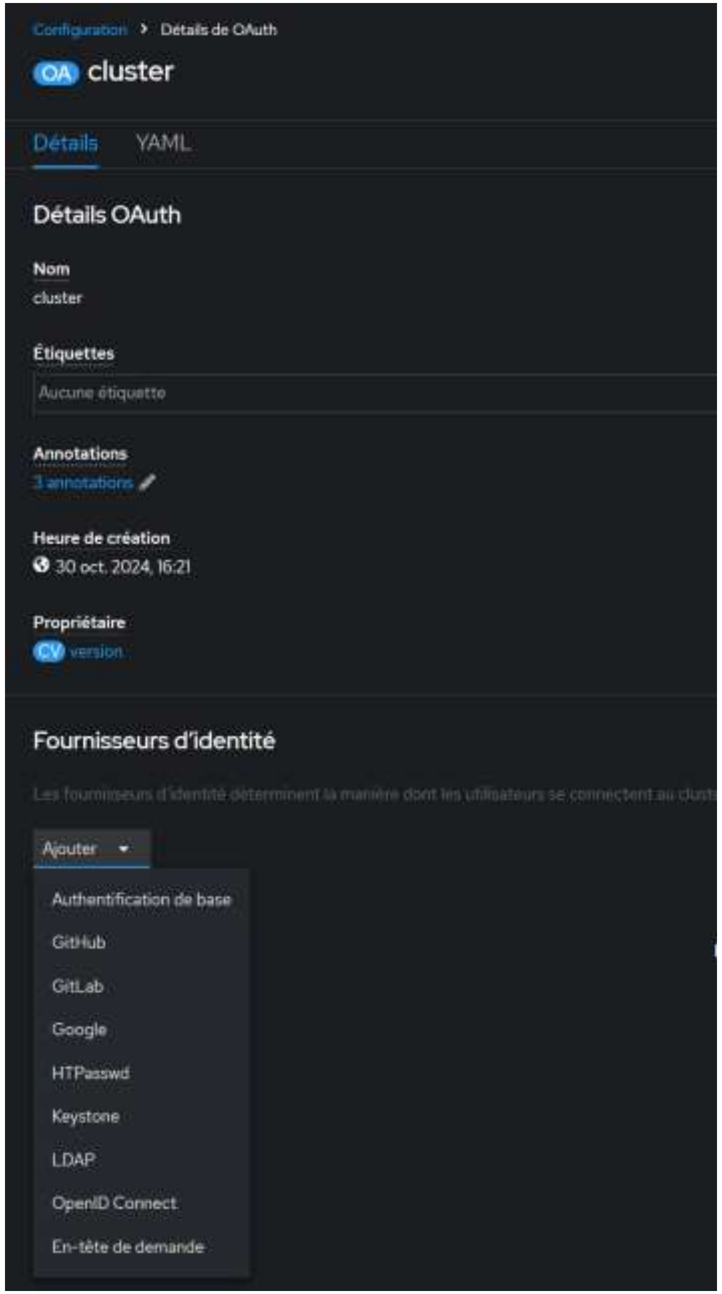
Aucun utilisateur trouvé

Ajoutez des fournisseurs d'identité (IDP) à la configuration OAuth pour permettre à d'autres utilisateurs de se connecter.

Fournisseurs d’identité

OpenShift / Kubernetes propose de nombreuses méthodes d’authentification des utilisateurs.

HTPasswd est simple à mettre en œuvre.



Détails OAuth

Nom
cluster

Étiquettes
Aucune étiquette

Annotations
3 annotations

Heure de création
30 oct. 2024, 16:21

Propriétaire
CV version

Fournisseurs d’identité

Les fournisseurs d’identité déterminent la manière dont les utilisateurs se connectent au clust

HTPasswd fournisseur d’identité

htpasswd est utilisé pour créer et mettre à jour le fichier texte qui stocke les noms et mot de passe des utilisateurs d’un serveur HTTP.

- Installation

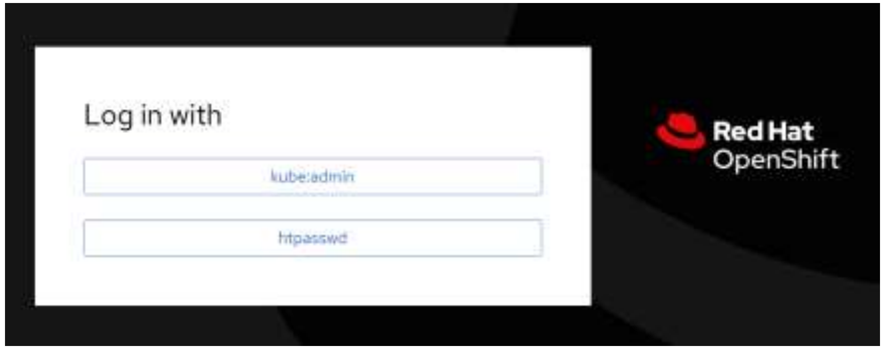
```
$ dnf install httpd-tools
```

- Création du fichier :

```
$ htpasswd -c -B -b users.htpasswd thibaud <mot-de-passe>
Adding password for user thibaud
```

- Vérification

```
$ cat users.htpasswd
thibaud:$2y$05$/4a7FcULXfDB0lj8AyUwfOXq5AenJeVMpbQ3zmfpFYB
76KTf2vznO
```

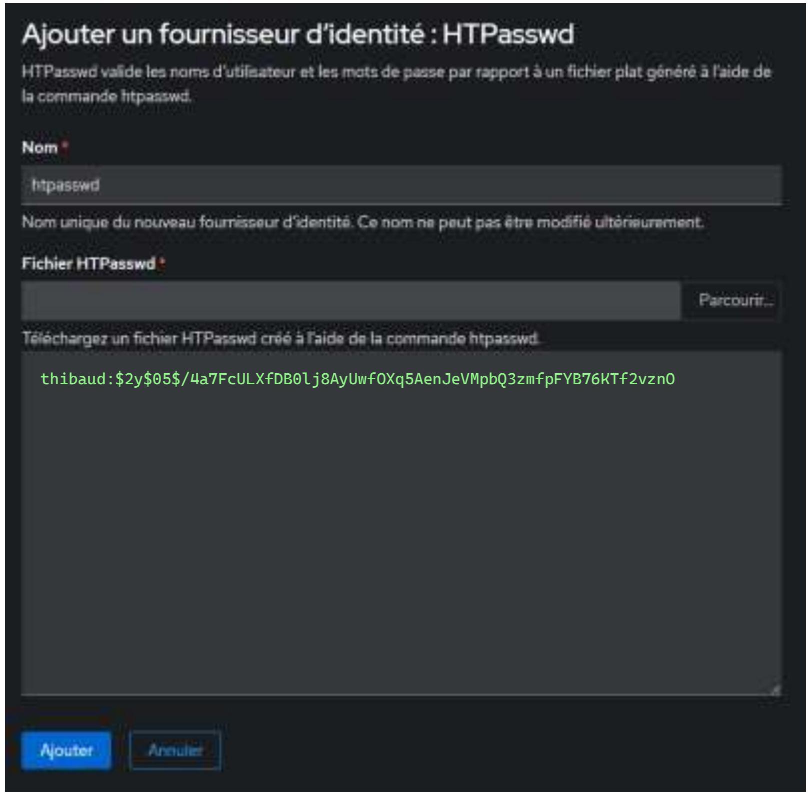


Commande htpasswd :

- Installation dans Red Hat Linux :

```
$ sudo dnf install httpd-tools
```

- Installation dans Cygwin : Package httpd-tools



Ajouter un fournisseur d’identité : HTPasswd

HTPasswd valide les noms d’utilisateur et les mots de passe par rapport à un fichier plat généré à l’aide de la commande htpasswd.

Nom *

htpasswd

Nom unique du nouveau fournisseur d’identité. Ce nom ne peut pas être modifié ultérieurement.

Fichier HTPasswd *

Téléchargez un fichier HTPasswd créé à l’aide de la commande htpasswd.

```
thibaud:$2y$05$/4a7FcULXfDB0lj8AyUwfOXq5AenJeVMpbQ3zmfpFYB76KTf2vznO
```

Configurer HTPasswd

Create a Secret object that contains the htpasswd users file:

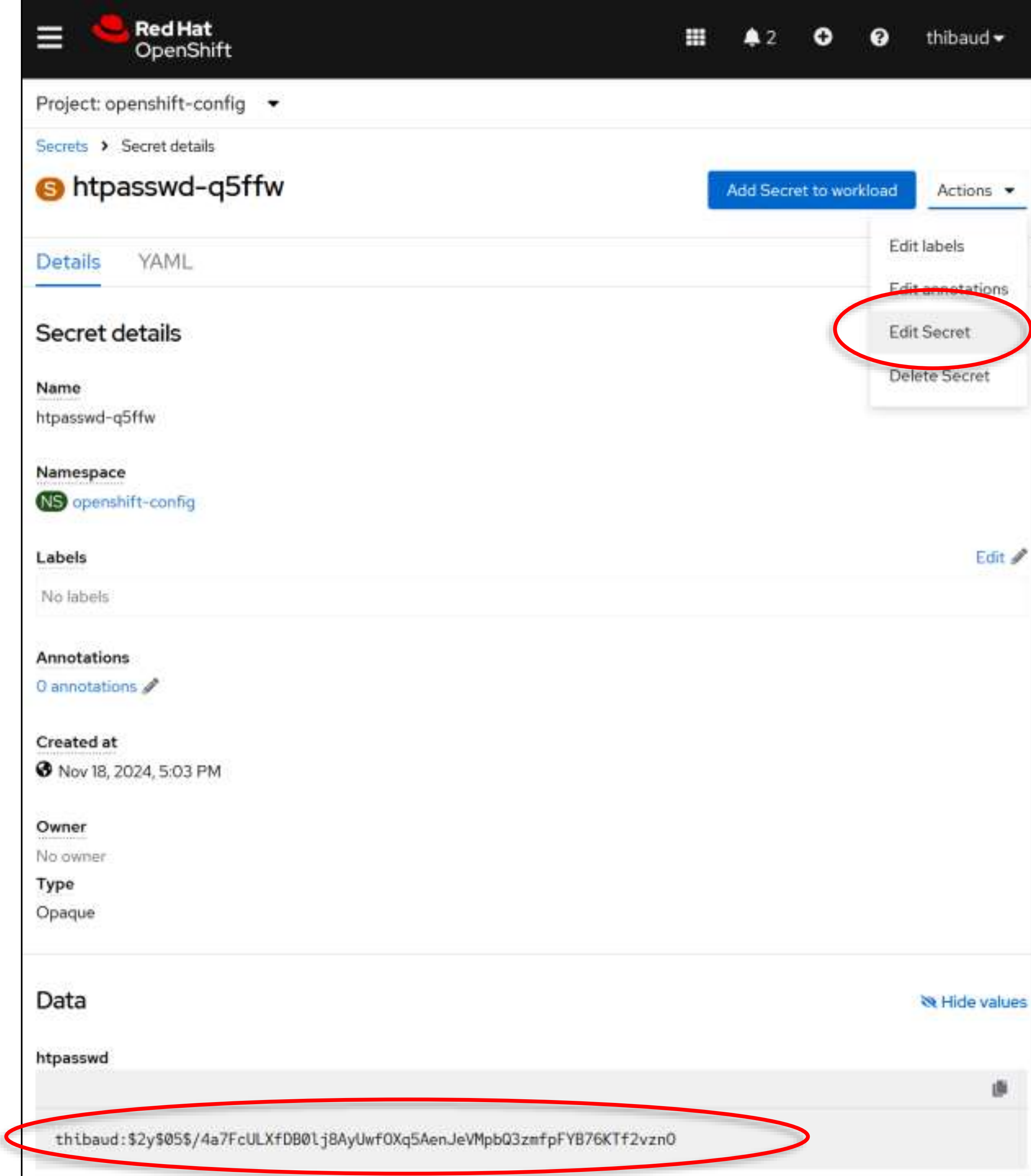
```
$ oc create secret generic htpass-secret --from-file=htpasswd=<path_to_users.htpasswd> -n
openshift-config
```

