Red Hat OpenShift avec NetApp

NetApp Solutions

NetApp
April 25, 2024

Sommaire

NVA-1160: Red Hat OpenShift avec NetApp

Alan Cowles et Nikhil M Kulkarni, NetApp

Ce document de référence assure la validation du déploiement de la solution Red Hat OpenShift, déployée via IPI (installer provisionnés Infrastructure) dans plusieurs environnements de data Center différents, comme validé par NetApp. Il détaille également l'intégration du stockage avec les systèmes de stockage NetApp grâce à l'orchestrateur de stockage Astra Trident pour la gestion du stockage persistant. Enfin, un certain nombre de validations de solutions et d'utilisations réelles sont explorées et documentées.

Cas d'utilisation

L'architecture de la solution Red Hat OpenShift avec NetApp a été conçue pour offrir une valeur exceptionnelle aux clients dans les cas d'utilisation suivants :

- Déploiement et gestion simples de Red Hat OpenShift déployé avec IPI (installation Provisioné Infrastructure) sur un serveur bare Metal, Red Hat OpenStack Platform, Red Hat Virtualization et VMware vSphere.
- L'association de la puissance des workloads virtualisés et des conteneurs d'entreprise avec Red Hat OpenShift est déployée virtuellement sur OSP, RHV ou vSphere, ou sur un système bare Metal avec OpenShift Virtualization.
- Exemples de configurations et d'utilisations réelles mettant en avant les fonctionnalités de Red Hat OpenShift avec le stockage NetApp et Astra Trident, l'orchestrateur de stockage open source pour Kubernetes.

Valeur commerciale

Les entreprises se tournent de plus en plus vers les pratiques DevOps pour créer de nouveaux produits, réduire les cycles de lancement et ajouter rapidement de nouvelles fonctionnalités. En raison de leur nature inné et agile, les conteneurs et les microservices ont un rôle essentiel dans l'accompagnement des pratiques DevOps. Cependant, la pratique du DevOps à l'échelle de production dans un environnement d'entreprise présente ses propres défis et impose certaines exigences à l'infrastructure sous-jacente, notamment :

- Haute disponibilité à tous les niveaux de la pile
- Simplicité des procédures de déploiement
- Des opérations et des mises à niveau non disruptives
- Une infrastructure programmable et basée sur des API pour suivre le rythme de l'agilité des microservices
- Colocation avec garanties de performances
- Possibilité d'exécuter simultanément des workloads virtualisés et conteneurisés
- Possibilité de faire évoluer indépendamment l'infrastructure en fonction des besoins des workloads

Red Hat OpenShift avec NetApp reconnaît ces défis et présente une solution qui aide à résoudre chaque problème en mettant en œuvre le déploiement entièrement automatisé de Red Hat OpenShift IPI dans l'environnement de data Center choisi par le client.

Présentation de la technologie

La solution Red Hat OpenShift avec NetApp comprend les principaux composants suivants :

Plateforme de conteneurs Red Hat OpenShift

Red Hat OpenShift Container Platform est une plateforme Kubernetes d'entreprise entièrement prise en charge. Red Hat apporte plusieurs améliorations à l'open source Kubernetes afin de fournir une plateforme applicative avec tous les composants entièrement intégrés pour créer, déployer et gérer des applications conteneurisées.

Pour en savoir plus, rendez-vous sur le site web d'OpenShift "ici".

Systèmes de stockage NetApp

NetApp propose plusieurs systèmes de stockage parfaitement adaptés aux data centers d'entreprise et aux déploiements de cloud hybride. Le portefeuille NetApp inclut des systèmes de stockage NetApp ONTAP, NetApp Element et E-Series, tous capables d'assurer un stockage persistant pour les applications conteneurisées.

Pour en savoir plus, rendez-vous sur le site Web de NetApp "ici".

Intégrations du stockage NetApp

NetApp Astra Control Center propose un ensemble complet de services de gestion du stockage et des données respectueuse des applications pour les workloads Kubernetes avec état, déployé dans un environnement sur site et optimisé par la technologie de protection des données NetApp de confiance.

Pour plus d'informations, rendez-vous sur le site Web NetApp Astra "ici".

Astra Trident est un orchestrateur de stockage open source entièrement pris en charge pour les conteneurs et les distributions Kubernetes, y compris Red Hat OpenShift.

Pour en savoir plus, rendez-vous sur le site Web Astra Trident "ici".

Options de configuration avancées

Cette section est dédiée aux personnalisations que les utilisateurs du monde réel devraient probablement réaliser lors du déploiement de cette solution en production, telles que la création d'un registre d'images privées dédié ou le déploiement d'instances personnalisées d'équilibreur de charge.

Matrice de prise en charge actuelle pour les versions validées

De déduplication	Objectif	Version logicielle
NetApp ONTAP	Stockage	9.8 février 9.9.1
NetApp Element	Stockage	12.3
NetApp Astra Control Center	Gestion des données intégrant la cohérence applicative	21.12.60
NetApp Astra Trident	Orchestration du stockage	22.01.0
Red Hat OpenShift	Orchestration de conteneurs	4.6 EUS, 4.7, 4.8
Plateforme Red Hat OpenStack	Infrastructure de cloud privé	16.1

Red Hat Virtualization	Virtualisation du data Center	4.4
VMware vSphere	Virtualisation du data Center	6.7U3

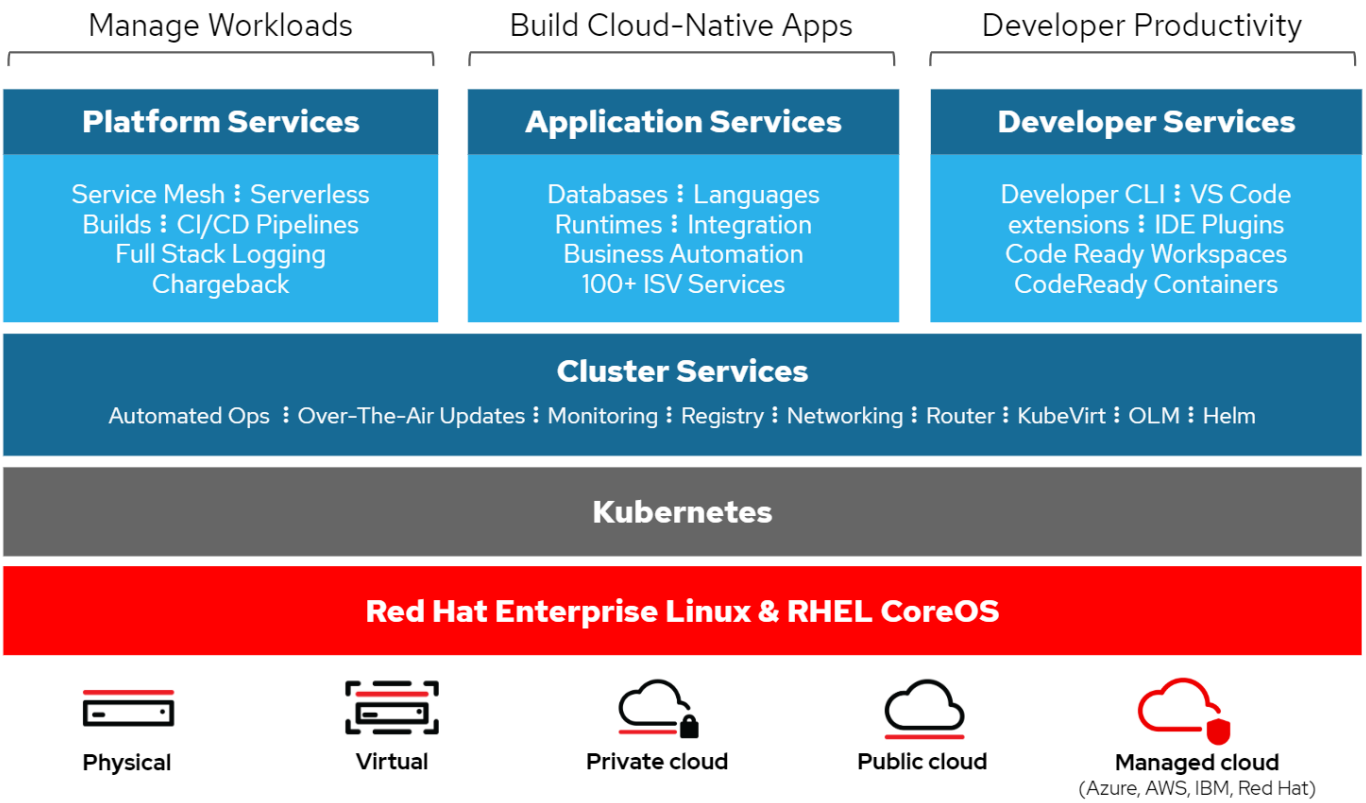
Présentation d'OpenShift

Red Hat OpenShift Container Platform réunit le développement et les opérations IT sur une plateforme unique pour concevoir, déployer et gérer de façon cohérente les applications dans l'ensemble des infrastructures de cloud hybride et sur site. Red Hat OpenShift repose sur l'innovation open source et sur les normes du secteur, notamment Kubernetes et Red Hat Enterprise Linux CoreOS, la principale distribution Linux d'entreprise au monde conçue pour les workloads basés sur des conteneurs. OpenShift fait partie du programme Kubernetes certifié Cloud Native Computing Foundation (CNCF), qui assure la portabilité et l'interopérabilité des workloads de conteneurs.

Red Hat OpenShift offre les fonctionnalités suivantes :

- **Provisionnement en libre-service** les développeurs peuvent créer rapidement et facilement des applications à la demande à partir des outils qu'ils utilisent le plus, tandis que les opérations conservent un contrôle total sur l'ensemble de l'environnement.
- **Stockage persistant** en prenant en charge le stockage persistant, OpenShift Container Platform vous permet d'exécuter à la fois des applications avec état et des applications cloud sans état.
- **Intégration continue et développement continu (ci/CD)** cette plate-forme de code source gère les images de construction et de déploiement à grande échelle.
- **Normes open source** ces normes incorporent l'Open Container Initiative (OCI) et Kubernetes pour l'orchestration de conteneurs, en plus d'autres technologies open source. Vous n'êtes pas limité aux technologies ou à la feuille de route commerciale d'un fournisseur spécifique.
- **Pipelines ci/CD** OpenShift fournit une prise en charge prête à l'emploi des pipelines ci/CD pour que les équipes de développement puissent automatiser chaque étape du processus de distribution des applications et s'assurer qu'elles sont exécutées à chaque modification apportée au code ou à la configuration de l'application.
- **Contrôle d'accès basé sur les rôles (RBAC)** cette fonction fournit un suivi d'équipe et d'utilisateur pour aider à organiser un grand groupe de développeurs.
- **Automated Build and Deploy** OpenShift offre aux développeurs la possibilité de créer leurs applications conteneurisées ou de faire construire les conteneurs à partir du code source de l'application ou même des binaires. La plateforme automatise ensuite le déploiement de ces applications dans l'infrastructure en fonction de la caractéristique définie pour les applications. Par exemple, la quantité de ressources à allouer et le lieu où elles doivent être déployées sur l'infrastructure, afin qu'elles soient compatibles avec les licences tierces.
- **Environnements cohérents** OpenShift veille à ce que l'environnement provisionné pour les développeurs et tout au long du cycle de vie de l'application soit cohérent du système d'exploitation aux bibliothèques, à la version d'exécution (par exemple, Java Runtime), et même le runtime de l'application en cours d'utilisation (par exemple, tomcat) afin de supprimer les risques provenant d'environnements incohérents.
- **Gestion de la configuration** la gestion de la configuration et des données sensibles est intégrée à la plate-forme pour s'assurer qu'une configuration d'application cohérente et indépendante de l'environnement est fournie à l'application, quelles que soient les technologies utilisées pour construire l'application ou l'environnement qu'elle est
déploiement.

- **Journaux d'application et mesures.** la rétroaction rapide est un aspect important du développement d'application. La surveillance intégrée et la gestion des journaux OpenShift fournissent aux développeurs des metrics immédiates afin d'étudier leur comportement à travers les changements et de pouvoir résoudre les problèmes le plus tôt possible au cours du cycle de vie de l'application.
- **Sécurité et catalogue de conteneurs** OpenShift offre la colocation et protège l'utilisateur contre l'exécution de code nuisible en utilisant la sécurité établie avec Security-Enhanced Linux (SELinux), CGroups et Secure Computing mode (seccomp) pour isoler et protéger les conteneurs. Il fournit également le cryptage via des certificats TLS pour les différents sous-systèmes et l'accès aux conteneurs certifiés Red Hat (access.redhat.com/containers) qui sont analysés et classés en mettant l'accent sur la sécurité afin de fournir aux utilisateurs des conteneurs d'applications certifiés, fiables et sécurisés.



Méthodes de déploiement pour Red Hat OpenShift

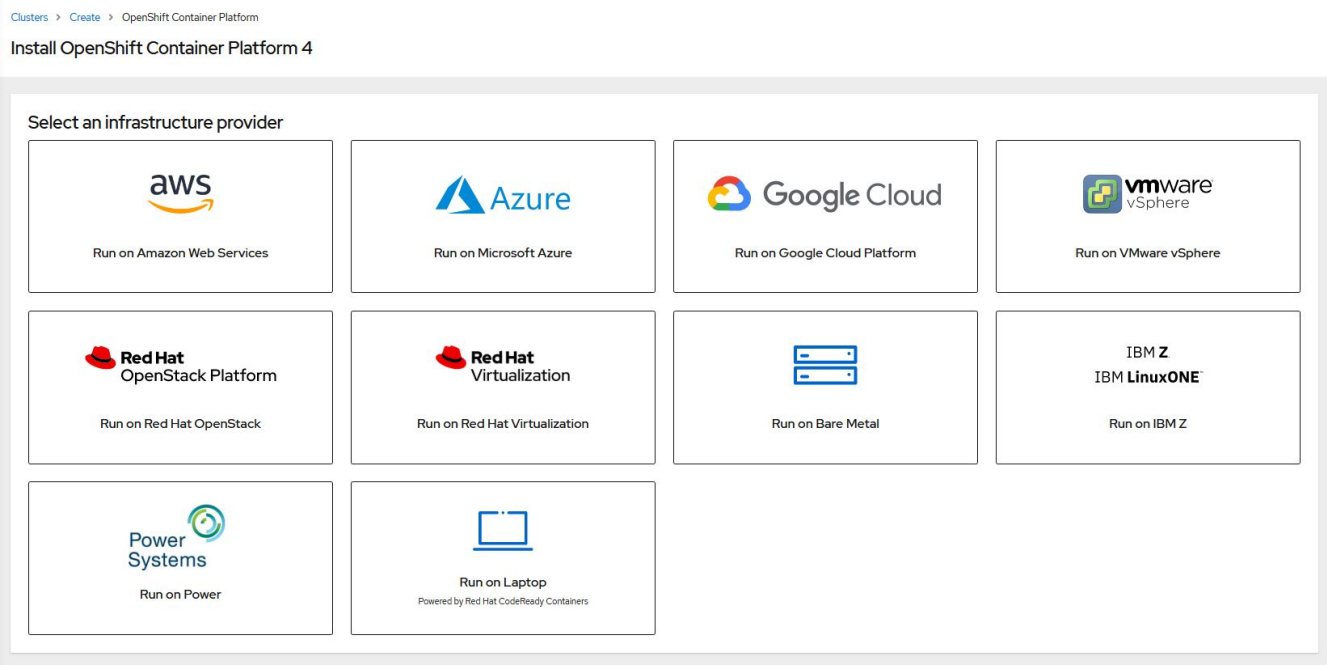
Depuis Red Hat OpenShift 4, les méthodes de déploiement d'OpenShift incluent les déploiements manuels utilisant l'UPI (User Provisioned Infrastructure) pour des déploiements hautement personnalisés ou des déploiements entièrement automatisés à l'aide de l'IPI (installer Provisioned Infrastructure).

Dans la plupart des cas, la méthode d'installation IPI est la plus recommandée, car elle permet le déploiement rapide des clusters OpenShift pour les environnements de développement, de test et de production.

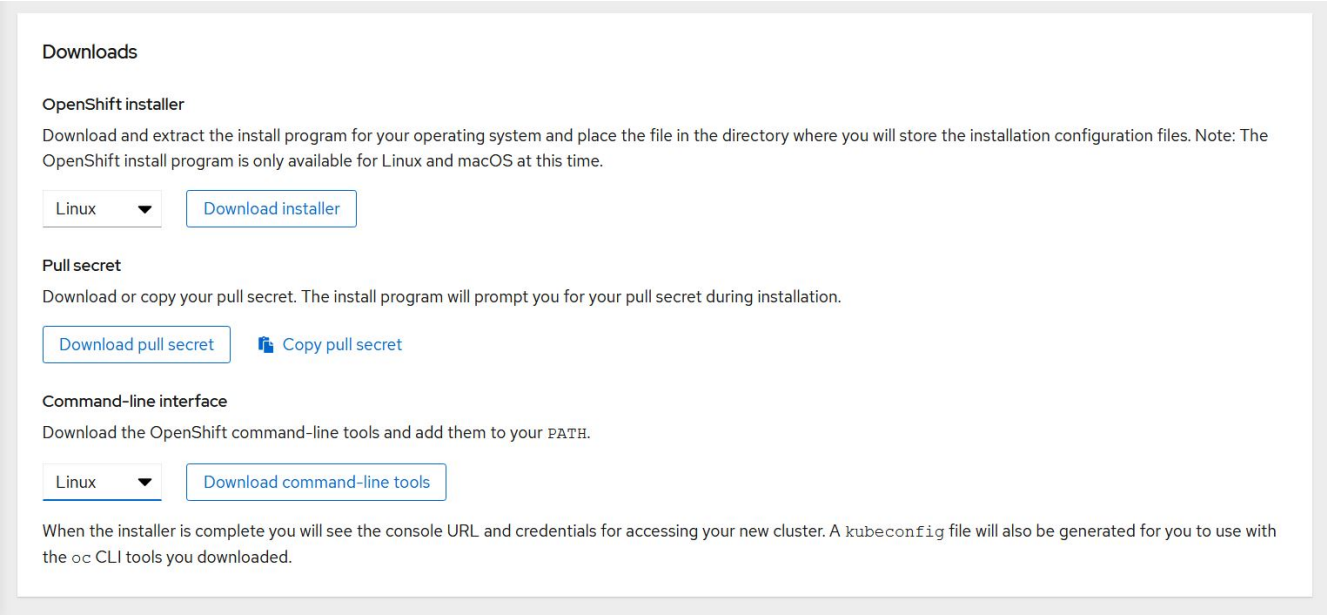
Installation IPI de Red Hat OpenShift

Le déploiement d'IPI (installer Provisionfortes Infrastructure) d'OpenShift implique les étapes de haut niveau suivantes :

1. Visitez Red Hat OpenShift "site web" Et connectez-vous à l'aide de votre login SSO.
2. Sélectionnez l'environnement dans lequel vous souhaitez déployer Red Hat OpenShift.



3. Sur l'écran suivant, téléchargez le programme d'installation, le secret de collecte unique et les outils CLI pour la gestion.



4. Suivez le "instructions d'installation" Fourni par Red Hat pour un déploiement dans l'environnement de votre choix.

Les déploiements OpenShift validés par NetApp

NetApp a testé et validé le déploiement de Red Hat OpenShift dans ses laboratoires à l'aide de la méthode de déploiement IPI (installer provisionnés Infrastructure) dans chacun des environnements de data Center suivants :

- "OpenShift sur bare Metal"
- "OpenShift sur Red Hat OpenStack Platform"

- "OpenShift sur Red Hat Virtualization"
- "OpenShift sur VMware vSphere"

OpenShift sur bare Metal

OpenShift sur bare Metal permet un déploiement automatisé de OpenShift Container Platform sur des serveurs génériques.

OpenShift sur bare Metal est similaire aux déploiements virtuels d'OpenShift. Ce système facilite le déploiement, accélère le provisionnement et permet l'évolutivité des clusters OpenShift, tout en supportant des workloads virtualisés pour les applications qui ne sont pas prêtes pour les conteneurs. En déployant sur un serveur bare Metal, vous n'avez pas à gérer l'environnement d'hyperviseur hôte sans frais supplémentaires, en plus de l'environnement OpenShift. En le déployant directement sur des serveurs bare Metal, vous pouvez également réduire les limitations de la surcharge physique liées au partage des ressources entre l'hôte et l'environnement OpenShift.

OpenShift sur bare Metal offre les fonctionnalités suivantes :

- **Déploiement IPI ou installation assistée** avec un cluster OpenShift déployé par l'IPI (installer Provisioning Infrastructure) sur des serveurs bare Metal, les clients peuvent déployer un environnement OpenShift extrêmement polyvalent et facilement évolutif directement sur des serveurs ordinaires, sans devoir gérer une couche d'hyperviseur.
- **Compact Cluster design** pour minimiser les exigences matérielles, OpenShift sur bare Metal permet aux utilisateurs de déployer des clusters de seulement 3 nœuds, en permettant aux nœuds du plan de contrôle OpenShift de faire également office de nœuds worker et de conteneurs hôtes.
- **OpenShift Virtualization** OpenShift peut exécuter des machines virtuelles dans des conteneurs à l'aide d'OpenShift Virtualization. Cette virtualisation native de conteneur exécute l'hyperviseur KVM dans un conteneur, et connecte les volumes persistants pour le stockage des machines virtuelles.
- **Infrastructure optimisée pour L'IA/ML** déployez des applications telles que Kubeflow pour les applications de machine learning en intégrant des nœuds workers basés sur des processeurs graphiques à votre environnement OpenShift et en exploitant OpenShift Advanced Scheduling.

Conception du réseau

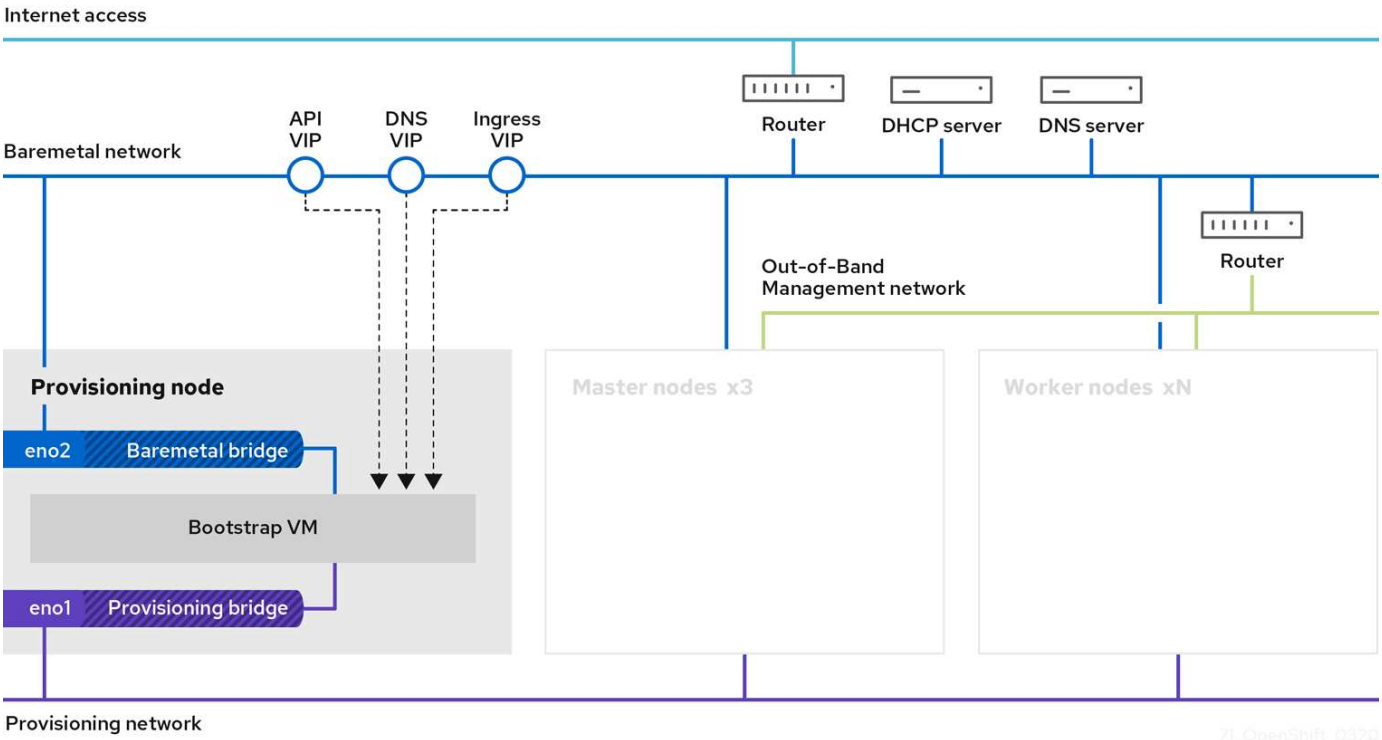
La solution Red Hat OpenShift sur NetApp utilise deux switchs de données pour assurer la connectivité des données primaires à 25 Gbit/s. Il utilise également deux commutateurs de gestion qui fournissent une connectivité à 1 Gbit/s pour la gestion intrabande des nœuds de stockage et la gestion hors bande pour la fonctionnalité IPMI.

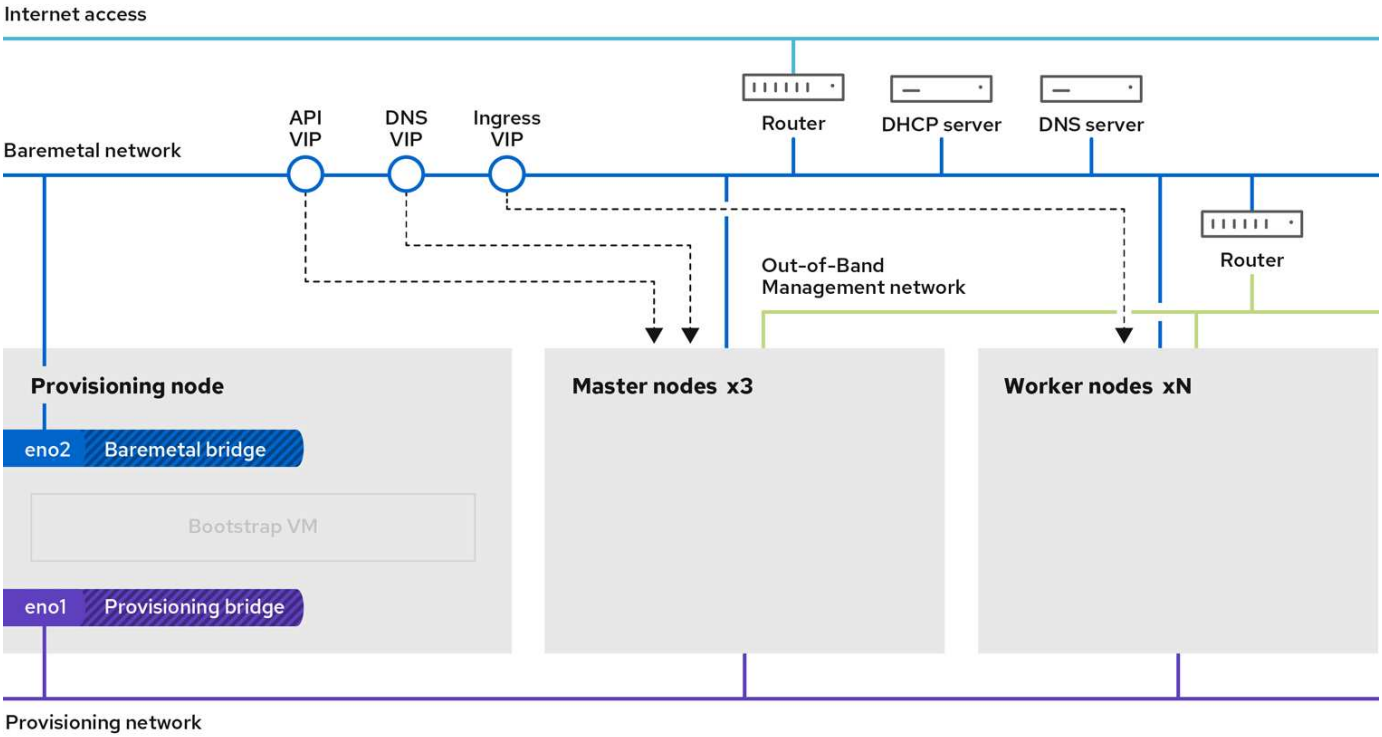
Pour le déploiement d'IPI sans système d'exploitation OpenShift, vous devez créer un nœud de provisionnement, une machine Red Hat Enterprise Linux 8 qui doit disposer d'interfaces réseau connectées à des réseaux distincts.

- **Provisioning network** ce réseau est utilisé pour démarrer les nœuds sans système d'exploitation et installer les images et packages nécessaires pour déployer le cluster OpenShift.
- **Réseau bare-Metal** ce réseau est utilisé pour la communication publique du cluster après son déploiement.

Dans le cadre de la configuration du nœud de provisionnement, le client crée des interfaces de pont qui permettent au trafic de s'acheminer correctement sur le nœud lui-même et sur la machine virtuelle de démarrage provisionnée pour le déploiement. Une fois le cluster déployé, l'API et les adresses VIP d'entrée sont migrées du nœud bootstrap vers le cluster récemment déployé.

Les images suivantes illustrent l'environnement au cours du déploiement IPI et une fois le déploiement terminé.





Exigences VLAN

La solution Red Hat OpenShift avec NetApp est conçue pour séparer de façon logique le trafic réseau à différents fins, à l'aide de réseaux locaux virtuels (VLAN).

VLAN	Objectif	ID VLAN
Réseau de gestion hors bande	Gestion pour les nœuds bare Metal et IPMI	16
Réseau sans système d'exploitation	Un seul cluster est disponible pour les services OpenShift	181
Réseau de provisionnement	Réseau pour l'amorçage PXE et l'installation de nœuds bare Metal via IPI	3485

> Bien que chacun de ces réseaux soit virtuellement séparé par des VLAN, chaque port physique doit être configuré en mode d'accès avec le VLAN principal affecté, car il n'existe aucun moyen de transmettre une balise VLAN au cours d'une séquence de démarrage PXE.

Ressources de prise en charge de l'infrastructure réseau

L'infrastructure suivante doit être en place avant le déploiement de la plateforme de conteneurs OpenShift :

- Au moins un serveur DNS qui fournit une résolution complète du nom d'hôte accessible à partir du réseau de gestion intrabande et du réseau de VM.
- Au moins un serveur NTP accessible depuis le réseau de gestion intrabande et le réseau de VM.
- (Facultatif) connectivité Internet sortante pour le réseau de gestion intrabande et le réseau VM.

OpenShift sur Red Hat OpenStack Platform

Red Hat OpenStack Platform offre une base intégrée pour créer, déployer et faire évoluer un cloud privé OpenStack sécurisé et fiable.