Ou appliquer le YAML suivant :

```
apiVersion: v1
kind: Secret
metadata:
  name: htpass-secret
  namespace: openshift-config
type: Opaque
data:
  htpasswd: <base64_encoded_htpasswd_file_contents>
```

Modifier un mot de pase

- En général, les mots de passe dans OpenShift sont stockés dans des « secrets »
- Éditez le secret htpasswd-xxx
- Mettre le hash du nouveau mot de passe



Secret details

Name
htpasswd-q5ffw

Namespace
NS openshift-config

Labels
Edit
No labels

Annotations
0 annotations

Created at
Nov 18, 2024, 5:03 PM

Owner
No owner

Type
Opaque

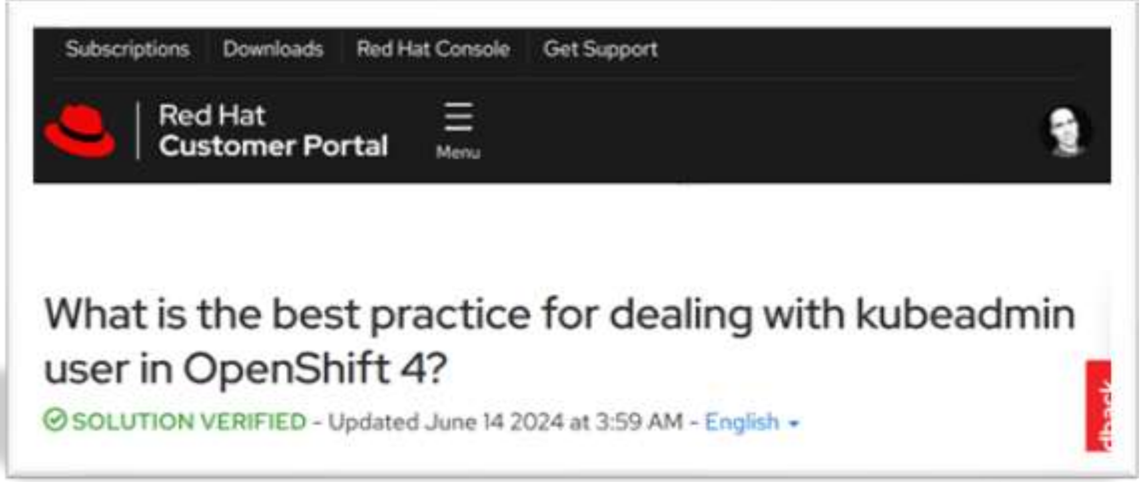
Data
Hide values

htpasswd

```
thibaud:$2y$05$/4a7FcULXfDB0lj8AyUwfOXq5AenJeVMpbQ3zmfpFYB76KTf2vznO
```

Que faire avec kubeadmin ?

- L’utilisateur kubeadmin est créé à l’installation du cluster
- Son mot de passe ne peut pas être changé (pas facilement). Dommage… `2LxtK-8k736-ipMfQ-ubmLU`
- Créer un nouvel utilisateur et élever ses privilèges à ceux du ClusterRole `cluster-admin`
- Vérifier les droits puis supprimer kubeadmin (attention…)



What is the best practice for dealing with kubeadmin user in OpenShift 4?

SOLUTION VERIFIED - Updated June 14 2024 at 3:59 AM - English

https://access.redhat.com/solutions/5309141

Ajouter le rôle d’administrateur

```
$ oc adm policy add-cluster-role-to-user cluster-admin thibaud
clusterrole.rbac.authorization.k8s.io/cluster-admin added: "thibaud"
```

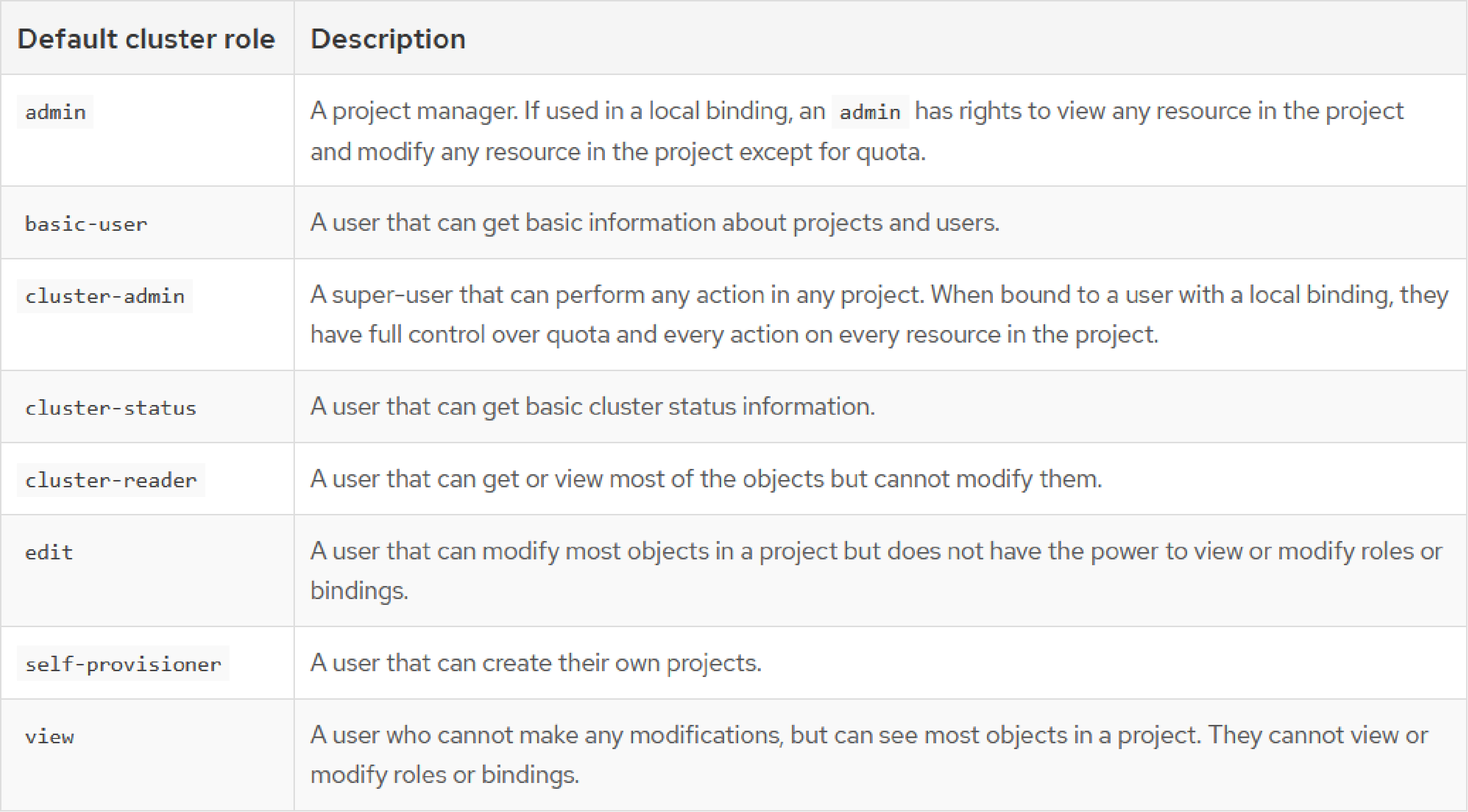
Vérification

```
$ oc get clusterrolebinding -o yaml | grep -A 1 -B 15
thibaud
    name: thanos-querier
    resourceVersion: "13941"
    uid: 18416b07-5d90-4942-b430-92a4a4fc4cf2
  roleRef:
    apiGroup: rbac.authorization.k8s.io
    kind: ClusterRole
    name: thanos-querier
  subjects:
  - kind: ServiceAccount
    name: thanos-querier
    namespace: openshift-monitoring
- apiVersion: rbac.authorization.k8s.io/v1
  kind: ClusterRoleBinding
  metadata:
    creationTimestamp: "2024-11-18T16:20:51Z"
    name: thibaud-admin
    resourceVersion: "4308602"
    uid: b3dd02c1-cf02-4eff-9ea2-d4eb2e4f39db
  roleRef:
    apiGroup: rbac.authorization.k8s.io
    kind: ClusterRole
    name: sudoer
  subjects:
  - apiGroup: rbac.authorization.k8s.io
    kind: User
    name: thibaud
```

```
$ oc get clusterrolebinding -o yaml | grep -A 1 -B 15 thibaud
name: system:masters
- apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
creationTimestamp: "2024-11-19T10:32:03Z"
name: cluster-admin-0
resourceVersion: "4357882"
uid: 64d1be80-3471-4467-abd1-3110a18481d0
roleRef:
apiGroup: rbac.authorization.k8s.io
kind: ClusterRole
name: cluster-admin
subjects:
- apiGroup: rbac.authorization.k8s.io
kind: User
name: thibaud
- apiVersion: rbac.authorization.k8s.io/v1
--
name: thanos-querier
resourceVersion: "13941"
uid: 18416b07-5d90-4942-b430-92a4a4fc4cf2
roleRef:
apiGroup: rbac.authorization.k8s.io
kind: ClusterRole
name: thanos-querier
subjects:
- kind: ServiceAccount
```

Rôles disponibles par défaut

https://docs.openshift.com/container-platform/4.17/authentication/using-rbac.html



Default cluster role	Description
`admin`	A project manager. If used in a local binding, an `admin` has rights to view any resource in the project and modify any resource in the project except for quota.
`basic-user`	A user that can get basic information about projects and users.
`cluster-admin`	A super-user that can perform any action in any project. When bound to a user with a local binding, they have full control over quota and every action on every resource in the project.
`cluster-status`	A user that can get basic cluster status information.
`cluster-reader`	A user that can get or view most of the objects but cannot modify them.
`edit`	A user that can modify most objects in a project but does not have the power to view or modify roles or bindings.
`self-provisioner`	A user that can create their own projects.
`view`	A user who cannot make any modifications, but can see most objects in a project. They cannot view or modify roles or bindings.

- OCP only contains two roles : "cluster-admin" and "admin"
- "cluster-admins" is a cluster-role-binding name which binds "USER:system:admin/GROUP:system:cluster-admins" and "ROLE:clusteradmin" , so that is not a real role. You can treat it as a relationship between role and user/group.
- "cluster-admin" is a constrained role that has the power to do many things inside of their project, but cannot affect (or destroy) the entire cluster. The scope of usage must be limited.
- The role "admin" is a power role can let the user has edit rights within the project and can change the project's membership. If you need just a user who administrates all projects, it is better to grant "admin" role to them.
- OCP uses RBAC to manage user permissions, the basic unit is rules and policies , then we can define a role binding user/group and multiple polices. So another consider is to create a custom role base on your detailed requirement. For more info about this, you can refer to this link

Université IBM i

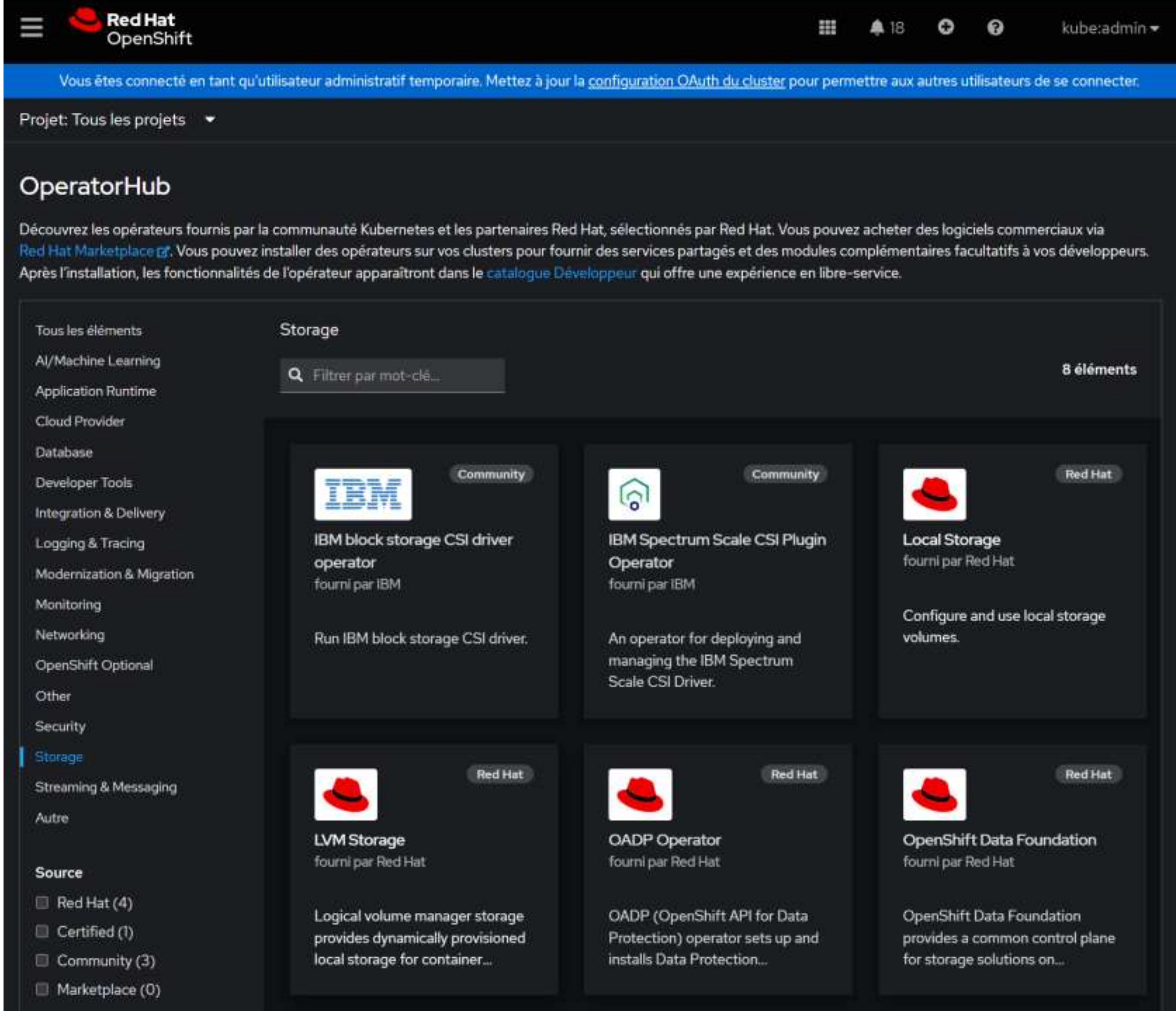
19 et 20 novembre 2024

IBM i
continuous innovation
continuous integration

Gestion du stockage

Opérateurs stockage

- Disponibles par défaut à l’installation d’OpenShift
- D’autres opérateurs peuvent être installés en s’abonnant à d’autres sources.



Vous êtes connecté en tant qu’utilisateur administratif temporaire. Mettez à jour la configuration OAuth du cluster pour permettre aux autres utilisateurs de se connecter.

OperatorHub

Découvrez les opérateurs fournis par la communauté Kubernetes et les partenaires Red Hat, sélectionnés par Red Hat. Vous pouvez acheter des logiciels commerciaux via Red Hat Marketplace. Vous pouvez installer des opérateurs sur vos clusters pour fournir des services partagés et des modules complémentaires facultatifs à vos développeurs. Après l’installation, les fonctionnalités de l’opérateur apparaîtront dans le catalogue Développeur qui offre une expérience en libre-service.

Storage

IBM block storage CSI driver operator
fourni par IBM
Run IBM block storage CSI driver.

IBM Spectrum Scale CSI Plugin Operator
fourni par IBM
An operator for deploying and managing the IBM Spectrum Scale CSI Driver.

Local Storage
fourni par Red Hat
Configure and use local storage volumes.

LVM Storage
fourni par Red Hat
Logical volume manager storage provides dynamically provisioned local storage for container...

OADP Operator
fourni par Red Hat
OADP (OpenShift API for Data Protection) operator sets up and installs Data Protection...

OpenShift Data Foundation
fourni par Red Hat
OpenShift Data Foundation provides a common control plane for storage solutions on...

Prérequis de l’opérateur LVM storage

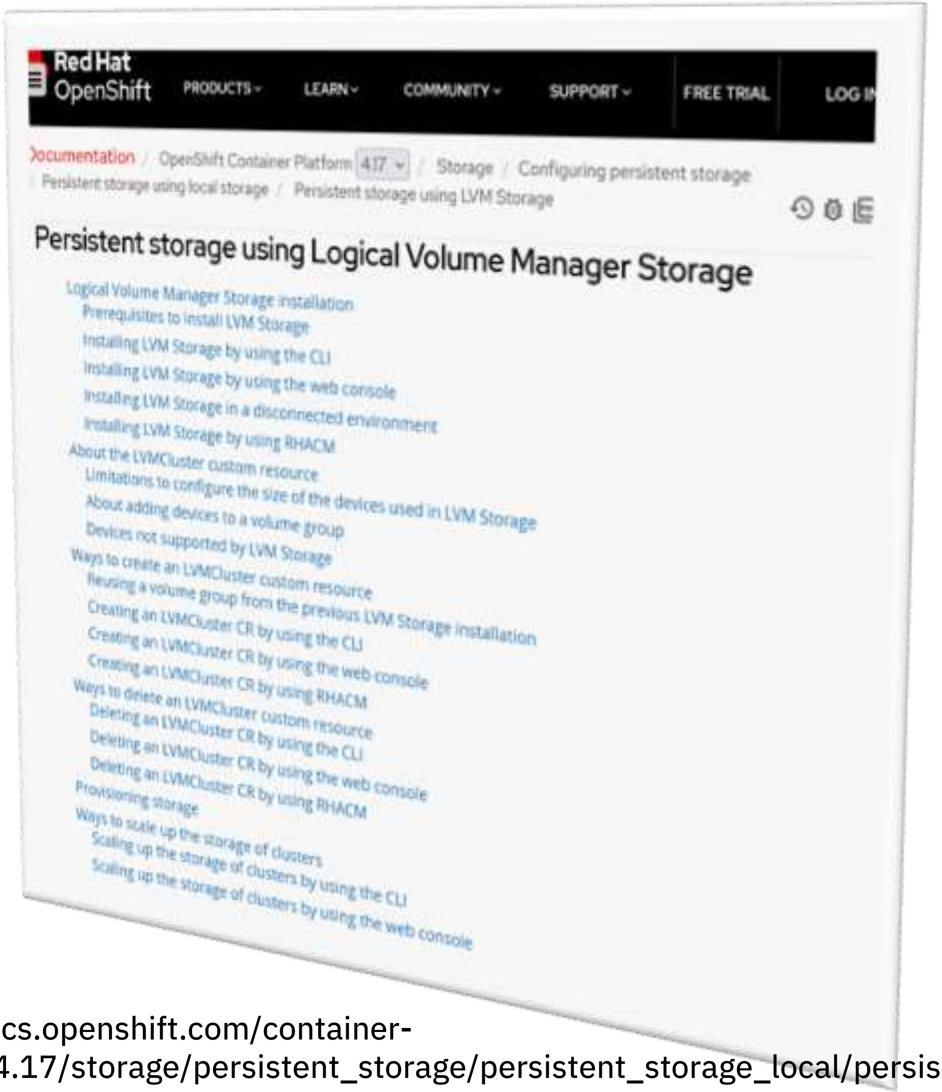
https://github.com/openshift/lvm-operator

https://docs.openshift.com/container-platform/4.17/storage/persistent_storage/persistent_storage_local/persistent-storage-using-lvms.html

- Un peu de CPU et de RAM : au moins 10 milliCPU et 100 Mio de RAM.

- Disque dédié sur le node. LVM Storage utilise uniquement les disques vides et ne contenant pas de signatures de système de fichiers. Effacez les disques avant de les utiliser.