OSP est un cloud IaaS (infrastructure en tant que service) implémenté par un ensemble de services de contrôle qui gèrent les ressources de calcul, de stockage et de mise en réseau. L'environnement est géré via une interface Web qui permet aux administrateurs et aux utilisateurs de contrôler, de provisionner et d'automatiser les ressources OpenStack. De plus, l'infrastructure OpenStack est simplifiée par une vaste interface de ligne de commande et une API poussée permettant de disposer de fonctionnalités d'automatisation complètes pour les administrateurs et les utilisateurs finaux.

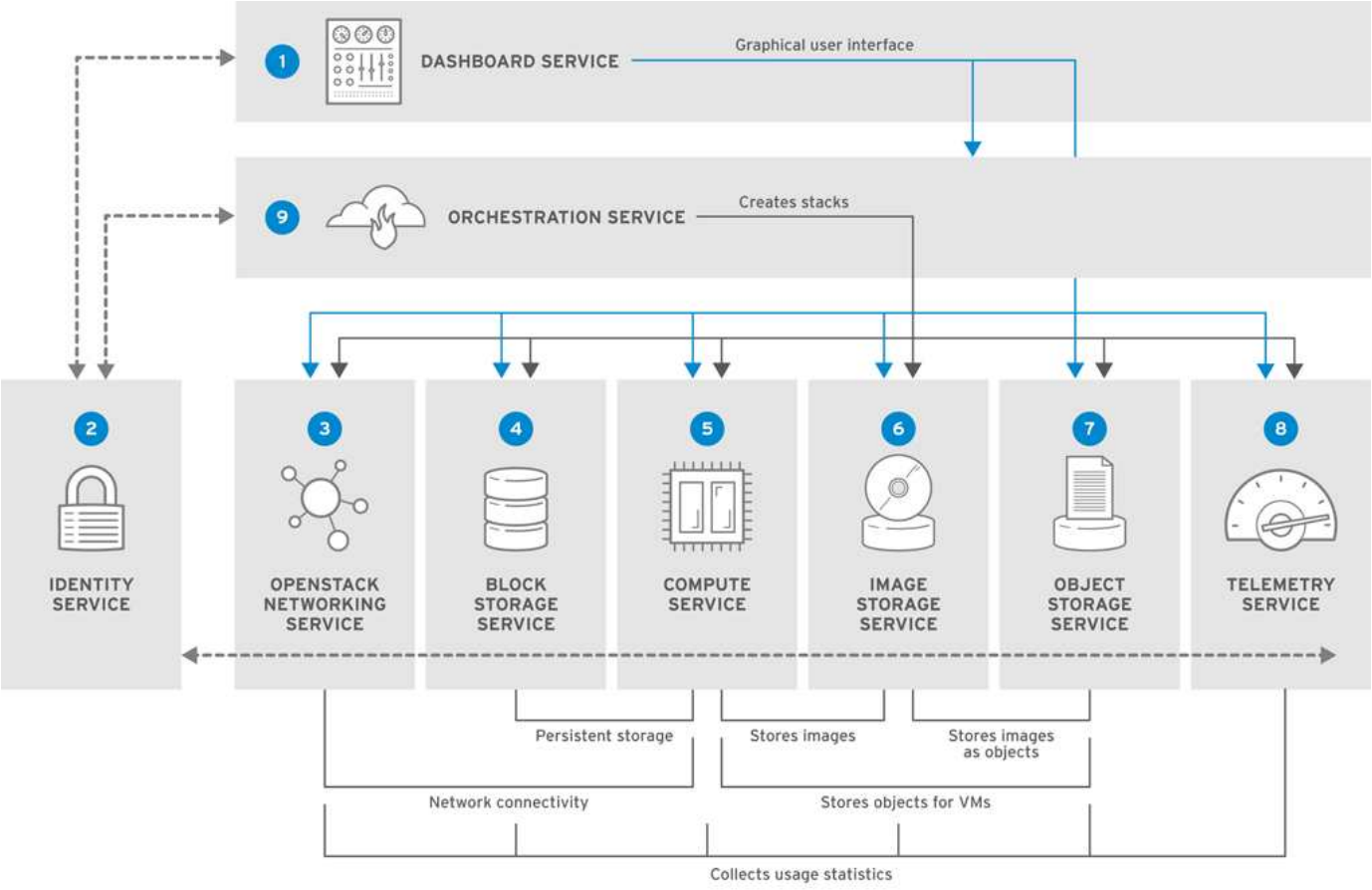
Le projet OpenStack est un projet communautaire rapidement développé qui propose des versions mises à jour tous les six mois. Dans un premier temps, Red Hat OpenStack Platform a su suivre le rythme de ce cycle de sortie en publiant une nouvelle version, ainsi que chaque version en amont, et en assurant une prise en charge à long terme pour chaque troisième version. Avec la version OSP 16.0 (basé sur OpenStack train), Red Hat a récemment choisi de ne pas suivre le rythme des numéros de version, mais de proposer de nouvelles fonctionnalités dans des sous-versions. La version la plus récente est Red Hat OpenStack Platform 16.1, qui inclut des fonctionnalités avancées backportées des versions Ussuri et Victoria en amont.

Pour plus d'informations sur OSP, consultez le "Site Web de Red Hat OpenStack Platform".

Services OpenStack

Les services de plateforme OpenStack sont déployés sous forme de conteneurs, qui permettent d'isoler les services les uns des autres et de faciliter les mises à niveau. La plateforme OpenStack utilise un ensemble de

conteneurs conçus et gérés avec Kolla. Le déploiement des services s'effectue en extrayant des images de conteneur à partir du portail personnalisé Red Hat. Ces conteneurs de service sont gérés à l'aide de la commande Podman, et sont déployés, configurés et gérés avec Red Hat OpenStack Director.



Service	Nom du projet	Description
Tableau de bord	Horizon	Tableau de bord Web que vous utilisez pour gérer les services OpenStack.
Identité	Keystone	Service centralisé d'authentification et d'autorisation des services OpenStack, et de gestion des utilisateurs, des projets et des rôles.
La mise en réseau d'OpenStack	Neutron	Assure la connectivité entre les interfaces des services OpenStack.
Stockage basé sur des blocs	Cinder	Gère les volumes de stockage bloc persistants pour les machines virtuelles (VM).
Calcul	Nouvelle	Gère et provisionne les VM s'exécutant sur les nœuds de calcul.
Image	Coup d'œil	Service de registre utilisé pour stocker des ressources telles que des images de machines virtuelles et des instantanés de volumes.
Stockage objet	SWIFT	Permet aux utilisateurs de stocker et de récupérer des fichiers et des données arbitraires.
Télémétrie	Ceilamomètre	Mesure l'utilisation des ressources du cloud.

Service	Nom du projet	Description
Orchestration	Chaleur	Moteur d'orchestration basé sur des modèles qui prend en charge la création automatique de piles de ressources.

Conception du réseau

La solution Red Hat OpenShift avec NetApp utilise deux switchs de données pour assurer la connectivité des données primaires à 25 Gbit/s. Il utilise également deux commutateurs de gestion supplémentaires qui fournissent une connectivité à 1 Gbit/s pour la gestion intrabande des nœuds de stockage et la gestion hors bande des fonctionnalités IPMI.

Red Hat OpenStack Director exige une fonctionnalité IPMI pour déployer Red Hat OpenStack Platform à l'aide du service de provisionnement sans système d'exploitation ironique.

Exigences VLAN

Red Hat OpenShift avec NetApp est conçu pour séparer logiquement le trafic réseau à différents fins à l'aide de réseaux locaux virtuels (VLAN). Cette configuration peut être adaptée aux besoins du client ou pour assurer une isolation supplémentaire pour des services réseau spécifiques. Le tableau suivant répertorie les VLAN nécessaires à la mise en œuvre de la solution lors de sa validation chez NetApp.

VLAN	Objectif	ID VLAN
Réseau de gestion hors bande	Réseau utilisé pour la gestion des nœuds physiques et du service IPMI pour ironique.	16
De stockage existante	Utilisé par le réseau pour les nœuds de contrôleur, pour mapper les volumes directement pour prendre en charge des services d'infrastructure tels que Swift.	201
Stockage Cinder	Réseau utilisé pour mapper et rattacher des volumes de blocs directement aux instances virtuelles déployées dans l'environnement.	202
API interne	Réseau utilisé pour la communication entre les services OpenStack à l'aide de la communication API, des messages RPC et de la communication avec les bases de données.	301
Locataire	Neutron fournit à chaque locataire ses propres réseaux par tunneling via VXLAN. Le trafic réseau est isolé dans chaque réseau de locataires. Chaque réseau de locataires est associé à un sous-réseau IP, et les espaces de noms réseau signifient que plusieurs réseaux de locataires peuvent utiliser la même plage d'adresses sans entraîner de conflits.	302
Gestion du stockage	OpenStack Object Storage (Swift) utilise ce réseau pour synchroniser les objets de données entre les nœuds de réplication participants. Le service proxy fait office d'interface intermédiaire entre les demandes des utilisateurs et la couche de stockage sous-jacente. Le proxy reçoit les demandes entrantes et localise la réplique nécessaire pour récupérer les données demandées.	303
PXE	OpenStack Director assure le démarrage PXE dans le service de provisionnement bare Metal ironique afin d'orchestrer l'installation du Overcloud OSP.	3484

VLAN	Objectif	ID VLAN
Externe	Réseau public qui héberge le tableau de bord OpenStack (Horizon) pour une gestion graphique et permet aux appels d'API publiques de gérer les services OpenStack.	3485
Réseau de gestion dans la bande	Permet d'accéder aux fonctions d'administration système telles que l'accès SSH, le trafic DNS et le trafic NTP (Network Time Protocol). Ce réseau fait également office de passerelle pour les nœuds sans contrôleur.	3486

Ressources de prise en charge de l'infrastructure réseau

L'infrastructure suivante doit être en place avant le déploiement de la plateforme de conteneurs OpenShift :

- Au moins un serveur DNS qui fournit une résolution complète de nom d'hôte.
- Au moins trois serveurs NTP qui peuvent garder le temps synchronisé pour les serveurs de la solution.
- (Facultatif) connectivité Internet sortante pour l'environnement OpenShift.

Bonnes pratiques pour les déploiements en production

Cette section répertorie plusieurs meilleures pratiques à prendre en considération avant de déployer cette solution en production.

Déployez OpenShift dans un cloud privé OSP avec au moins trois nœuds de calcul

L'architecture vérifiée décrite dans ce document présente le déploiement matériel minimum adapté aux opérations HA en déployant trois nœuds de contrôleur OSP et deux nœuds de calcul OSP. Cette architecture garantit une configuration tolérante aux pannes dans laquelle les deux nœuds de calcul peuvent lancer des instances virtuelles et les machines virtuelles déployées peuvent migrer entre les deux hyperviseurs.

Dans la mesure où Red Hat OpenShift se déploie initialement avec trois nœuds maîtres, une configuration à deux nœuds risque d'entraîner l'occupation d'au moins deux maîtres du même nœud, ce qui peut entraîner une interruption possible d'OpenShift si ce nœud spécifique devient indisponible. C'est pourquoi il s'agit d'une meilleure pratique Red Hat de déployer au moins trois nœuds de calcul OSP afin que les maîtres OpenShift puissent être distribués uniformément et que la solution reçoive un degré supplémentaire de tolérance aux pannes.

Configuration de l'affinité hôte/machine virtuelle

Distribution des maîtres OpenShift sur plusieurs nœuds d'hyperviseur peut être obtenue grâce à l'affinité VM/hôte.

L'affinité est un moyen de définir des règles pour un ensemble de VM et/ou d'hôtes qui déterminent si les VM s'exécutent sur le même hôte ou sur des hôtes du groupe ou sur des hôtes différents. Elle est appliquée aux VM par la création de groupes d'affinités comprenant des VM et/ou des hôtes avec un ensemble de paramètres et de conditions identiques. Selon que les VM d'un groupe d'affinité s'exécutent sur le même hôte ou sur les hôtes du groupe ou séparément sur des hôtes différents, les paramètres du groupe d'affinités peuvent définir une affinité positive ou négative. Dans Red Hat OpenStack Platform, il est possible de créer et d'appliquer des règles d'affinité des hôtes et d'anti-affinité en créant des groupes de serveurs et en configurant des filtres de sorte que les instances déployées par Nova dans un groupe de serveurs se déploient sur différents nœuds de calcul.

Un groupe de serveurs possède un maximum de 10 instances virtuelles par défaut pour lesquelles il peut gérer le placement. Ceci peut être modifié en mettant à jour les quotas par défaut pour Nova.

> Il existe une limite stricte d'affinité/d'anti-affinité pour les groupes de serveurs OSP. S'il n'y a pas suffisamment de ressources à déployer sur des nœuds distincts ou pas assez de ressources pour permettre le partage des nœuds, la machine virtuelle ne démarre pas.

Pour configurer des groupes d'affinités, voir "Comment configurer l'affinité et la anti-affinité pour les instances OpenStack ?".

Utilisez un fichier d'installation personnalisé pour le déploiement OpenShift

IPI facilite le déploiement des clusters OpenShift via l'assistant interactif présenté plus haut dans ce document. Cependant, il est possible que vous deviez modifier certaines valeurs par défaut dans le cadre d'un déploiement de cluster.

Dans ces cas, vous pouvez exécuter et effectuer la tâche sans déployer immédiatement un cluster ; il crée alors un fichier de configuration à partir duquel le cluster peut être déployé ultérieurement. Cette approche est très utile pour modifier les valeurs par défaut des IPI ou pour déployer plusieurs clusters identiques dans votre environnement pour d'autres utilisations telles que la colocation. Pour plus d'informations sur la création d'une configuration d'installation personnalisée pour OpenShift, consultez "Red Hat OpenShift installation d'un cluster sur OpenStack avec personnalisation".

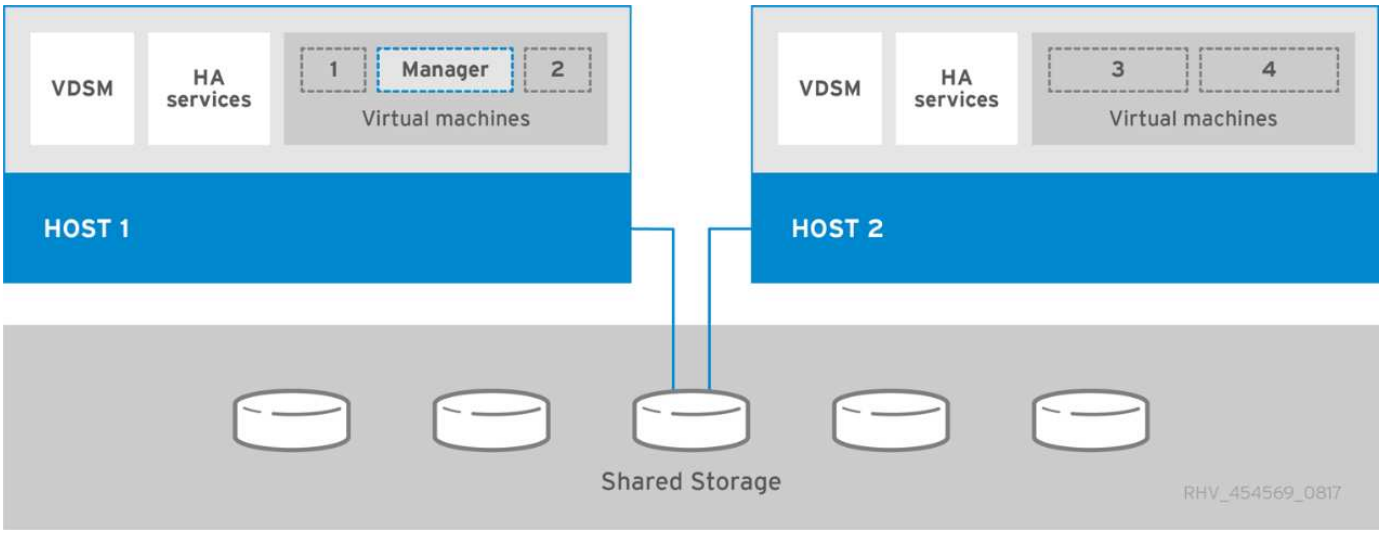
OpenShift sur Red Hat Virtualization

Red Hat Virtualization (RHV) est une plateforme de data Center virtuel d'entreprise qui s'exécute sur Red Hat Enterprise Linux (RHEL) et utilise l'hyperviseur KVM.