Documentation de l’opérateur



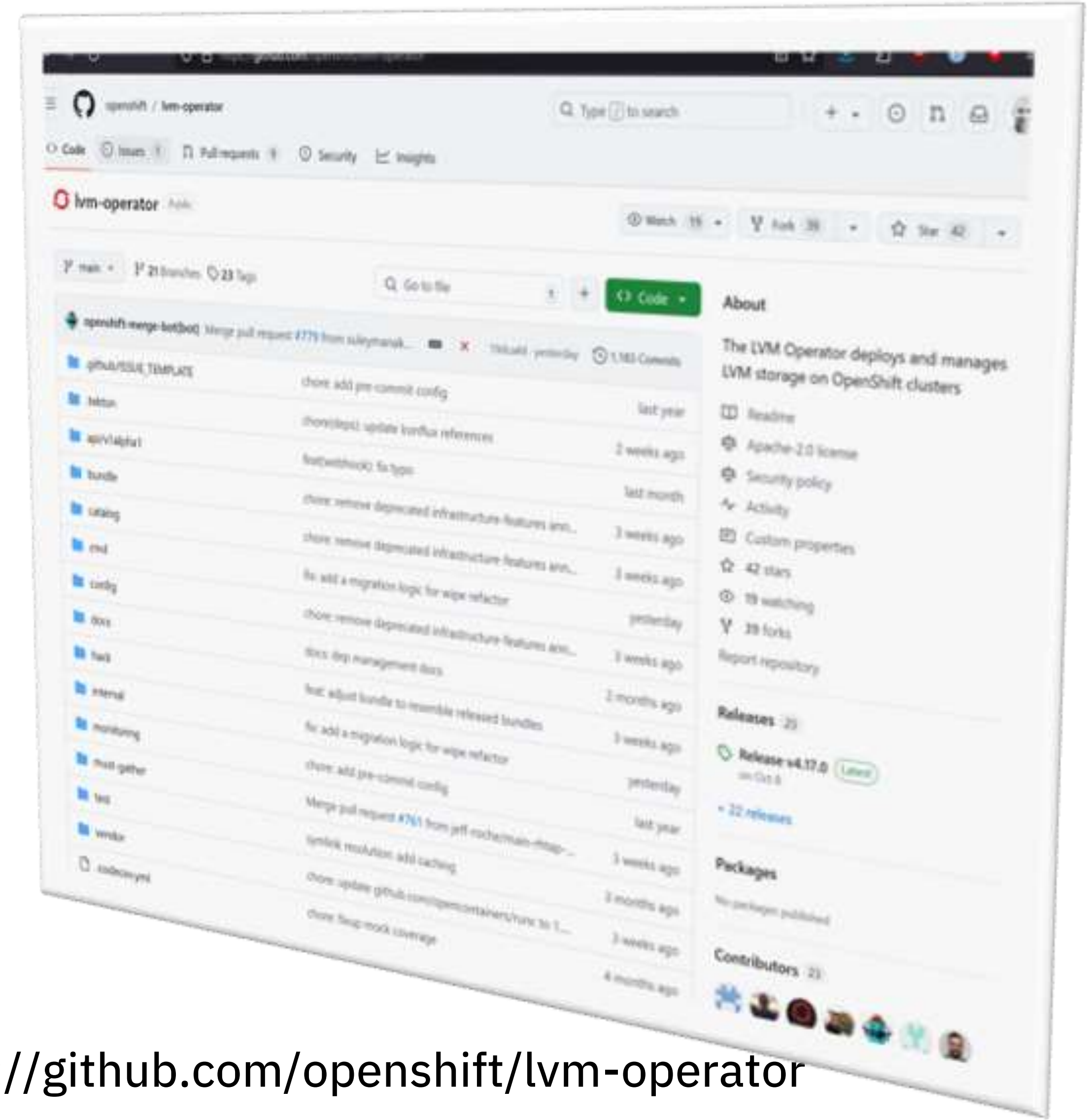
Red Hat OpenShift

Documentation / OpenShift Container Platform 4.17 / Storage / Configuring persistent storage / Persistent storage using local storage / Persistent storage using LVM Storage

Persistent storage using Logical Volume Manager Storage

- Logical Volume Manager Storage installation
  - Prerequisites to install LVM Storage
  - Installing LVM Storage by using the CLI
  - Installing LVM Storage by using the web console
  - Installing LVM Storage in a disconnected environment
  - Installing LVM Storage by using RHACM
- About the LVMCluster custom resource
  - Limitations to configure the size of the devices used in LVM Storage
  - About adding devices to a volume group
  - Devices not supported by LVM Storage
- Ways to create an LVMCluster custom resource
  - Reusing a volume group from the previous LVM Storage installation
  - Creating an LVMCluster CR by using the CLI
  - Creating an LVMCluster CR by using the web console
  - Creating an LVMCluster CR by using RHACM
- Ways to delete an LVMCluster custom resource
  - Deleting an LVMCluster CR by using the CLI
  - Deleting an LVMCluster CR by using the web console
  - Deleting an LVMCluster CR by using RHACM
- Provisioning storage
- Ways to scale up the storage of clusters
  - Scaling up the storage of clusters by using the CLI
  - Scaling up the storage of clusters by using the web console

https://docs.openshift.com/container-platform/4.17/storage/persistent_storage/persistent_storage_local/persistent-storage-using-lvms.html



openshift / lvm-operator

lvm-operator Public

openshift-merge-bot[bot] Merge pull request #779 from suleymanak... ✕ 1,183 Commits

.github/ISSUE_TEMPLATE	chore: add pre-commit config	last year
.tekton	chore(deps): update konflux references	2 weeks ago
api/v1alpha1	feat(webhook): fix typo	last month
bundle	chore: remove deprecated infrastructure features ann...	3 weeks ago
catalog	chore: remove deprecated infrastructure features ann...	3 weeks ago
cmd	fix: add a migration logic for wipe refactor	yesterday
config	chore: remove deprecated infrastructure features ann...	3 weeks ago
docs	docs: dep management docs	2 months ago
hack	feat: adjust bundle to resemble released bundles	3 weeks ago
internal	fix: add a migration logic for wipe refactor	yesterday
monitoring	chore: add pre-commit config	last year
must-gather	Merge pull request #761 from jeff-roche/main-...	3 weeks ago
test	symlink resolution: add caching	3 months ago
vendor	chore: update github.com/opencontainers/runc to 1...	3 weeks ago
.codecov.yml	chore: fixup mock coverage	4 months ago

About

The LVM Operator deploys and manages LVM storage on OpenShift clusters

Releases 23

Release v4.17.0 Latest on Oct 8

+ 22 releases

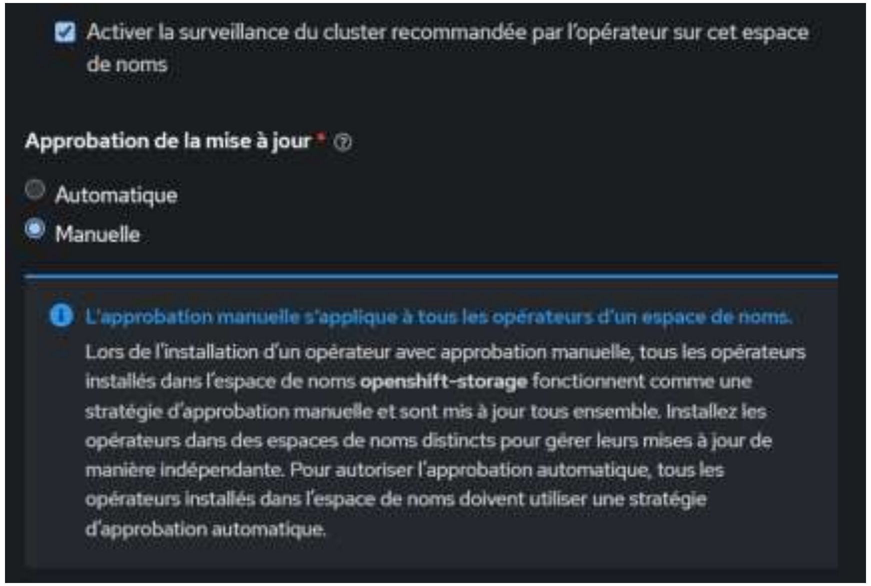
Packages

No packages published

Contributors 23

https://github.com/openshift/lvm-operator

Installation de l’opérateur



☑ Activer la surveillance du cluster recommandée par l’opérateur sur cet espace de noms

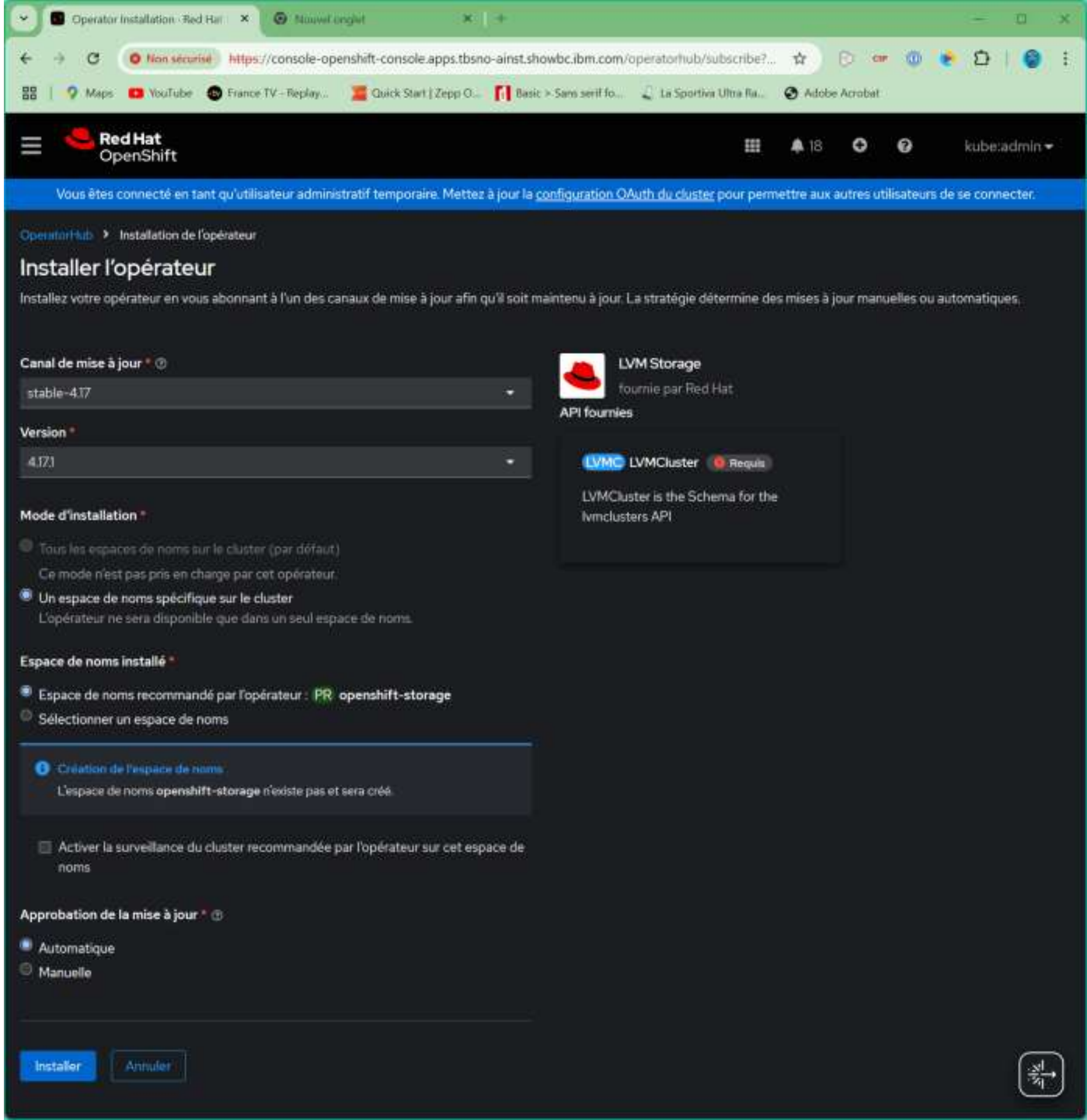
Approbation de la mise à jour * ⓘ

○ Automatique

● Manuelle

L’approbation manuelle s’applique à tous les opérateurs d’un espace de noms.

Lors de l’installation d’un opérateur avec approbation manuelle, tous les opérateurs installés dans l’espace de noms **openshift-storage** fonctionnent comme une stratégie d’approbation manuelle et sont mis à jour tous ensemble. Installez les opérateurs dans des espaces de noms distincts pour gérer leurs mises à jour de manière indépendante. Pour autoriser l’approbation automatique, tous les opérateurs installés dans l’espace de noms doivent utiliser une stratégie d’approbation automatique.



https://console-openshift-console.apps.tbsno-ainst.showbc.ibm.com/operatorhub/subscribe?...

Vous êtes connecté en tant qu'utilisateur administratif temporaire. Mettez à jour la configuration OAuth du cluster pour permettre aux autres utilisateurs de se connecter.

OperatorHub > Installation de l'opérateur

Installer l'opérateur

Installez votre opérateur en vous abonnant à l'un des canaux de mise à jour afin qu'il soit maintenu à jour. La stratégie détermine des mises à jour manuelles ou automatiques.

Canal de mise à jour * ⓘ

stable-4.17

Version *

4.17.1

Mode d'installation *

○ Tous les espaces de noms sur le cluster (par défaut)
Ce mode n'est pas pris en charge par cet opérateur.

● Un espace de noms spécifique sur le cluster
L'opérateur ne sera disponible que dans un seul espace de noms.

Espace de noms installé *

● Espace de noms recommandé par l'opérateur : PR **openshift-storage**

○ Sélectionner un espace de noms

Création de l'espace de noms
L'espace de noms **openshift-storage** n'existe pas et sera créé.

☐ Activer la surveillance du cluster recommandée par l'opérateur sur cet espace de noms

Approbation de la mise à jour * ⓘ

● Automatique

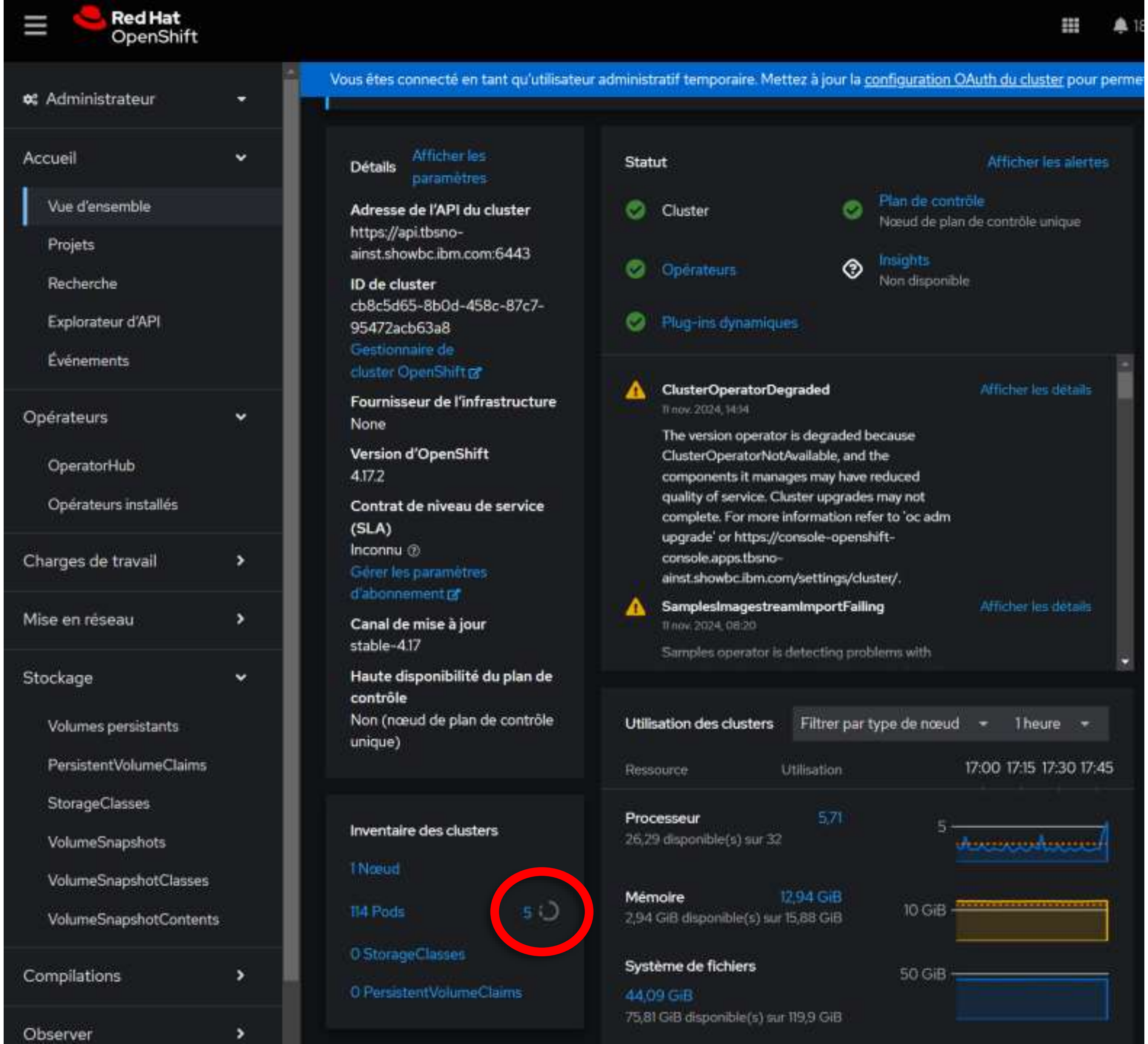
○ Manuelle

Installer | Annuler

LVM Storage
fournie par Red Hat

API fournies

LVMC LVMCluster Requis
LVMCluster is the Schema for the lvmclusters API



Vous êtes connecté en tant qu'utilisateur administratif temporaire. Mettez à jour la configuration OAuth du cluster pour perme

Détails Afficher les paramètres

Adresse de l'API du cluster
https://api.tbsno-ainst.showbc.ibm.com:6443

ID de cluster
cb8c5d65-8b0d-458c-87c7-95472acb63a8
Gestionnaire de cluster OpenShift

Fournisseur de l'infrastructure
None

Version d'OpenShift
4.17.2

Contrat de niveau de service (SLA)
Inconnu
Gérer les paramètres d'abonnement

Canal de mise à jour
stable-4.17

Haute disponibilité du plan de contrôle
Non (nœud de plan de contrôle unique)

Statut Afficher les alertes

- ✓ Cluster
- ✓ Plan de contrôle – Nœud de plan de contrôle unique
- ✓ Opérateurs
- Insights – Non disponible
- ✓ Plug-ins dynamiques

⚠ **ClusterOperatorDegraded** Afficher les détails
11 nov. 2024, 14:14
The version operator is degraded because ClusterOperatorNotAvailable, and the components it manages may have reduced quality of service. Cluster upgrades may not complete. For more information refer to 'oc adm upgrade' or https://console-openshift-console.apps.tbsno-ainst.showbc.ibm.com/settings/cluster/.

⚠ **SamplesImagestreamImportFailing** Afficher les détails
11 nov. 2024, 08:20
Samples operator is detecting problems with

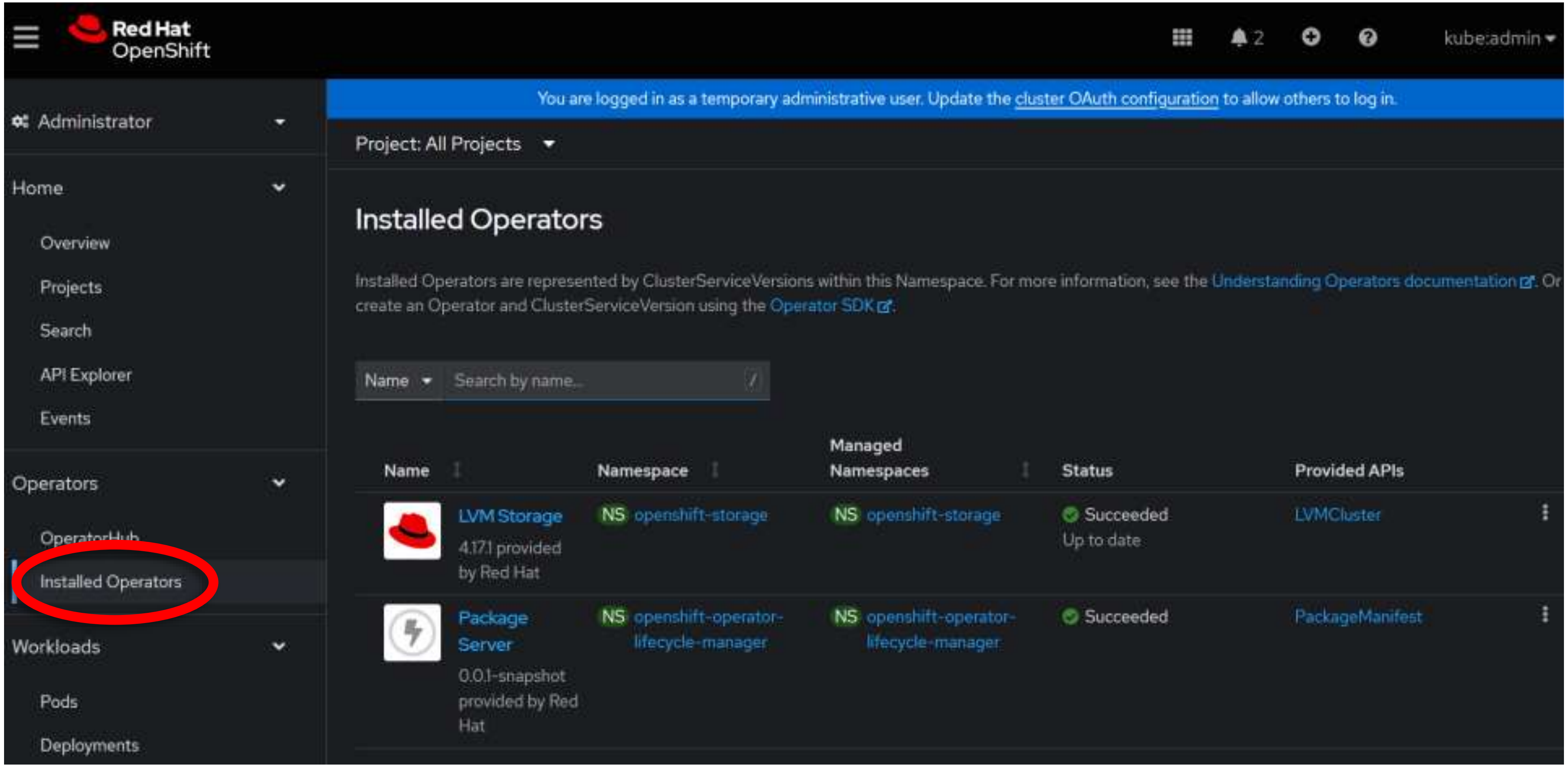
Inventaire des clusters

- 1 Nœud
- 114 Pods 5
- 0 StorageClasses
- 0 PersistentVolumeClaims

Utilisation des clusters Filtrer par type de nœud 1 heure

Ressource	Utilisation
Processeur 26,29 disponible(s) sur 32	5,71
Mémoire 2,94 GiB disponible(s) sur 15,88 GiB	12,94 GiB
Système de fichiers 75,81 GiB disponible(s) sur 119,9 GiB	44,09 GiB

Opérateur installé



You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.