Pour plus d'informations sur RHV, reportez-vous au "Site Web Red Hat Virtualization".

RHV offre les caractéristiques suivantes :

- **Gestion centralisée des machines virtuelles et des hôtes** le gestionnaire RHV s'exécute en tant que machine virtuelle (VM) physique ou virtuelle (VM) dans le déploiement et fournit une interface graphique Web pour la gestion de la solution à partir d'une interface centrale.
- **Self-Hosted Engine** pour minimiser les exigences matérielles, RHV permet le déploiement de RHV Manager (RHV-M) en tant que machine virtuelle sur les mêmes hôtes qui exécutent des machines virtuelles invitées.
- **Haute disponibilité** pour éviter les interruptions en cas de défaillances de l'hôte, RHV permet de configurer les machines virtuelles pour une haute disponibilité. Les machines virtuelles haute disponibilité sont contrôlées au niveau du cluster à l'aide de règles de résilience.
- **Haute évolutivité** Un seul cluster RHV peut avoir jusqu'à 200 hôtes d'hyperviseur, ce qui lui permet de prendre en charge des machines virtuelles volumineuses pour héberger des charges de travail d'entreprise gourmandes en ressources.
- **La sécurité améliorée** héritée de RHV, les technologies Secure Virtualization (sVirt) et Security Enhanced Linux (SELinux) sont utilisées par RHV dans le but de renforcer la sécurité des hôtes et des machines virtuelles. L'avantage principal de ces fonctionnalités est l'isolation logique d'une machine virtuelle et des ressources qui lui sont associées.



Conception du réseau

La solution Red Hat OpenShift sur NetApp utilise deux switchs de données pour assurer la connectivité des données primaires à 25 Gbit/s. Il utilise également deux commutateurs de gestion supplémentaires qui fournissent une connectivité à 1 Gbit/s pour la gestion intrabande des nœuds de stockage et la gestion hors bande des fonctionnalités IPMI. OCP utilise le réseau logique de la machine virtuelle sur RHV pour la gestion des clusters. Cette section décrit l'organisation et l'objectif de chaque segment de réseau virtuel utilisé dans la solution et décrit les conditions préalables au déploiement de la solution.

Exigences VLAN

Red Hat OpenShift sur RHV est conçu pour séparer logiquement le trafic réseau à différents fins à l'aide de réseaux locaux virtuels (VLAN). Cette configuration peut être adaptée aux besoins du client ou pour assurer une isolation supplémentaire pour des services réseau spécifiques. Le tableau suivant répertorie les VLAN nécessaires à la mise en œuvre de la solution lors de sa validation chez NetApp.

VLAN	Objectif	ID VLAN
Réseau de gestion hors bande	Gestion des nœuds physiques et IPMI	16
Réseau de VM	Accès réseau invité virtuel	1172
Réseau de gestion dans la bande	Gestion des nœuds RHV-H, RHV-Manager et du réseau d'administration serveur	3343
Réseau de stockage	Réseau de stockage pour NetApp Element iSCSI	3344
Réseau de migration	Réseau pour migration invité virtuel	3345

Ressources de prise en charge de l'infrastructure réseau

L'infrastructure suivante doit être en place avant le déploiement de la plateforme de conteneurs OpenShift :

- Au moins un serveur DNS fournissant une résolution complète du nom d'hôte accessible depuis le réseau de gestion intrabande et le réseau VM.
- Au moins un serveur NTP accessible depuis le réseau de gestion intrabande et le réseau de VM.
- (Facultatif) connectivité Internet sortante pour le réseau de gestion intrabande et le réseau VM.

Bonnes pratiques pour les déploiements en production

Cette section répertorie plusieurs meilleures pratiques à prendre en considération avant de déployer cette solution en production.

Déployez OpenShift sur un cluster RHV d'au moins trois nœuds

L'architecture vérifiée décrite dans ce document présente le déploiement matériel minimum adapté aux opérations haute disponibilité en déployant deux nœuds d'hyperviseur RHV-H, et en assurant une configuration avec tolérance aux pannes dans laquelle les deux hôtes peuvent gérer le moteur hébergé et les VM déployés peuvent migrer entre les deux hyperviseurs.

Red Hat OpenShift se déployant initialement avec trois nœuds maîtres, il est garanti dans une configuration à deux nœuds qui occupera au moins deux maîtres, ce qui peut entraîner une interruption possible pour OpenShift si ce nœud spécifique devient indisponible. C'est donc une meilleure pratique de Red Hat qu'au moins trois nœuds d'hyperviseur RHV-H peuvent être déployés dans le cadre de la solution de façon à ce que les maîtres OpenShift puissent être distribués uniformément et que la solution bénéficie d'un degré de tolérance aux pannes supplémentaire.

Configuration de l'affinité hôte/machine virtuelle

Vous pouvez distribuer les maîtres OpenShift sur plusieurs nœuds d'hyperviseur en activant l'affinité VM/hôte.

L'affinité est un moyen de définir des règles pour un ensemble de VM et/ou d'hôtes qui déterminent si les VM s'exécutent sur le même hôte ou sur des hôtes du groupe ou sur des hôtes différents. Elle est appliquée aux VM par la création de groupes d'affinités comprenant des VM et/ou des hôtes avec un ensemble de paramètres et de conditions identiques. Selon que les VM d'un groupe d'affinité s'exécutent sur le même hôte ou sur les hôtes du groupe ou séparément sur des hôtes différents, les paramètres du groupe d'affinités peuvent définir une affinité positive ou négative.

Les conditions définies pour les paramètres peuvent être soit application stricte, soit application souple. Une mise en œuvre stricte permet de garantir que les VM d'un groupe d'affinité suivent toujours l'affinité positive ou négative strictement sans égard aux conditions externes. La mise en œuvre logicielle garantit qu'une préférence plus élevée est définie pour les VM d'un groupe d'affinité afin de suivre l'affinité positive ou négative lorsque cela est possible. Dans la configuration à deux ou trois hyperviseurs décrite dans ce document, soft affinité est le paramètre recommandé. Dans les clusters de plus grande taille, l'affinité matérielle peut distribuer correctement les nœuds OpenShift.

Pour configurer des groupes d'affinités, reportez-vous à la section "Red Hat 6.11. Documentation des groupes d'affinités".

Utilisez un fichier d'installation personnalisé pour le déploiement OpenShift

IPI facilite le déploiement des clusters OpenShift via l'assistant interactif présenté plus haut dans ce document. Cependant, il est possible qu'il y ait des valeurs par défaut qui devront être modifiées dans le cadre du déploiement du cluster.

Dans ces instances, vous pouvez exécuter et tâches l'assistant sans déployer immédiatement un cluster. Au contraire, un fichier de configuration est créé à partir duquel le cluster peut être déployé ultérieurement. Cette fonction s'avère très utile pour modifier les valeurs par défaut des IPI ou pour déployer plusieurs clusters identiques dans votre environnement pour d'autres utilisations telles que la colocation. Pour plus d'informations sur la création d'une configuration d'installation personnalisée pour OpenShift, consultez "Red Hat OpenShift installation d'un cluster sur RHV avec personnalisation".

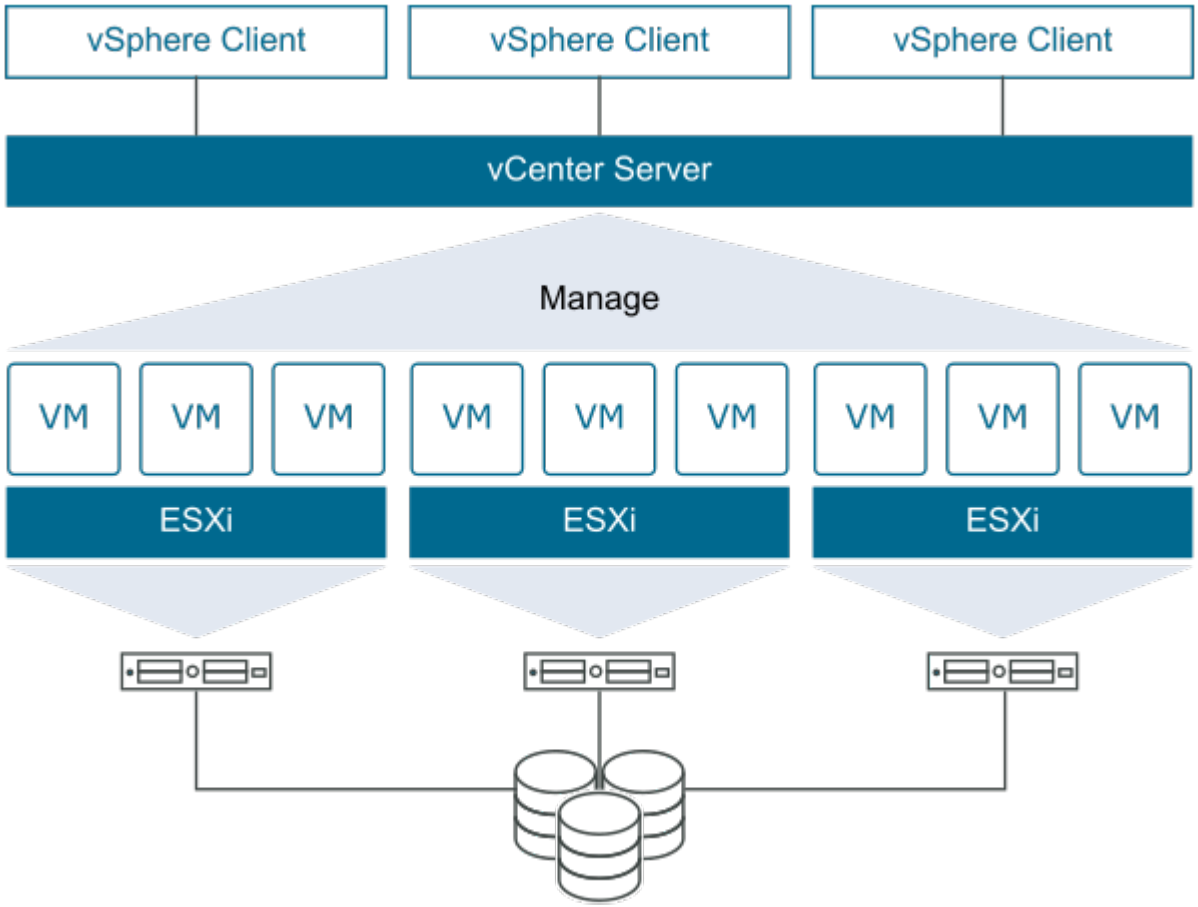
OpenShift sur VMware vSphere

VMware vSphere est une plateforme de virtualisation qui permet de gérer de manière centralisée un grand nombre de serveurs et de réseaux virtualisés exécutés sur l'hyperviseur ESXi.

Pour plus d'informations sur VMware vSphere, consultez "Site Web VMware vSphere".

VMware vSphere offre les fonctionnalités suivantes :

- **VMware vCenter Server** VMware vCenter Server assure la gestion unifiée de tous les hôtes et machines virtuelles à partir d'une seule console et agrège la surveillance des performances des clusters, des hôtes et des machines virtuelles.
- **VMware vSphere vMotion** VMware vCenter vous permet de migrer à chaud des machines virtuelles entre les nœuds du cluster sur demande, sans interruption.
- **VSphere High Availability** pour éviter les interruptions en cas de défaillance de l'hôte, VMware vSphere permet de mettre en cluster les hôtes et de les configurer pour la haute disponibilité. Les machines virtuelles interrompues par une défaillance hôte sont redémarrée prochainement sur d'autres hôtes du cluster, afin de restaurer les services.
- **Distributed Resource Scheduler (DRS)** Un cluster VMware vSphere peut être configuré pour équilibrer la charge des besoins en ressources des machines virtuelles qu'il héberge. Les machines virtuelles avec contention de ressources peuvent être migrées à chaud vers d'autres nœuds du cluster pour garantir qu'un nombre suffisant de ressources est disponible.



Conception du réseau

La solution Red Hat OpenShift sur NetApp utilise deux switchs de données pour assurer la connectivité des données primaires à 25 Gbit/s. Il utilise également deux commutateurs de gestion supplémentaires qui fournissent une connectivité à 1 Gbit/s pour la gestion intrabande des nœuds de stockage et la gestion hors bande des fonctionnalités IPMI. OCP utilise le réseau logique VM sur VMware vSphere pour la gestion de son cluster. Cette section décrit l'organisation et l'objectif de chaque segment de réseau virtuel utilisé dans la solution et décrit les conditions préalables au déploiement de la solution.

Exigences VLAN

Red Hat OpenShift sur VMware vSphere est conçu pour séparer logiquement le trafic réseau à différents fins à l'aide de réseaux locaux virtuels (VLAN). Cette configuration peut être adaptée aux besoins du client ou pour assurer une isolation supplémentaire pour des services réseau spécifiques. Le tableau suivant répertorie les VLAN nécessaires à la mise en œuvre de la solution lors de sa validation chez NetApp.

VLAN	Objectif	ID VLAN
Réseau de gestion hors bande	Gestion des nœuds physiques et IPMI	16
Réseau de VM	Accès réseau invité virtuel	181
Réseau de stockage	Réseau de stockage pour ONTAP NFS	184
Réseau de stockage	Réseau de stockage pour ONTAP iSCSI	185
Réseau de gestion dans la bande	Gestion des nœuds ESXi, vCenter Server, ONTAP Select	3480
Réseau de stockage	Réseau de stockage pour NetApp Element iSCSI	3481
Réseau de migration	Réseau pour migration invité virtuel	3482

Ressources de prise en charge de l'infrastructure réseau

L'infrastructure suivante doit être en place avant le déploiement de la plateforme de conteneurs OpenShift :

- Au moins un serveur DNS fournissant une résolution complète du nom d'hôte accessible depuis le réseau de gestion intrabande et le réseau VM.
- Au moins un serveur NTP accessible depuis le réseau de gestion intrabande et le réseau de VM.
- (Facultatif) connectivité Internet sortante pour le réseau de gestion intrabande et le réseau VM.

Bonnes pratiques pour les déploiements en production

Cette section répertorie plusieurs meilleures pratiques à prendre en considération avant de déployer cette solution en production.

Déployez OpenShift sur un cluster ESXi d'au moins trois nœuds

L'architecture vérifiée dans ce document présente le déploiement matériel minimum adapté aux opérations haute disponibilité en déployant deux nœuds d'hyperviseur ESXi et en assurant une configuration avec tolérance aux pannes en activant VMware vSphere HA et VMware vMotion. Cette configuration permet aux VM déployées de migrer entre les deux hyperviseurs et de redémarrer en cas d'indisponibilité d'un hôte.