Project: All Projects

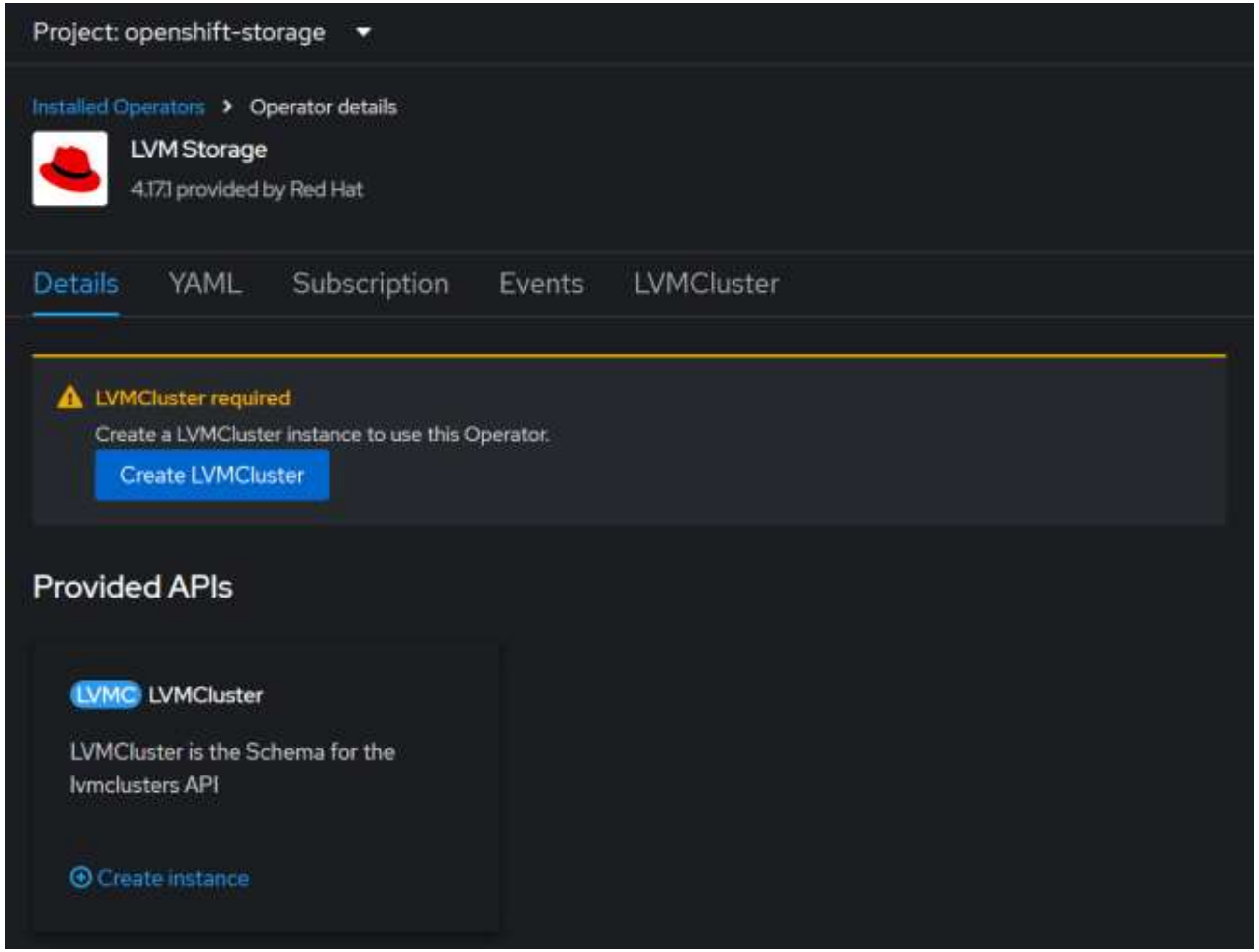
Installed Operators

Installed Operators are represented by ClusterServiceVersions within this Namespace. For more information, see the Understanding Operators documentation. Or create an Operator and ClusterServiceVersion using the Operator SDK.

Name ▾ Search by name... /

Name	Namespace	Managed Namespaces	Status	Provided APIs
LVM Storage 4.17.1 provided by Red Hat	NS openshift-storage	NS openshift-storage	✓ Succeeded Up to date	LVMCluster
Package Server 0.0.1-snapshot provided by Red Hat	NS openshift-operator-lifecycle-manager	NS openshift-operator-lifecycle-manager	✓ Succeeded	PackageManifest

Créer une instance de LVMCluster

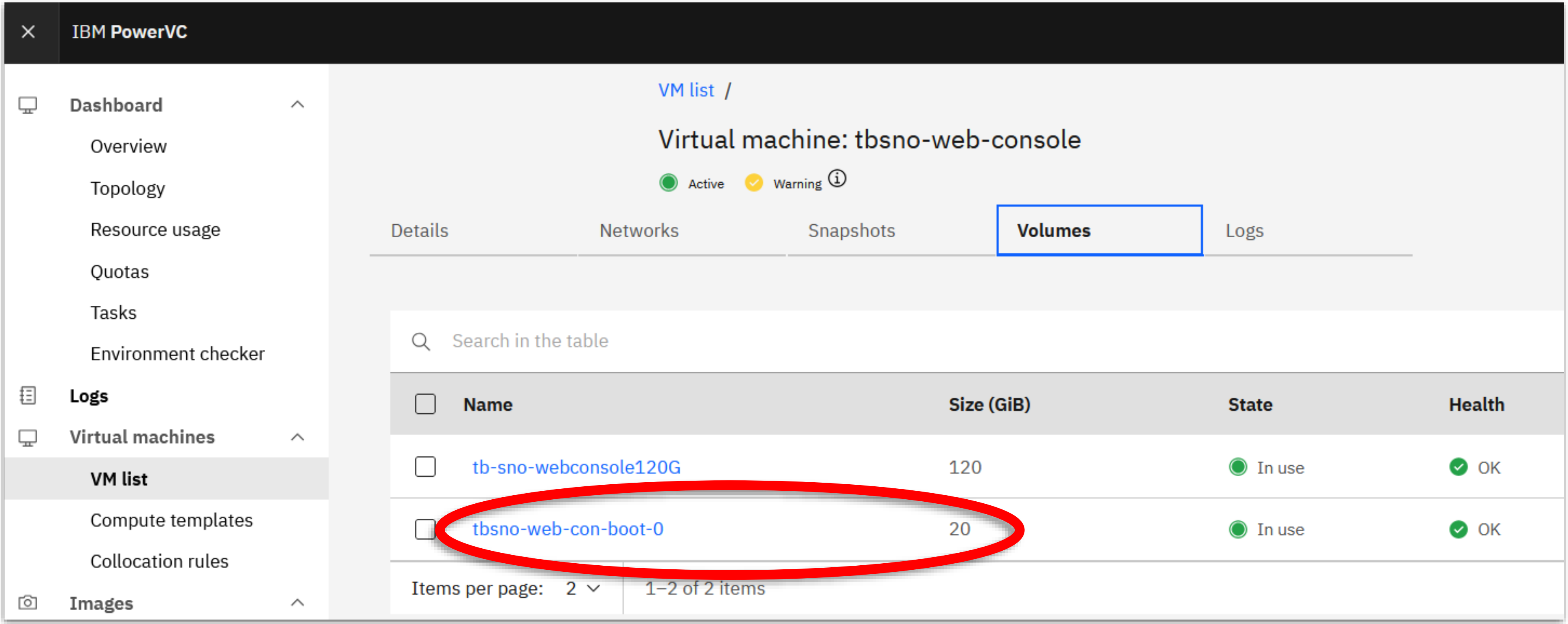


LVMCluster required

Create a LVMCluster instance to use this Operator.

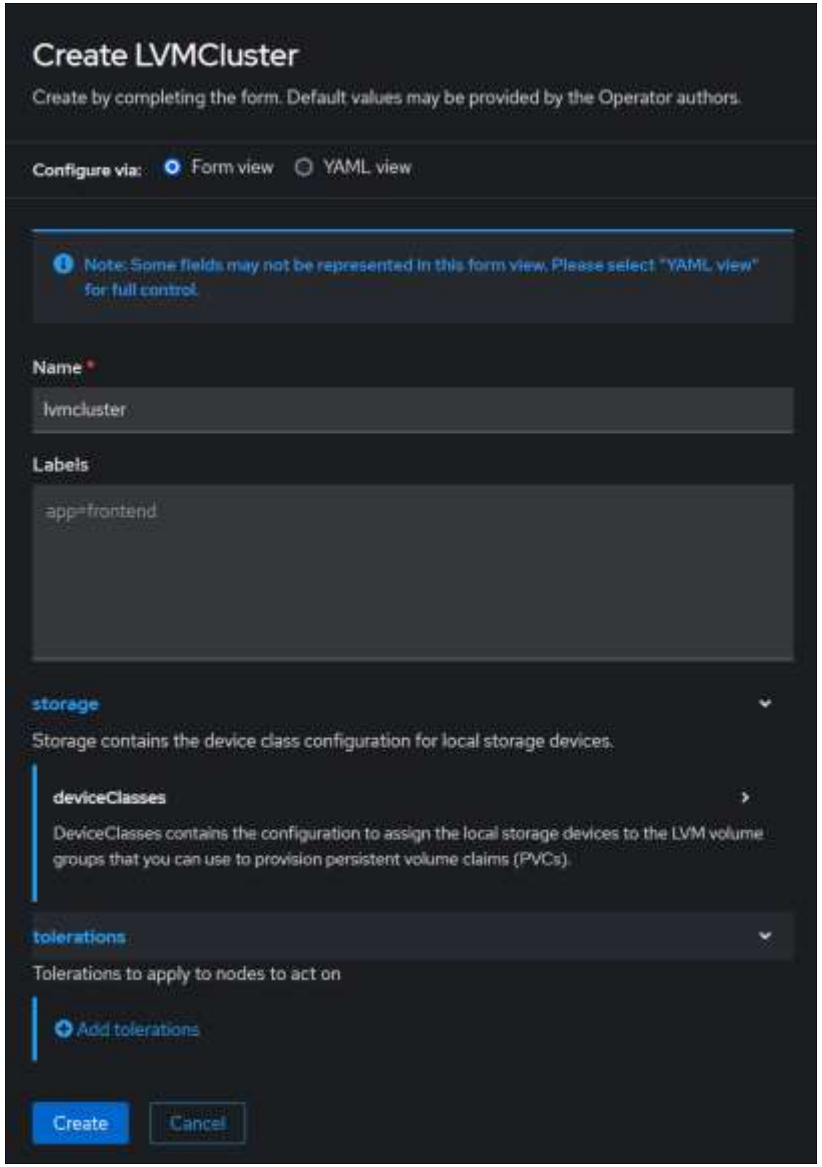
LVMCluster is the Schema for the lvmclusters API

Disque SAN pour LVM storage dans OCP



Virtual machine: tbsno-web-console

☐	Name	Size (GiB)	State	Health
☐	tb-sno-webconsole120G	120	● In use	✓ OK
☐	tbsno-web-con-boot-0	20	● In use	✓ OK



Create LVMCluster

Create by completing the form. Default values may be provided by the Operator authors.

Configure via: ● Form view ○ YAML view

Note: Some fields may not be represented in this form view. Please select "YAML view" for full control.

Name *

lvmcluster

Labels

app=frontend

storage

Storage contains the device class configuration for local storage devices.

deviceClasses

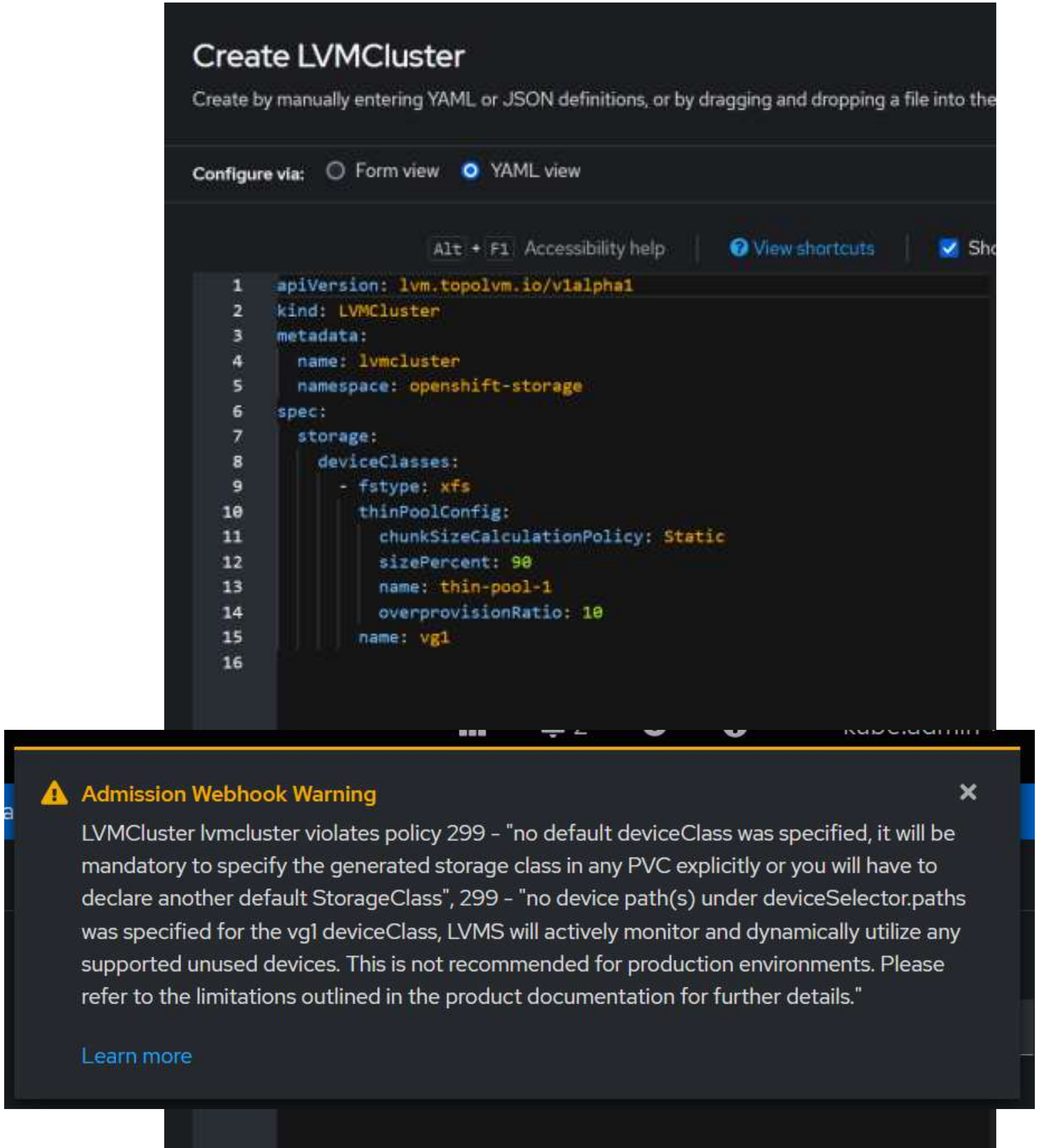
DeviceClasses contains the configuration to assign the local storage devices to the LVM volume groups that you can use to provision persistent volume claims (PVCs).

tolerations

Tolerations to apply to nodes to act on

Add tolerations

Create Cancel



Create LVMCluster

Create by manually entering YAML or JSON definitions, or by dragging and dropping a file into the

Configure via: ○ Form view ● YAML view

```yaml
apiVersion: lvm.topolvm.io/v1alpha1
kind: LVMCluster
metadata:
  name: lvmcluster
  namespace: openshift-storage
spec:
  storage:
    deviceClasses:
      - fstype: xfs
        thinPoolConfig:
          chunkSizeCalculationPolicy: Static
          sizePercent: 90
          name: thin-pool-1
          overprovisionRatio: 10
        name: vg1
```

Admission Webhook Warning

LVMCluster lvmcluster violates policy 299 - "no default deviceClass was specified, it will be mandatory to specify the generated storage class in any PVC explicitly or you will have to declare another default StorageClass", 299 - "no device path(s) under deviceSelector.paths was specified for the vg1 deviceClass, LVMS will actively monitor and dynamically utilize any supported unused devices. This is not recommended for production environments. Please refer to the limitations outlined in the product documentation for further details."

Learn more

Vue du LVM dans CoreOS

```
# ssh -i ~/.ssh/id_rsa core@9.nnn.nnn.nnn
Red Hat Enterprise Linux CoreOS 417.94.202410160352-0
Part of OpenShift 4.17, RHCOS is a Kubernetes-native operating system
managed by the Machine Config Operator (`clusteroperator/machine-config`).
WARNING: Direct SSH access to machines is not recommended; instead,
make configuration changes via `machineconfig` objects:
https://docs.openshift.com/container-platform/4.17/architecture/architecture-rhcos.html
---
[core@fa-fb-45-ag-9ac-20 ~]$ sudo su -
[root@fa-fb-45-ag-9ac-20 ~]# pvs
PV VG Fmt Attr PSize PFree
/dev/mapper/mpatha vg1 lvm2 a-
[root@fa-fb-45-ag-9ac-20 ~]# lvs
LV VG Attr LSize Pool Origin Data% Meta% Move Log Cpy%Sync Convert
thin-pool-1 vg1 twi-a-tz-- 17.97g 0.00 10.58
```

Connexion par SSH sur CoreOS.

Par un POD de debug depuis openshift ou par oc debug :

```
sh-5.1# chroot /host
sh-5.1# pvs
  PV                 VG  Fmt  Attr PSize   PFree
  /dev/mapper/mpatha vg1 lvm2 a--  <20.00g 2.00g
```

```
[root@fa-fb-45-ag-9ac-20 ~]# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINTS
sda 8:0 0 20G 0 disk
└─mpatha 253:0 0 20G 0 mpath
├─vg1-thin--pool--1_tmeta 253:6 0 12M 0 lvm
│ └─vg1-thin--pool--1 253:8 0 18G 0 lvm
└─vg1-thin--pool--1_tdata 253:7 0 18G 0 lvm
└─vg1-thin--pool--1 253:8 0 18G 0 lvm
sdb 8:16 0 120G 0 disk
└─mpathb 253:1 0 120G 0 mpath
├─mpathb1 253:2 0 4M 0 part
├─mpathb2 253:3 0 1M 0 part
├─mpathb3 253:4 0 384M 0 part /boot
└─mpathb4 253:5 0 119.6G 0 part /var/lib/kubelet/pods/1e1cdbba-6aad-4a3c-a666-56fcb0a923da/volume-
subpaths/nginx-conf/networking-console-plugin/1
/var/lib/kubelet/pods/522a0b0a-bab5-466b-9f5e-4e5501706397/volume-subpaths/nginx-conf/monitoring-plugin/1
/var
/sysroot/ostree/deploy/rhcos/var
/usr
/etc
/
/sysroot
sdc 8:32 0 20G 0 disk
└─mpatha 253:0 0 20G 0 mpath
├─vg1-thin--pool--1_tmeta 253:6 0 12M 0 lvm
│ └─vg1-thin--pool--1 253:8 0 18G 0 lvm
└─vg1-thin--pool--1_tdata 253:7 0 18G 0 lvm
└─vg1-thin--pool--1 253:8 0 18G 0 lvm
sdd 8:48 0 120G 0 disk
└─mpathb 253:1 0 120G 0 mpath
├─mpathb1 253:2 0 4M 0 part
├─mpathb2 253:3 0 1M 0 part
├─mpathb3 253:4 0 384M 0 part /boot
└─mpathb4 253:5 0 119.6G 0 part /var/lib/kubelet/pods/1e1cdbba-6aad-4a3c-a666-56fcb0a923da/volume-
subpaths/nginx-conf/networking-console-plugin/1
/var/lib/kubelet/pods/522a0b0a-bab5-466b-9f5e-4e5501706397/volume-subpaths/nginx-conf/monitoring-plugin/1
/var
/sysroot/ostree/deploy/rhcos/var
/usr
/etc
/
/sysroot
```

Effacer le disque dans CoreOS pour le réutiliser

Quel disque ?

```
sh-5.1# chroot /host
sh-5.1# pvs
PV VG Fmt Attr PSize PFree
/dev/mapper/mpatha vg1 lvm2 a--
sh-5.1# fdisk
fdisk: bad usage
Try 'fdisk --help' for more information.
sh-5.1# lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINTS
sda 8:0 0 20G 0 disk
`-mpatha 253:0 0 20G 0 mpath
|-vg1-thin--pool--1_tmeta 253:6 0 12M 0 lvm
| `-vg1-thin--pool--1 253:8 0 18G 0 lvm
`-vg1-thin--pool--1_tdata 253:7 0 18G 0 lvm
`-vg1-thin--pool--1 253:8 0 18G 0 lvm
sdb 8:16 0 120G 0 disk
`-mpathb 253:1 0 120G 0 mpath
|-mpathb1 253:2 0 4M 0 part
|-mpathb2 253:3 0 1M 0 part
|-mpathb3 253:4 0 384M 0 part /boot
`-mpathb4 253:5 0 119.6G 0 part /var/lib/kubelet/pods/1e1cdbba-6aad-4a3c-a666-56fcb0a923da/volume-subpaths/nginx-conf/networking-console-plugin/1
/var/lib/kubelet/pods/522a0b0a-bab5-466b-9f5e-4e5501706397/volume-subpaths/nginx-conf/monitoring-plugin/1
```

Effacer avec fdisk :