Red Hat OpenShift se déploie initialement avec trois nœuds maîtres, au moins deux maîtres dans une configuration à deux nœuds peuvent occuper le même nœud dans certains cas, ce qui peut entraîner une

interruption possible pour OpenShift si ce nœud spécifique devient indisponible. C'est donc une meilleure pratique Red Hat qu'au moins trois nœuds d'hyperviseur ESXi doivent être déployés de manière à ce que les maîtres OpenShift puissent être répartis de façon homogène, ce qui offre un degré supplémentaire de tolérance aux pannes.

Configuration de l'affinité des hôtes et des machines virtuelles

Assurer la distribution des maîtres OpenShift sur plusieurs nœuds d'hyperviseur peut être obtenue grâce à l'activation des VM et de l'affinité des hôtes.

Une affinité ou une anti-affinité permet de définir des règles pour un ensemble de VM et/ou d'hôtes qui déterminent si les VM s'exécutent sur le même hôte ou sur des hôtes du groupe ou sur des hôtes différents. Elle est appliquée aux VM par la création de groupes d'affinités comprenant des VM et/ou des hôtes avec un ensemble de paramètres et de conditions identiques. Selon que les VM d'un groupe d'affinité s'exécutent sur le même hôte ou sur les hôtes du groupe ou séparément sur des hôtes différents, les paramètres du groupe d'affinités peuvent définir une affinité positive ou négative.

Pour configurer des groupes d'affinités, reportez-vous à la section "Documentation vSphere 6.7 : utilisation des règles d'affinité DRS".

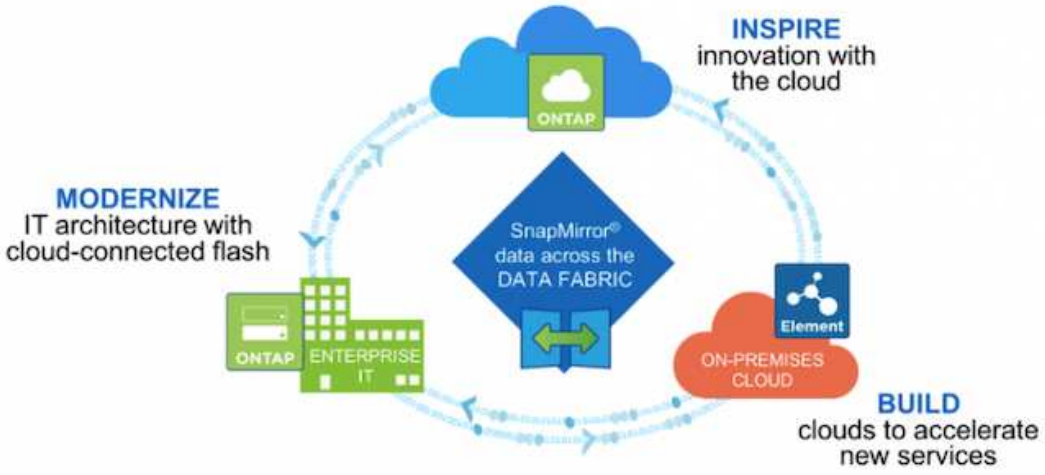
Utilisez un fichier d'installation personnalisé pour le déploiement OpenShift

IPI facilite le déploiement des clusters OpenShift via l'assistant interactif présenté plus haut dans ce document. Cependant, il est possible que vous deviez modifier certaines valeurs par défaut dans le cadre d'un déploiement de cluster.

Dans ces instances, vous pouvez exécuter et exécuter l'assistant sans déployer immédiatement un cluster, mais à la place, l'assistant crée un fichier de configuration à partir duquel le cluster peut être déployé ultérieurement. Cette approche est très utile pour modifier les paramètres par défaut des IPI ou pour déployer plusieurs clusters identiques dans votre environnement à des fins autres que la colocation. Pour plus d'informations sur la création d'une configuration d'installation personnalisée pour OpenShift, consultez "Red Hat OpenShift installation d'un cluster sur vSphere avec personnalisation".

Présentation du stockage NetApp

NetApp propose plusieurs plateformes de stockage compatibles avec notre orchestrateur de stockage Astra Trident qui sert à provisionner le stockage pour les applications déployées sur Red Hat OpenShift.



- Les systèmes AFF et FAS exécutent NetApp ONTAP et fournissent aussi bien le stockage en mode fichier (NFS) que en mode bloc (iSCSI).
- Cloud Volumes ONTAP et ONTAP Select offrent les mêmes avantages, respectivement, dans le cloud et dans l'espace virtuel.
- NetApp Cloud Volumes Service (AWS/GCP) et Azure NetApp Files proposent un stockage basé sur des fichiers dans le cloud.
- Les systèmes de stockage NetApp Element fournissent des cas d'utilisation basés sur les blocs (iSCSI) dans un environnement hautement évolutif.

> Chaque système de stockage du portefeuille NetApp simplifie la gestion et le déplacement des données entre les sites sur site et le cloud, ce qui vous permet d'assurer que vos données sont là où sont vos applications.

Les pages suivantes présentent des informations supplémentaires sur les systèmes de stockage NetApp validés dans la solution Red Hat OpenShift avec NetApp :

- "NetApp ONTAP"
- "NetApp Element"

NetApp ONTAP

NetApp ONTAP est un puissant outil de gestion du stockage. Il inclut des fonctionnalités telles qu'une interface graphique intuitive, des API REST avec intégration de l'automatisation, des analyses prédictives et des actions correctives informées par IA, des mises à niveau matérielles sans interruption et des importations intersystèmes de stockage.

Pour en savoir plus sur la baie de stockage NetApp ONTAP, consultez la "Site Web NetApp ONTAP".

ONTAP offre les fonctionnalités suivantes :

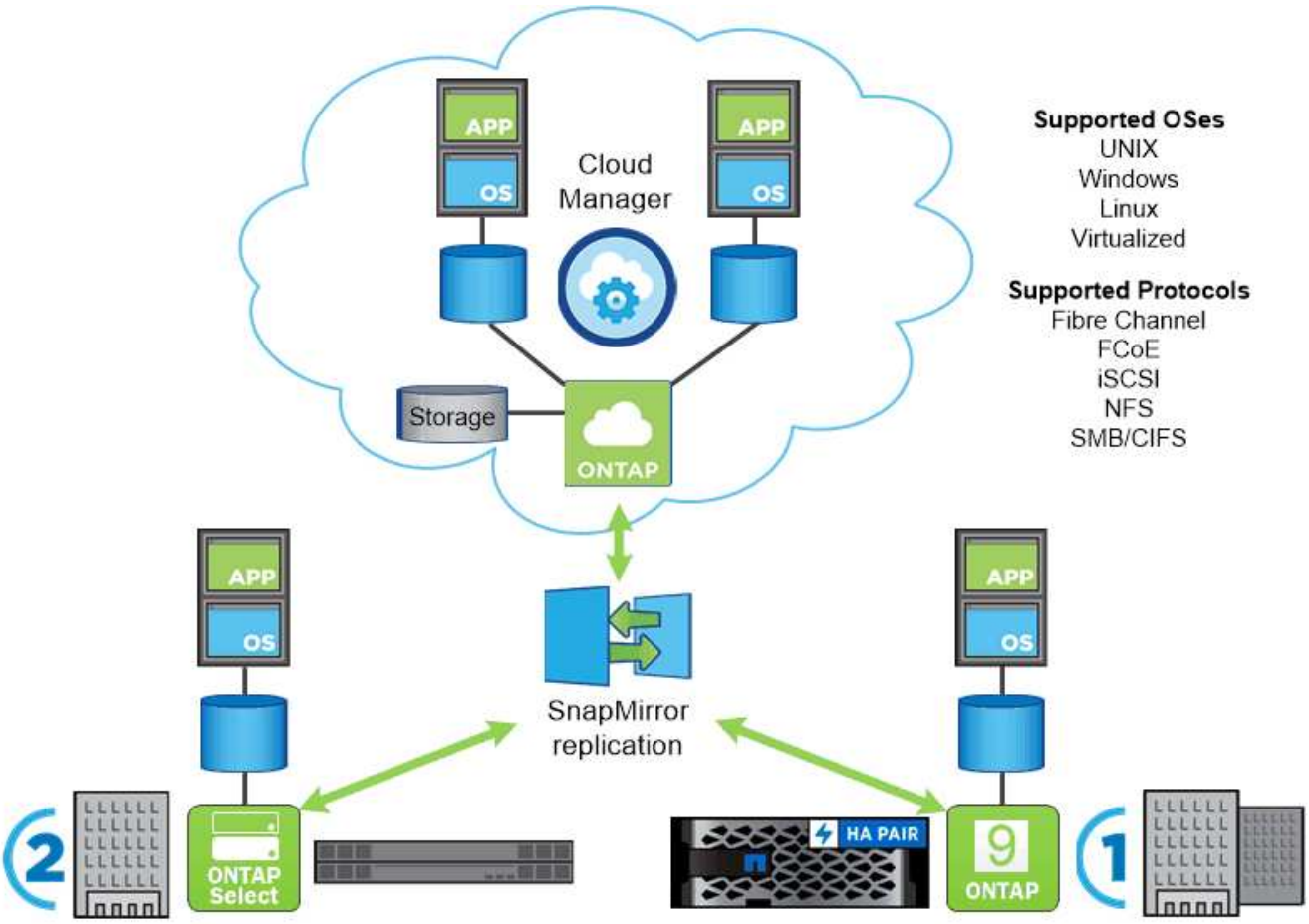
- Système de stockage unifié avec accès et gestion simultanés aux données de NFS, CIFS, iSCSI, FC,

FCoE, Et les protocoles FC-NVMe.

- Différents modèles de déploiement incluent des configurations matérielles sur site 100 % Flash, hybrides et 100 % HDD, des plateformes de stockage basées sur des VM sur un hyperviseur pris en charge comme ONTAP Select, et dans le cloud comme Cloud Volumes ONTAP.
- Amélioration de l'efficacité du stockage des données sur les systèmes ONTAP avec la prise en charge du Tiering automatique des données, de la compression des données à la volée, de la déduplication et de la compaction.
- Stockage basé sur la charge de travail, contrôlé par QoS.
- Intégration transparente avec le cloud public pour le Tiering et la protection des données. ONTAP fournit également des fonctionnalités robustes de protection des données qui le distinguent dans tous les environnements :
  - **Copies NetApp Snapshot.** sauvegarde instantanée rapide des données en utilisant un espace disque minimal, sans impact supplémentaire sur les performances.
  - **NetApp SnapMirror.** miroir les copies Snapshot des données d'un système de stockage à un autre. ONTAP prend également en charge la mise en miroir des données vers d'autres plateformes physiques et des services clouds natifs.
  - **SnapLock de NetApp.** pour une administration efficace des données non réinscriptibles, en les écrivant sur des volumes spéciaux qui ne peuvent pas être écrasés ou effacés pour une période déterminée.
  - **NetApp SnapVault.** sauvegarde les données de plusieurs systèmes de stockage sur une copie Snapshot centrale qui sert de sauvegarde à tous les systèmes désignés.
  - **NetApp SyncMirror.** permet la mise en miroir des données en temps réel au niveau RAID sur deux plexes différents de disques connectés physiquement au même contrôleur.
  - **NetApp SnapRestore** permet une restauration rapide des données sauvegardées à la demande à partir de copies Snapshot.
  - **NetApp FlexClone.** assure le provisionnement instantané d'une copie lisible et inscriptible d'un volume NetApp à partir d'une copie Snapshot.

Pour plus d'informations sur ONTAP, consultez le "Centre de documentation ONTAP 9".

NetApp ONTAP est disponible sur site, virtualisé ou dans le cloud.



Plateformes NetApp

NetApp AFF/FAS

NetApp offre des AFF plateformes de stockage FAS (100 % Flash) et scale-out, sur mesure et dotées d'une faible latence, d'une protection des données intégrée et d'une prise en charge multiprotocole.

Ces deux systèmes sont optimisés par le logiciel de gestion des données NetApp ONTAP, le logiciel de gestion des données le plus avancé du secteur pour une gestion du stockage simplifiée, intégrée au cloud et hautement disponible. Il offre la vitesse, l'efficacité et la sécurité dont votre environnement Data Fabric a besoin.

Pour en savoir plus sur les plateformes NetApp AFF/FAS, cliquez "ici".

ONTAP Select

ONTAP Select est un déploiement Software-defined de NetApp ONTAP qui peut être déployé sur un hyperviseur de votre environnement. Installée sur VMware vSphere ou KVM, cette solution permet de bénéficier de toutes les fonctionnalités et de l'expérience d'un système matériel ONTAP.

Pour plus d'informations sur ONTAP Select, cliquez sur "ici".

Cloud Volumes ONTAP

NetApp Cloud Volumes ONTAP est une version de NetApp ONTAP déployée dans le cloud et qui peut être

déployée dans plusieurs clouds publics, notamment Amazon AWS, Microsoft Azure et Google Cloud.

Pour plus d'informations sur Cloud Volumes ONTAP, cliquez sur "ici".

NetApp Element : Red Hat OpenShift avec NetApp

Le logiciel NetApp Element offre des performances modulaires et évolutives, avec chaque nœud de stockage, qui garantissent la capacité et le débit à l'environnement. Les systèmes NetApp Element peuvent évoluer de 4 à 100 nœuds dans un seul cluster et offrir de nombreuses fonctionnalités avancées de gestion du stockage.

Pour plus d'informations sur les systèmes de stockage NetApp Element, consultez la "Site Web NetApp SolidFire".

Fonctionnalités de redirection de connexion iSCSI et d'auto-rétablissement

Le logiciel NetApp Element s'appuie sur le protocole de stockage iSCSI, une méthode standard pour encapsuler les commandes SCSI sur un réseau TCP/IP traditionnel. Lorsque les normes SCSI changent ou que les performances des réseaux Ethernet s'améliorent, le protocole de stockage iSCSI est avantageux sans qu'il soit nécessaire de procéder à des modifications.

Bien que tous les nœuds de stockage aient une adresse IP de gestion et une adresse IP de stockage, le logiciel NetApp Element annonce une adresse IP virtuelle de stockage unique (adresse SVIP) pour l'ensemble du trafic de stockage du cluster. Dans le cadre du processus de connexion iSCSI, le stockage peut répondre que le volume cible a été déplacé vers une autre adresse et qu'il ne peut donc pas poursuivre le processus de négociation. L'hôte réémet alors la demande de connexion vers la nouvelle adresse dans un processus qui ne nécessite aucune reconfiguration côté hôte. Ce processus est connu sous le nom de redirection de connexion iSCSI.

La redirection de connexion iSCSI est un élément clé du cluster logiciel NetApp Element. En cas de réception d'une requête de connexion d'hôte, le nœud décide quel membre du cluster doit gérer le trafic en fonction des IOPS et des exigences de capacité du volume. Les volumes sont répartis sur le cluster logiciel NetApp Element et sont redistribués si un seul nœud traite un trop grand trafic pour ses volumes ou si un nouveau nœud est ajouté. Plusieurs copies d'un volume donné sont allouées à travers la baie.

Ainsi, si une défaillance de nœud est suivie d'une redistribution du volume, la connectivité hôte n'a aucun effet au-delà d'une déconnexion et d'une connexion avec redirection vers le nouvel emplacement. Avec la redirection de connexion iSCSI, un cluster logiciel NetApp Element est une architecture scale-out autoréparatrice qui permet des mises à niveau et des opérations sans interruption.