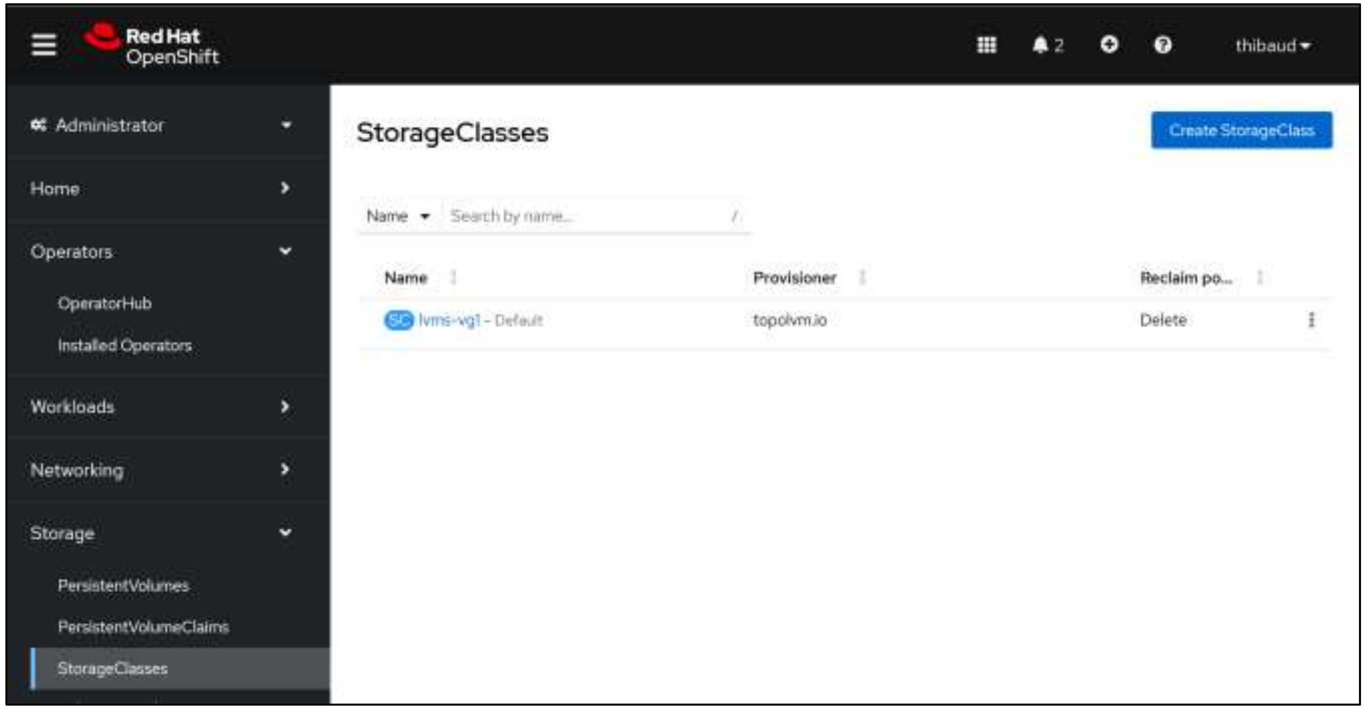
```
sfdisk --delete /dev/sda
```

Recycler un disque

- **Reusing a volume group from the previous LVM Storage installation**
- https://docs.openshift.com/container-platform/4.17/storage/persistent_storage/persistent_storage_local/persistent-storage-using-lvms.html#lvms-reusing-vg-from-prev-installation_logical-volume-manager-storage

Une StorageClass est créée

- A StorageClass provides a way for administrators to describe the *classes* of storage they offer. Different classes might map to quality-of-service levels, or to backup policies, or to arbitrary policies determined by the cluster administrators. Kubernetes itself is unopinionated about what classes represent.
- The Kubernetes concept of a storage class is similar to “profiles” in some other storage system designs.

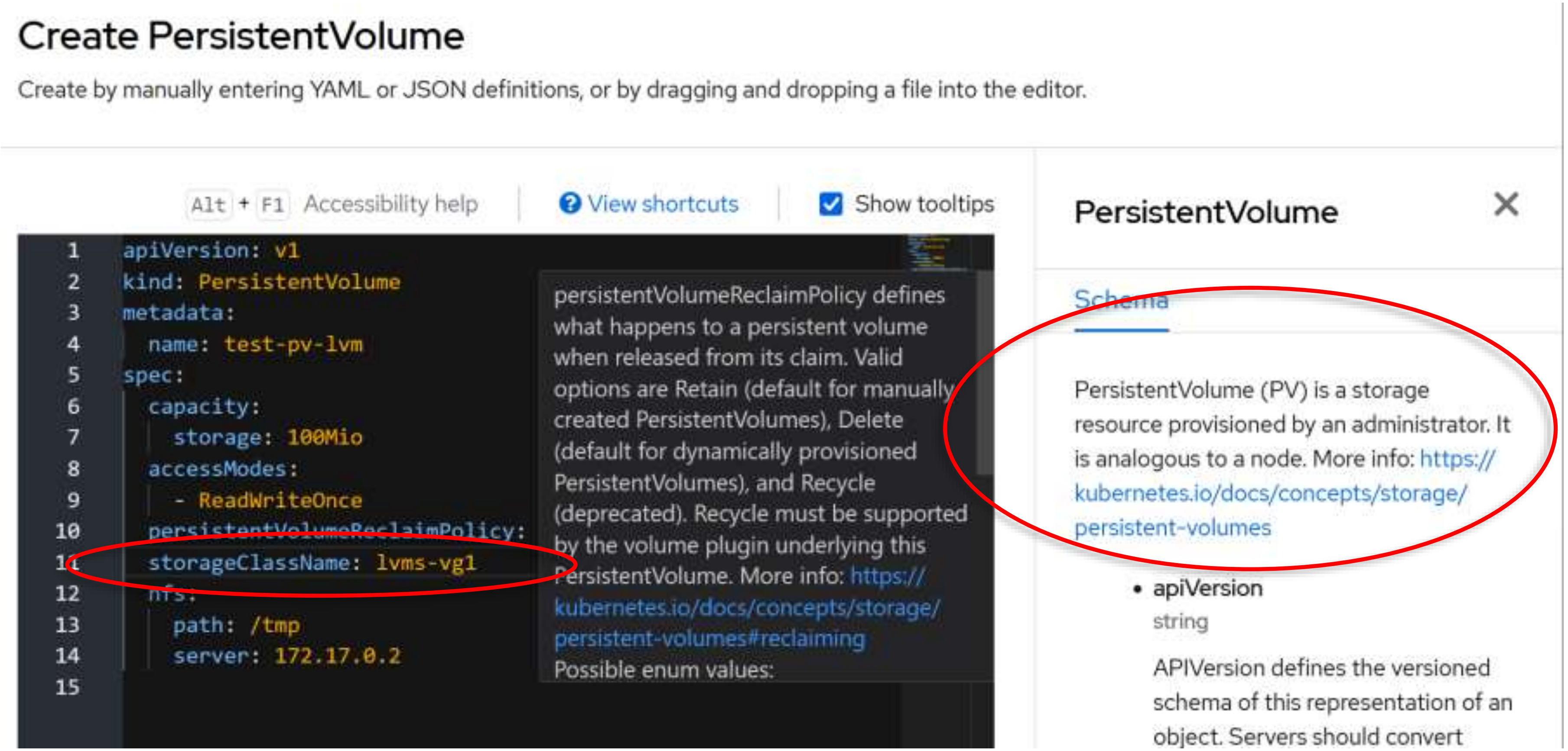


StorageClasses

Name	Provisioner	Reclaim po...
SC lvms-vg1 - Default	topolvm.io	Delete

Créer un volume

Attention : on ne peut pas renommer un PV



Create PersistentVolume

Create by manually entering YAML or JSON definitions, or by dragging and dropping a file into the editor.

```
apiVersion: v1
kind: PersistentVolume
metadata:
  name: test-pv-lvm
spec:
  capacity:
    storage: 100Mio
  accessModes:
    - ReadWriteOnce
  persistentVolumeReclaimPolicy:
  storageClassName: lvms-vg1
  nfs:
    path: /tmp
    server: 172.17.0.2
```

persistentVolumeReclaimPolicy defines what happens to a persistent volume when released from its claim. Valid options are Retain (default for manually created PersistentVolumes), Delete (default for dynamically provisioned PersistentVolumes), and Recycle (deprecated). Recycle must be supported by the volume plugin underlying this PersistentVolume. More info: https://kubernetes.io/docs/concepts/storage/persistent-volumes#reclaiming
Possible enum values:

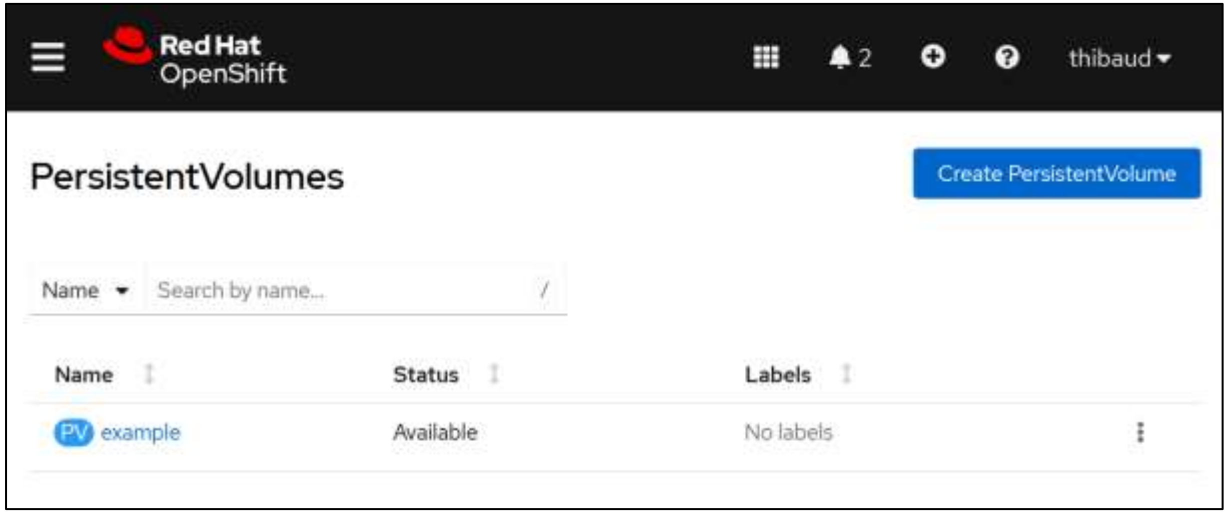
PersistentVolume

PersistentVolume (PV) is a storage resource provisioned by an administrator. It is analogous to a node. More info: https://kubernetes.io/docs/concepts/storage/persistent-volumes

- apiVersion
  string

  APIVersion defines the versioned schema of this representation of an object. Servers should convert

- Un volume est prêt à être utilisé par un pod pour stocker des données



PersistentVolumes

Name	Status	Labels
PV example	Available	No labels

```
thibaud@thibaud-x86:~$ oc debug node/fa-fb-23-ag-9ac-20
Starting pod/fa-fb-23-ag-9ac-20-debug-f269r ...
To use host binaries, run `chroot /host`
Pod IP: 9.xxx
If you don't see a command prompt, try pressing enter.
sh-5.1# chroot /host
sh-5.1# pvs
PV VG Fmt Attr PSize PFree
/dev/mapper/mpatha vg1 lvm2 a--
sh-5.1# lvs
LV VG Attr LSize Pool Origin Data% Meta% Move Log Cpy%Sync Convert
thin-pool-1 vg1 twi-a-tz-- 17.97g 0.00 10.58

thibaud@thibaud-x86:~$ oc get pv
NAME      CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS      CLAIM   STORAGECLASS   VOLUMEATTRIBUTESCLASS   REASON   AGE
example   5Gi        RWO            Retain           Available           lvms-vg1       <unset>                          95m
```

MERCI