Qualité de service du cluster logiciel NetApp Element

Un cluster logiciel NetApp Element permet la configuration dynamique de la QoS par volume. Vous pouvez utiliser les paramètres QoS par volume pour contrôler les performances du stockage en fonction des SLA que vous définissez. Les trois paramètres configurables suivants définissent la QoS :

- **IOPS minimum.** nombre minimum d'IOPS soutenues que le cluster logiciel NetApp Element fournit à un volume. La valeur d'IOPS minimale configurée pour un volume correspond au niveau garanti de performance d'un volume. La performance par volume ne descend pas en dessous de ce niveau.
- **Nombre maximal d'IOPS.** nombre maximal d'IOPS soutenu que le cluster logiciel NetApp Element fournit à un volume donné.
- **IOPS en rafale.** le nombre maximal d'IOPS autorisé dans un scénario en rafale courte. Le paramètre de durée de rafale est configurable, avec une valeur par défaut de 1 minute. Si un volume a été exécuté en dessous du niveau d'IOPS maximal, les crédits de bursting sont accumulés. Lorsque les niveaux de performance deviennent très élevés et sont poussés, les pics d'IOPS en dehors des IOPS maximales sont autorisés sur le volume.

Colocation

La colocation sécurisée offre les fonctionnalités suivantes :

- **Authentification sécurisée.** le protocole CHAP (Challenge-Handshake Authentication Protocol) est utilisé pour sécuriser l'accès au volume. Le protocole LDAP (Lightweight Directory Access Protocol) est utilisé pour sécuriser l'accès au cluster à des fins de gestion et de reporting.
- **Groupes d'accès de volume (VAGs).** si vous le souhaitez, les VAGs peuvent être utilisés à la place de l'authentification, mappant n'importe quel nombre de noms iSCSI qualifiés (IQN) spécifiques à un initiateur iSCSI à un ou plusieurs volumes. Pour accéder à un volume dans un VAG, l'IQN de l'initiateur doit figurer dans la liste IQN autorisé pour le groupe de volumes.
- **Réseaux locaux virtuels (VLAN) locataires.** au niveau du réseau, la sécurité réseau de bout en bout entre les initiateurs iSCSI et le cluster logiciel NetApp Element est facilitée par l'utilisation de VLAN. Pour tout VLAN créé pour isoler une charge de travail ou un locataire, le logiciel NetApp Element crée une adresse SVIP cible iSCSI distincte accessible uniquement via le VLAN spécifique.
- **VLAN activés par VRF** pour prendre en charge encore plus la sécurité et l'évolutivité dans le data Center, le logiciel NetApp Element vous permet d'activer n'importe quel VLAN locataire pour les fonctionnalités de type VRF. Cette fonctionnalité offre deux fonctionnalités clés :
  - **Routage L3 vers une adresse SVIP locataire.** cette fonctionnalité vous permet de situer les initiateurs iSCSI sur un réseau ou VLAN séparé de celui du cluster logiciel NetApp Element.
  - **Sous-réseaux IP redondants ou dupliqués.** cette fonctionnalité vous permet d'ajouter un modèle aux environnements de tenant, permettant à chaque VLAN locataire respectif d'être affectés à des adresses IP à partir du même sous-réseau IP. Cette fonctionnalité peut être utile pour les environnements de fournisseurs de services où l'évolutivité et la préservation de l'IPspace sont importantes.

Fonctionnalités d'efficacité du stockage

Le cluster logiciel NetApp Element améliore les performances et l'efficacité de stockage globales. Les fonctionnalités suivantes sont effectuées en ligne, sont toujours disponibles et ne nécessitent aucune configuration manuelle de la part de l'utilisateur :

- **Déduplication.** le système stocke uniquement des blocs 4K uniques. Tous les blocs de 4 Ko dupliqués sont automatiquement associés à une version déjà stockée des données. Les données se trouvent sur des disques de niveau bloc et sont mises en miroir à l'aide du logiciel NetApp Element, protection des données Helix. Ce système réduit considérablement la consommation de capacité et les opérations d'écriture dans le système.
- **Compression.** la compression est effectuée en ligne avant que les données ne soient écrites dans la NVRAM. Les données sont compressées, stockées sous forme de blocs de 4 Ko, et restent compressées dans le système. Cette compression réduit considérablement la consommation de capacité, les opérations

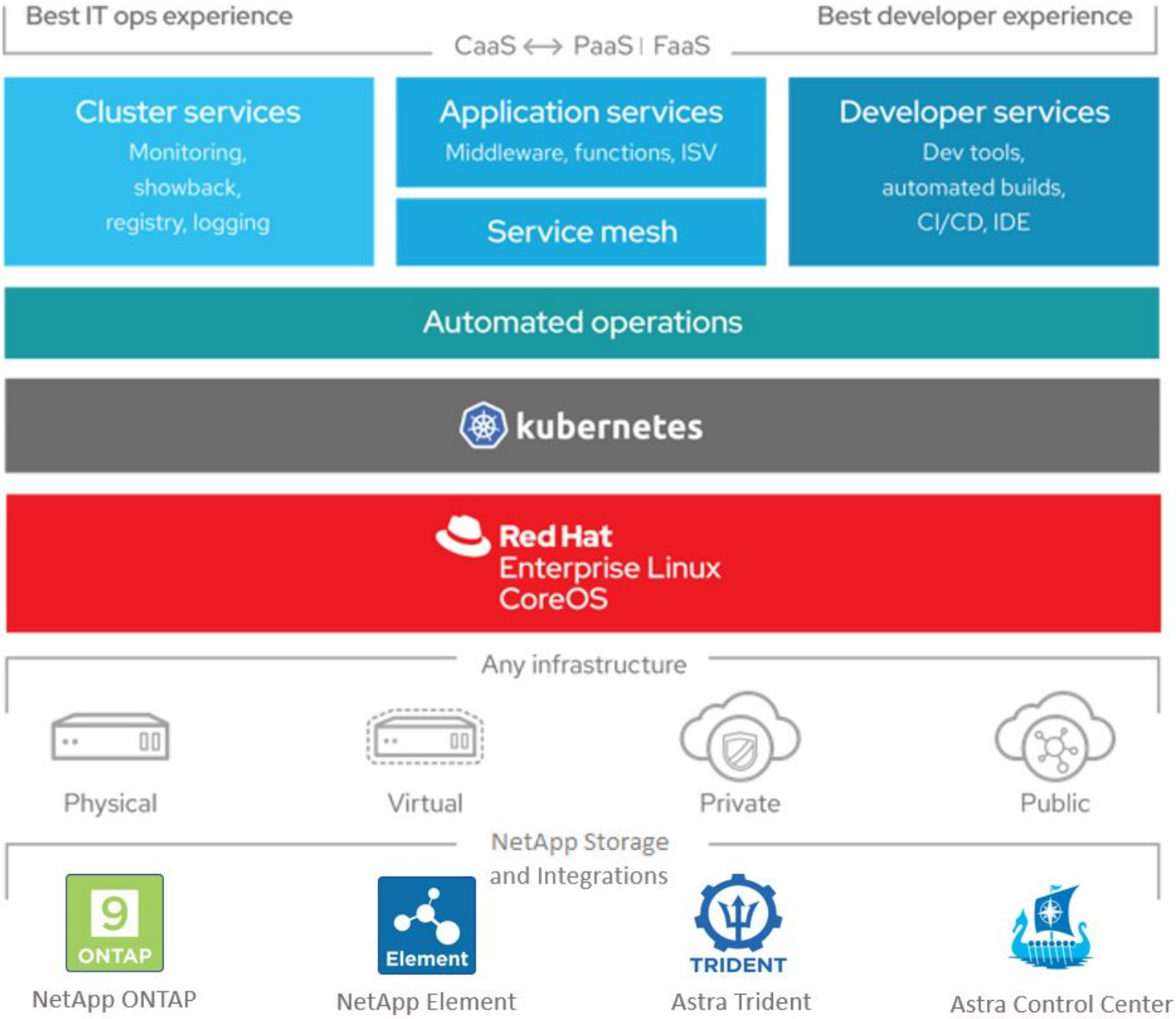
d'écriture et la consommation de bande passante dans le cluster.

- **Provisionnement fin.** cette fonctionnalité fournit la quantité de stockage appropriée au moment où vous en avez besoin, ce qui élimine la consommation de capacité provoquée par des volumes surprovisionnés ou sous-exploités.
- **Helix.** les métadonnées d'un volume individuel sont stockées sur un lecteur de métadonnées et sont répliquées sur un lecteur de métadonnées secondaire pour assurer la redondance.

> Element a été conçu pour l'automatisation. Toutes les fonctionnalités de stockage sont disponibles par le biais d'API. Ces API sont la seule méthode que l'interface utilisateur utilise pour contrôler le système.

Présentation de l'intégration du stockage NetApp

NetApp propose plusieurs produits pour orchestrer et gérer les données persistantes dans des environnements basés sur des conteneurs, tels que Red Hat OpenShift.



NetApp Astra Control propose un ensemble complet de services de gestion du stockage et des données

respectueuse des applications pour les workloads Kubernetes avec état, optimisés par la technologie de protection des données NetApp. Astra Control Service est disponible pour la prise en charge des workloads avec état dans les déploiements Kubernetes cloud natifs. Le centre de contrôle Astra est disponible pour les workloads avec état dans les déploiements sur site, tels que Red Hat OpenShift. Pour en savoir plus, rendez-vous sur le site Web NetApp Astra Control "ici".

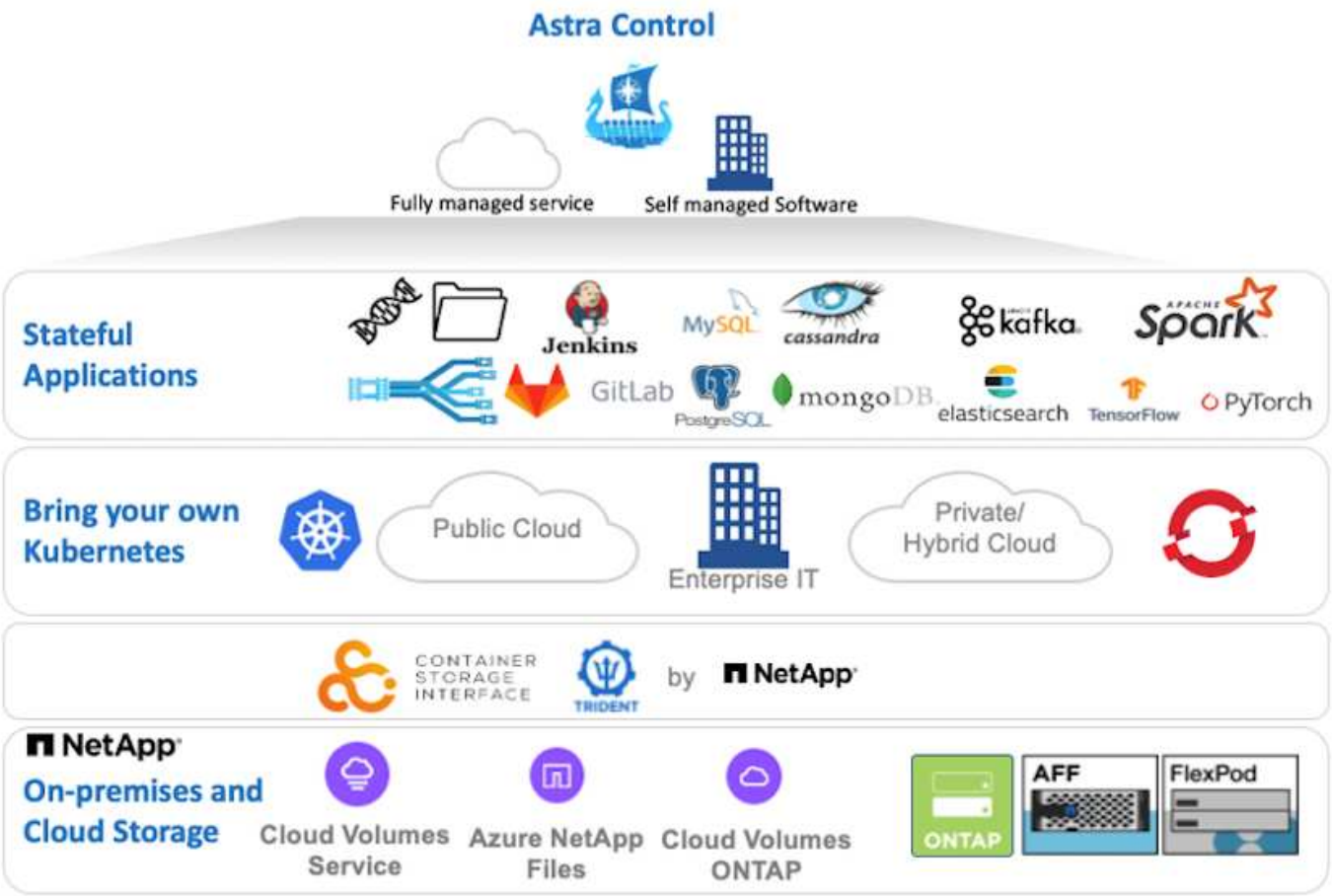
NetApp Astra Trident est un orchestrateur de stockage open source entièrement pris en charge pour les conteneurs et les distributions Kubernetes, y compris Red Hat OpenShift. Pour en savoir plus, rendez-vous sur le site Web Astra Trident "ici".

Les pages suivantes présentent des informations supplémentaires sur les produits NetApp validés pour la gestion du stockage persistant et des applications dans la solution Red Hat OpenShift avec NetApp :

- "NetApp Astra Control Center"
- "NetApp Astra Trident"

Présentation de NetApp Astra Control Center

NetApp Astra Control Center propose un ensemble complet de services de gestion du stockage et des données respectueuse des applications pour les workloads Kubernetes avec état, déployés dans un environnement sur site et optimisé par les technologies NetApp de protection des données.



NetApp Astra Control Center peut être installé sur un cluster Red Hat OpenShift que l'orchestrateur de stockage Astra Trident est déployé et configuré avec des classes de stockage et des systèmes back-end de stockage dans des systèmes de stockage NetApp ONTAP.

Pour l'installation et la configuration d'Astra Trident pour prendre en charge Astra Control Center, voir "ce document ici".

Dans un environnement connecté au cloud, Astra Control Center utilise Cloud Insights pour fournir des fonctionnalités avancées de surveillance et de télémétrie. En l'absence de connexion Cloud Insights, un contrôle limité et une télémétrie (valeurs de metrics de 7 jours) sont disponibles et exportés vers les outils de contrôle natifs Kubernetes (Prometheus et Grafana) via des terminaux ouverts.

Le centre de contrôle Astra est entièrement intégré à l'écosystème NetApp AutoSupport et Active IQ qui fournit un soutien aux utilisateurs, fournit des conseils pour la résolution de problèmes et affiche des statistiques d'utilisation.

En plus de la version payante d'Astra Control Center, une licence d'évaluation de 90 jours est disponible. La version d'évaluation est prise en charge par e-mail et par la communauté (Channel Slack). Les clients ont accès à ces articles, ainsi qu'à d'autres articles de la base de connaissances, et à la documentation disponible dans le tableau de bord de support des produits.

Pour commencer avec NetApp Astra Control Center, rendez-vous sur le "Site Web d'Astra".

Conditions préalables à l'installation d'Astra Control Center

1. Un ou plusieurs clusters Red Hat OpenShift. Les versions 4.6 EUS et 4.7 sont actuellement prises en charge.
2. Astra Trident doit déjà être installé et configuré sur chaque cluster Red Hat OpenShift.
3. Un ou plusieurs systèmes de stockage NetApp ONTAP exécutant ONTAP 9.5 ou version ultérieure.

   > Il est recommandé que chaque installation OpenShift sur un site dispose d'un SVM dédié pour le stockage persistant. Les déploiements multisites requièrent des systèmes de stockage supplémentaires.

4. Un système back-end de stockage Trident doit être configuré sur chaque cluster OpenShift avec un SVM sauvegardé par un cluster ONTAP.
5. Classe de stockage par défaut configurée sur chaque cluster OpenShift avec Astra Trident comme provisionneur de stockage.
6. Un équilibreur de charge doit être installé et configuré sur chaque cluster OpenShift pour équilibrer les charges et exposer les services OpenShift.

   > Voir le lien "ici" pour plus d'informations sur les équilibreurs de charge qui ont été validés à cet effet.

7. Un registre d'images privées doit être configuré pour héberger les images du NetApp Astra Control Center.

   > Voir le lien "ici" Pour installer et configurer un registre privé OpenShift à cet effet.

8. Vous devez disposer d'un accès Cluster Admin au cluster Red Hat OpenShift.
9. Vous devez disposer d'un accès d'administration aux clusters NetApp ONTAP.
10. Une station de travail d'administration avec docker ou podman, tridentctl et oc ou kubectl a été installée et ajoutée à votre $PATH

Les installations Docker doivent avoir une version docker supérieure à 20.10 et les installations Podman doivent avoir une version podman supérieure à 3.0.

Poser le centre de contrôle Astra

Utilisation de OperatorHub

1. Connectez-vous au site de support NetApp et téléchargez la dernière version de NetApp Astra Control Center. Une licence associée à votre compte NetApp est requise. Après avoir téléchargé le fichier tarball, transférez-le sur le poste de travail d'administration.

   > Pour commencer avec une licence d'essai d'Astra Control, visitez le "Site d'inscription à Astra".

2. Déballez la boule tar et remplacez le répertoire de travail par le dossier obtenu.

   ```
   [netapp-user@rhel7 ~]$ tar -vxzf astra-control-center-
   21.12.60.tar.gz
   [netapp-user@rhel7 ~]$ cd astra-control-center-21.12.60
   ```

3. Avant de commencer l'installation, poussez les images du centre de contrôle Astra vers un registre d'images. Vous pouvez choisir de le faire avec Docker ou Podman, les instructions pour les deux sont fournies dans cette étape.

Podman

a. Exportez le FQDN du Registre avec le nom de l'organisation/espace de noms/projet comme variable d'environnement 'regiant'.

```
[netapp-user@rhel7 ~]$ export REGISTRY=astra-
registry.apps.ocp-vmw.cie.netapp.com/netapp-astra
```

b. Connectez-vous au registre.

```
[netapp-user@rhel7 ~]$ podman login -u ocp-user -p password
--tls-verify=false astra-registry.apps.ocp-vmw.cie.netapp.com
```

> Si vous utilisez `kubeadmin` l'utilisateur doit se connecter au registre privé, puis utiliser un jeton au lieu du mot de passe - `podman login -u ocp-user -p token --tls-verify=false astra-registry.apps.ocp-vmw.cie.netapp.com`.

> Vous pouvez également créer un compte de service, attribuer un rôle d'éditeur de registre et/ou de visualiseur de registre (selon que vous avez besoin d'un accès Push/Pull) et vous connecter au registre à l'aide du jeton du compte de service.

c. Créez un fichier de script shell et collez le contenu suivant dans celui-ci.

```
[netapp-user@rhel7 ~]$ vi push-images-to-registry.sh

for astraImageFile in $(ls images/*.tar) ; do
  # Load to local cache. And store the name of the loaded
image trimming the 'Loaded images: '
  astraImage=$(podman load --input ${astraImageFile} | sed
's/Loaded image(s): //')
  astraImage=$(echo ${astraImage} | sed 's!localhost/!!')
  # Tag with local image repo.
  podman tag ${astraImage} ${REGISTRY}/${astraImage}
  # Push to the local repo.
  podman push ${REGISTRY}/${astraImage}
done
```

> Si vous utilisez des certificats non approuvés pour votre registre, modifiez le script de shell et utilisez-le `--tls-verify=false` pour la commande push podman `podman push $REGISTRY/$(echo $astraImage | sed 's/[\/]\+\///') --tls-verify=false`.

d. Rendre le fichier exécutable.

```
[netapp-user@rhel7 ~]$ chmod +x push-images-to-registry.sh
```

e. Exécutez le script de shell.

```
[netapp-user@rhel7 ~]$ ./push-images-to-registry.sh
```

Docker

a. Exportez le FQDN du Registre avec le nom de l'organisation/espace de noms/projet comme variable d'environnement 'regiant'.

```
[netapp-user@rhel7 ~]$ export REGISTRY=astra-
registry.apps.ocp-vmw.cie.netapp.com/netapp-astra
```

b. Connectez-vous au registre.

```
[netapp-user@rhel7 ~]$ docker login -u ocp-user -p password
astra-registry.apps.ocp-vmw.cie.netapp.com
```

> Si vous utilisez `kubeadmin` l'utilisateur doit se connecter au registre privé, puis utiliser un jeton au lieu du mot de passe - `docker login -u ocp-user -p token astra-registry.apps.ocp-vmw.cie.netapp.com`.

> Vous pouvez également créer un compte de service, attribuer un rôle d'éditeur de registre et/ou de visualiseur de registre (selon que vous avez besoin d'un accès Push/Pull) et vous connecter au registre à l'aide du jeton du compte de service.

c. Créez un fichier de script shell et collez le contenu suivant dans celui-ci.

```
[netapp-user@rhel7 ~]$ vi push-images-to-registry.sh

for astraImageFile in $(ls images/*.tar) ; do
  # Load to local cache. And store the name of the loaded
image trimming the 'Loaded images: '
  astraImage=$(docker load --input ${astraImageFile} | sed
's/Loaded image: //')
  astraImage=$(echo ${astraImage} | sed 's!localhost/!!')
  # Tag with local image repo.
  docker tag ${astraImage} ${REGISTRY}/${astraImage}
  # Push to the local repo.
  docker push ${REGISTRY}/${astraImage}
done
```

d. Rendre le fichier exécutable.

```
[netapp-user@rhel7 ~]$ chmod +x push-images-to-registry.sh
```

e. Exécutez le script de shell.

```
[netapp-user@rhel7 ~]$ ./push-images-to-registry.sh
```

4. Lorsque vous utilisez des registres d'images privés qui ne sont pas de confiance publique, chargez les certificats TLS du registre d'images sur les nœuds OpenShift. Pour ce faire, créez une config map dans l'espace de noms openshift-config à l'aide des certificats TLS et installez-la sur la configuration d'images du cluster pour que le certificat soit fiable.

```
[netapp-user@rhel7 ~]$ oc create configmap default-ingress-ca -n
openshift-config --from-file=astra-registry.apps.ocp
-vmw.cie.netapp.com=tls.crt

[netapp-user@rhel7 ~]$ oc patch image.config.openshift.io/cluster
--patch '{"spec":{"additionalTrustedCA":{"name":"default-ingress-
ca"}}}' --type=merge
```

> Si vous utilisez un registre interne OpenShift avec des certificats TLS par défaut de l'opérateur d'entrée portant une route, vous devez suivre l'étape précédente pour corriger le nom d'hôte de la route. Pour extraire les certificats de l'opérateur Ingress, vous pouvez utiliser la commande `oc extract secret/router-ca --keys=tls.crt -n openshift-ingress-operator`.

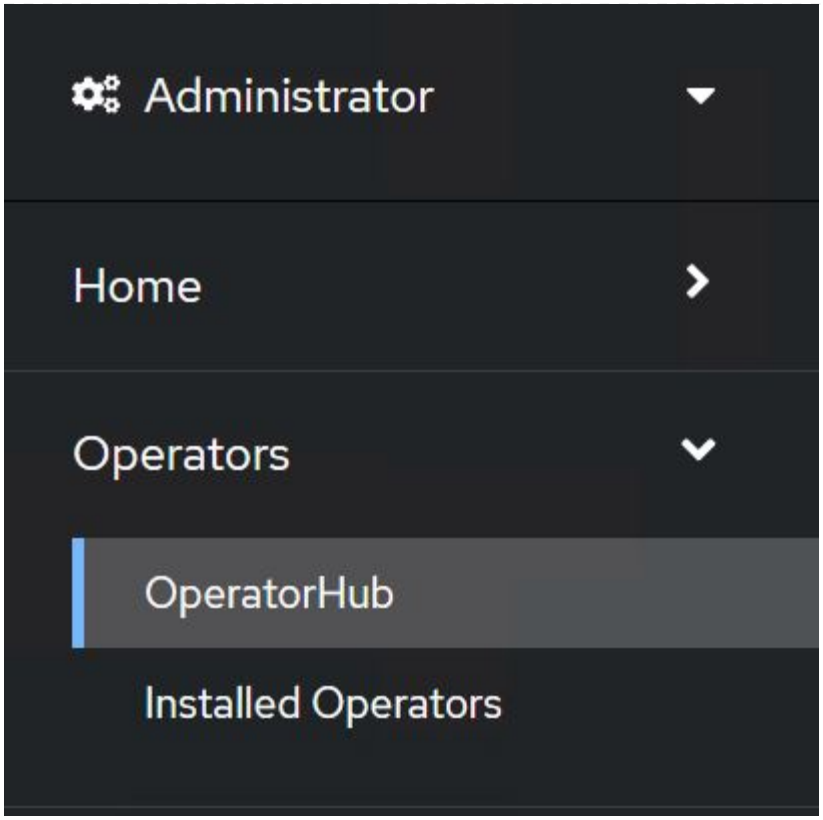
5. Créer un espace de noms `netapp-acc-operator` Pour Astra Control Center.

```
[netapp-user@rhel7 ~]$ oc create ns netapp-acc-operator

namespace/netapp-acc-operator created
```

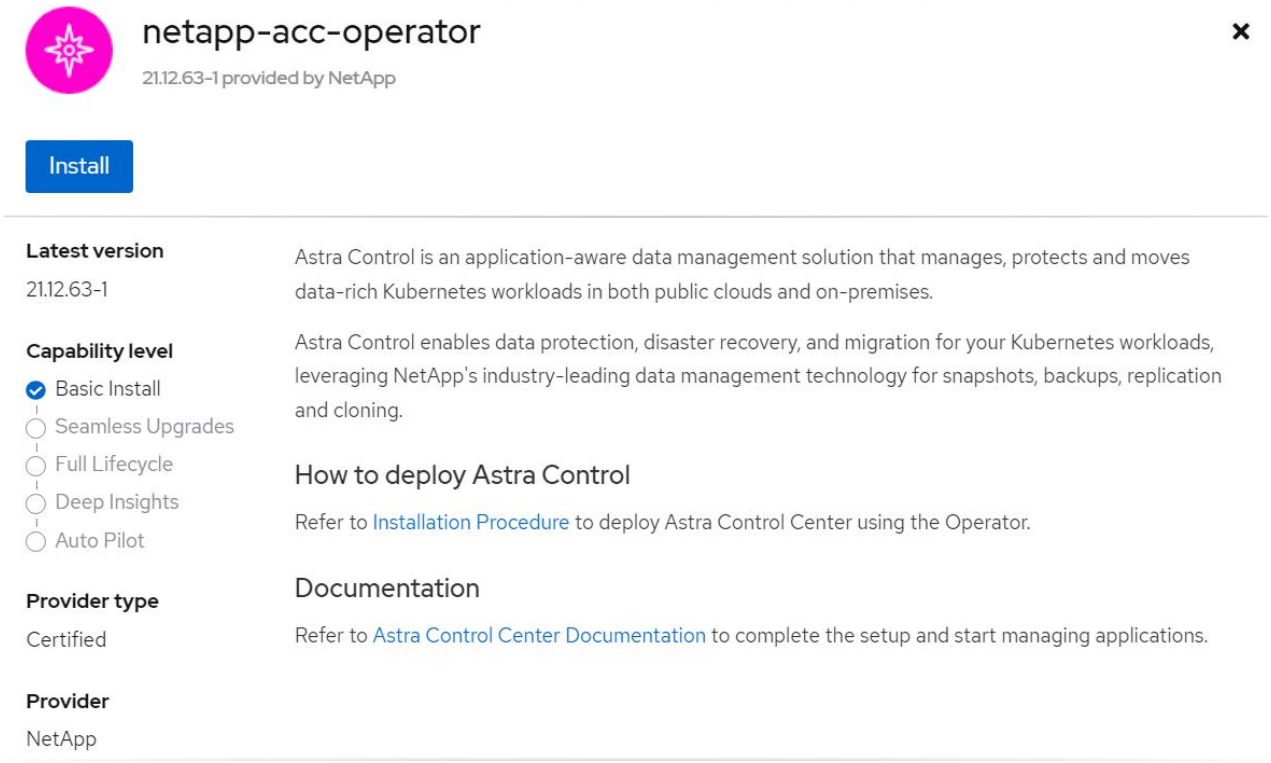
6. Créez un secret avec des informations d'identification pour vous connecter au registre d'images dans `netapp-acc-operator` espace de noms.

```
[netapp-user@rhel7 ~]$ oc create secret docker-registry astra-
registry-cred --docker-server=astra-registry.apps.ocp
-vmw.cie.netapp.com --docker-username=ocp-user --docker
-password=password -n netapp-acc-operator

secret/astra-registry-cred created
```

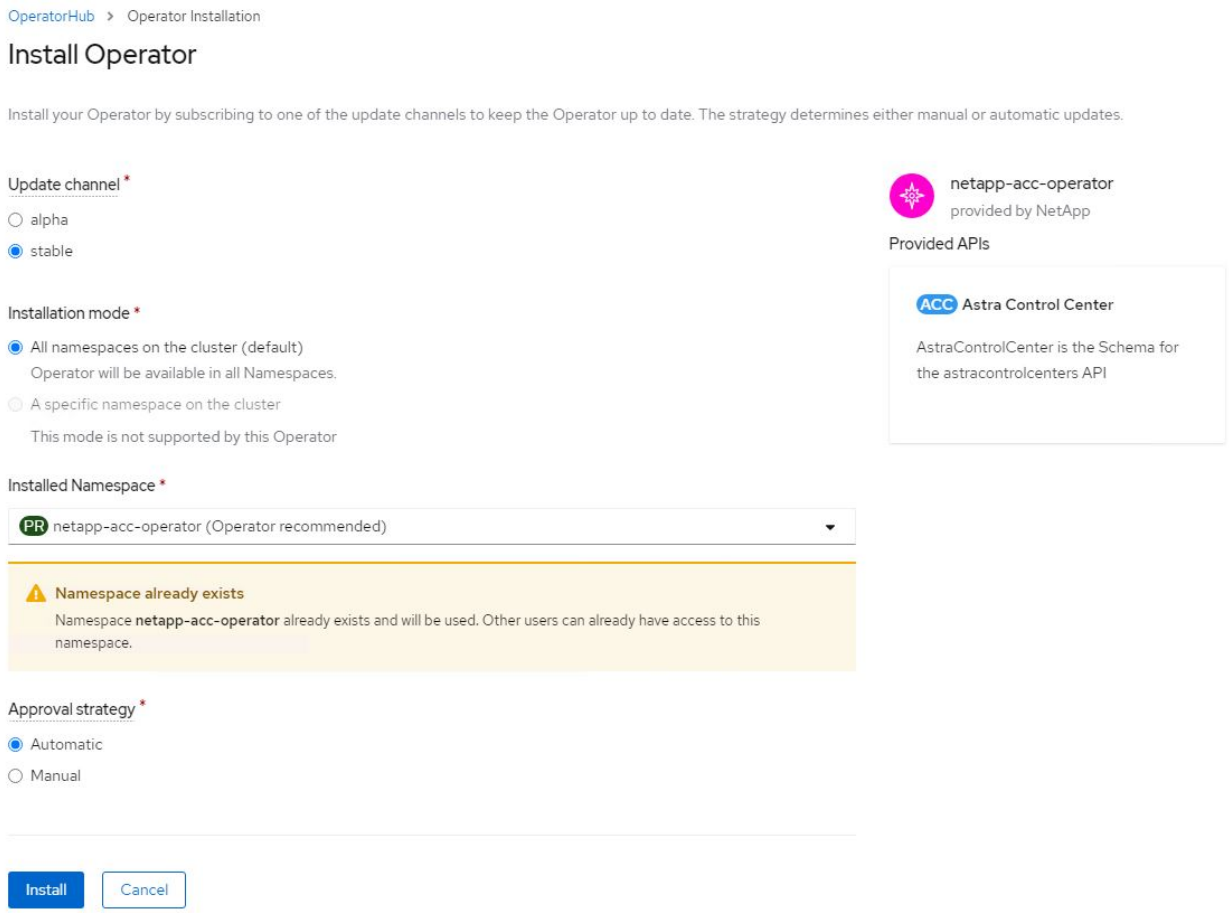
7. Connectez-vous à la console IUG de Red Hat OpenShift avec un accès cluster-admin.
8. Sélectionnez Administrateur dans la liste déroulante perspective.
9. Accédez à Operators > OperatorHub et recherchez Astra.



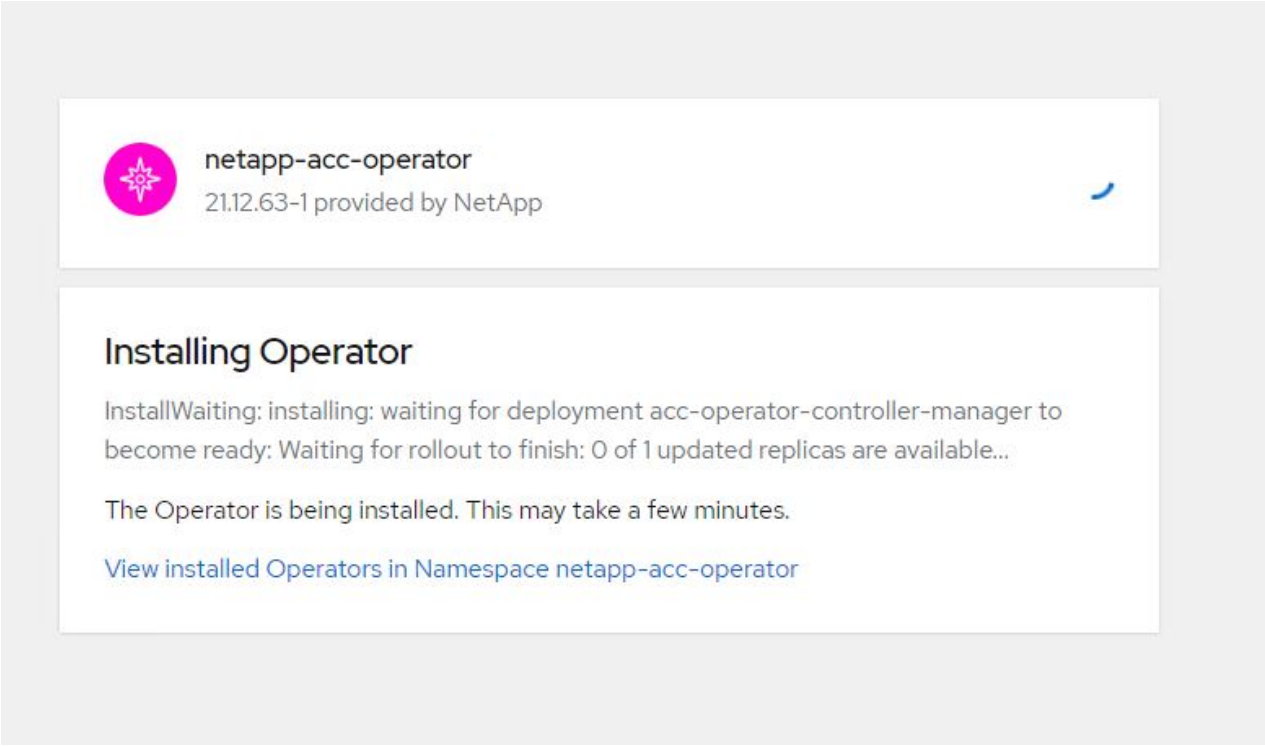
10. Sélectionnez `netapp-acc-operator` mosaïque et clic `Install`.



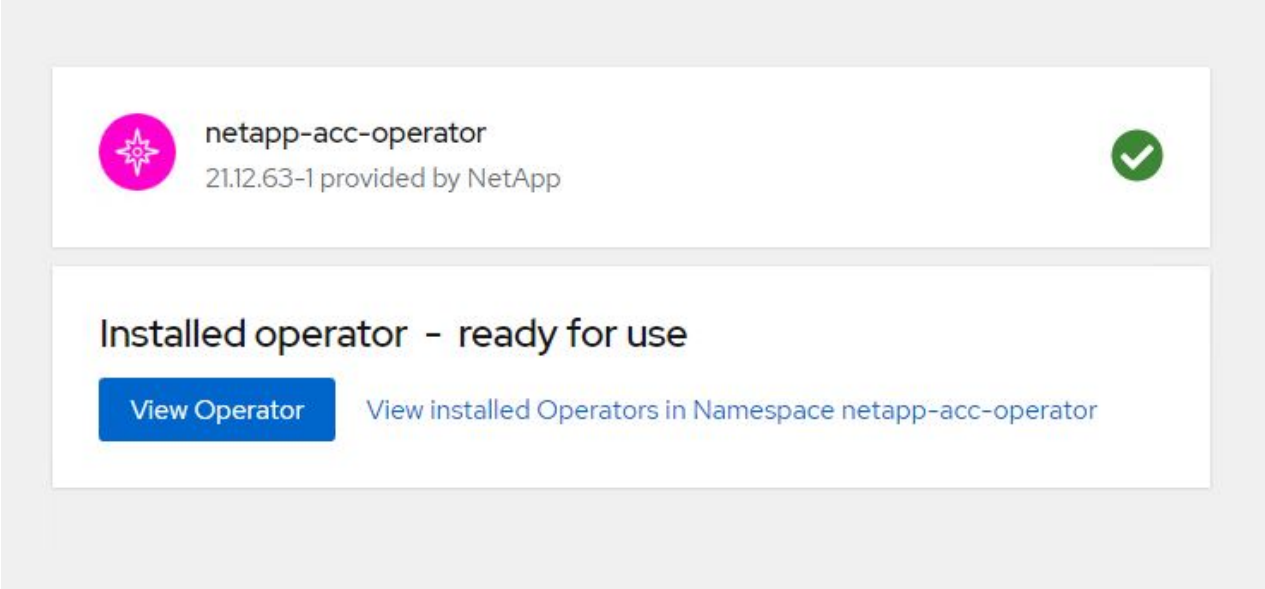
11. Sur l'écran installer l'opérateur, acceptez tous les paramètres par défaut et cliquez sur `Install`.



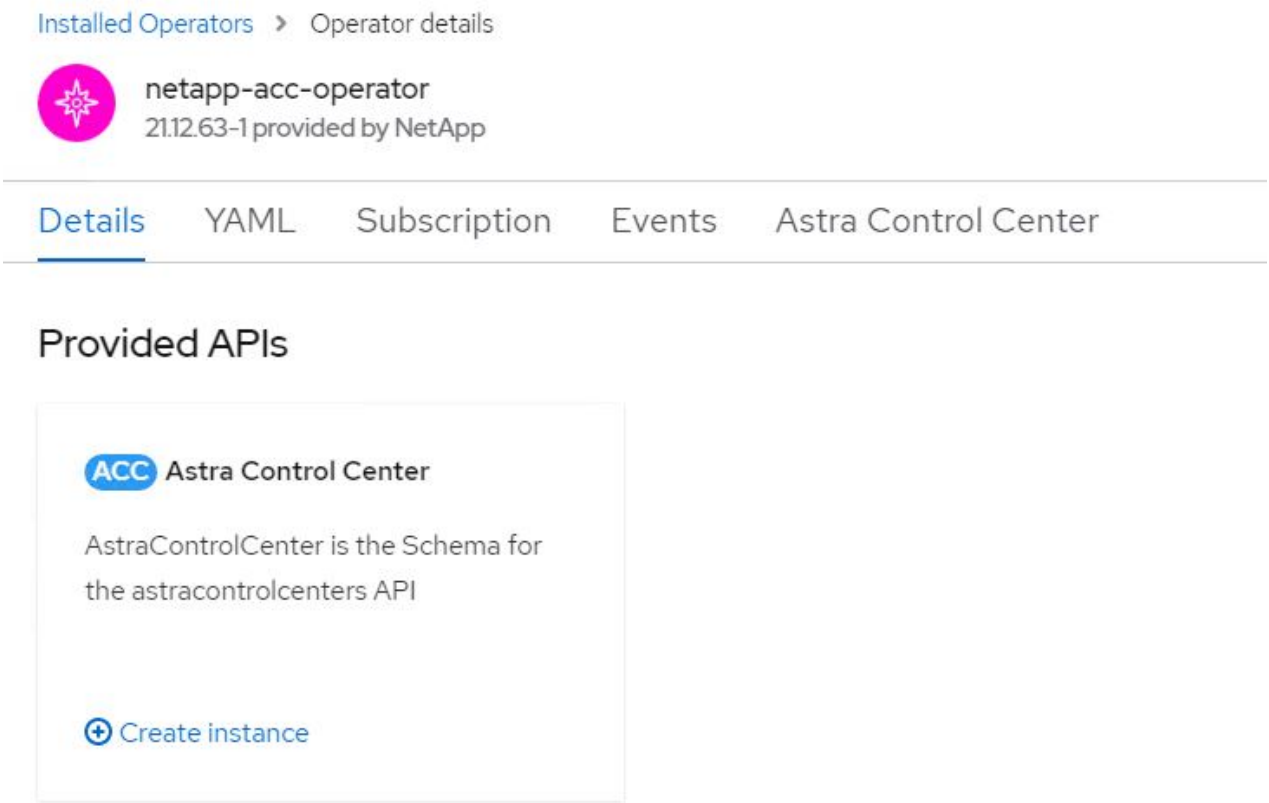
12. Attendre la fin de l'installation par l'opérateur.



13. Une fois l'installation de l'opérateur réussie, cliquez sur `View Operator`.



14. Cliquez ensuite sur `Create Instance` Dans la mosaïque Astra Control Center du conducteur.



15. Remplissez le `Create AstraControlCenter` et cliquez sur `Create`.

    a. Vous pouvez modifier le nom de l'instance du Centre de contrôle Astra.
    b. Vous pouvez éventuellement activer ou désactiver Auto support. Il est recommandé de conserver la fonctionnalité Auto support.
    c. Saisissez le nom de domaine complet pour Astra Control Center.
    d. Accédez à la version du Centre de contrôle Astra ; la dernière est affichée par défaut.

e. Entrez un nom de compte pour le centre de contrôle Astra et des détails d'administrateur tels que le prénom, le nom et l'adresse e-mail.

f. Entrez la règle de récupération du volume. La valeur par défaut est conservation.

g. Dans le Registre d'images, entrez le FQDN de votre registre ainsi que le nom d'organisation tel qu'il a été donné lors de l'envoi des images au Registre (dans cet exemple, `astra-registry.apps.ocp-vmw.cie.netapp.com/netapp-astra`)

h. Si vous utilisez un registre qui nécessite une authentification, entrez le nom secret dans la section Registre d'images.

i. Configurez les options d'échelle pour les limites de ressources Astra Control Center.

j. Entrez le nom de la classe de stockage si vous souhaitez placer des ESV sur une classe de stockage non-défaut.

k. Définissez les préférences de gestion de CRD.

Project: netapp-acc-operator

Name *

astra

Labels

app=frontend

Account Name *

HCG Solutions Engineering

Astra Control Center account name

Astra Address *

astra-control-center.cie.netapp.com

AstraAddress defines how Astra will be found in the data center. This IP address and/or DNS A record must be created prior to provisioning Astra Control Center. Example - "astra.example.com" The A record and its IP address must be allocated prior to provisioning Astra Control Center

Astra Version *

21.12.60

Version of AstraControlCenter to deploy. You are provided a Helm repository with a corresponding version. Example - 1.5.2, 1.4.2-patch

Email *

solutions_tme@netapp.com

EmailAddress will be notified by Astra as events warrant.

Auto Support *

AutoSupport indicates willingness to participate in NetApp's proactive support application, NetApp Active IQ. The default election is true and indicates support data will be sent to NetApp. An empty or blank election is the same as a default election. Air gapped installations should enter false.

First Name

HCG

The first name of the SRE supporting Astra.

Last Name

Admin

The last name of the SRE supporting Astra.

Image Registry

The container image registry that is hosting the Astra application images, ACC Operator and ACC Helm Repository.

Name

astra-registry.apps.ocp-vmw.cie.netapp.com/netapp-astra

The name of the image registry. For example "example.registry/astra". Do not prefix with protocol.

Secret

astra-registry-cred

The name of the Kubernetes secret that will authenticate with the image registry.

Volume Reclaim Policy

Retain

Reclaim policy to be set for persistent volumes

Astra Resources Scaler

Default

Scaling options for AstraControlCenter Resource limits.

Storage Class

The storage class to be used for PVCs. If not set, default storage class will be used.

Crds

Options for how ACC should handle CRDs.

Create Cancel

Automatisation [Ansible]

1. Pour déployer Astra Control Center sur un playbooks Ansible, vous devez utiliser un ordinateur Ubuntu/RHEL avec Ansible installé. Suivre les procédures "ici" Pour Ubuntu et RHEL.
2. Clonez le référentiel GitHub qui héberge le contenu Ansible.

```
git clone https://github.com/NetApp-
Automation/na_astra_control_suite.git
```

3. Connectez-vous au site de support NetApp et téléchargez la dernière version de NetApp Astra Control Center. Une licence associée à votre compte NetApp est requise. Après avoir téléchargé le tarball, transférez-le sur le poste de travail.

> Pour commencer avec une licence d'essai d'Astra Control, visitez le "Site d'inscription à Astra".

4. Créez ou obtenez le fichier kubeconfig avec un accès administrateur au cluster OpenShift sur lequel vous devez installer Astra Control Center.

5. Remplacez le répertoire par na_astra_control_suite.

```
cd na_astra_control_suite
```

6. Modifiez le `vars/vars.yml` et remplissez les variables avec les informations requises.

```
#Define whether or not to push the Astra Control Center images to
your private registry [Allowed values: yes, no]
push_images: yes

#The directory hosting the Astra Control Center installer
installer_directory: /home/admin/

#Specify the ingress type. Allowed values - "AccTraefik" or
"Generic"
#"AccTraefik" if you want the installer to create a LoadBalancer
type service to access ACC, requires MetalLB or similar.
#"Generic" if you want to create or configure ingress controller
yourself, installer just creates a ClusterIP service for traefik.
ingress_type: "AccTraefik"

#Name of the Astra Control Center installer (Do not include the
extension, just the name)
astra_tar_ball_name: astra-control-center-22.04.0

#The complete path to the kubeconfig file of the
kubernetes/openshift cluster Astra Control Center needs to be
installed to.
hosting_k8s_cluster_kubeconfig_path: /home/admin/cluster-
kubeconfig.yml

#Namespace in which Astra Control Center is to be installed
astra_namespace: netapp-astra-cc

#Astra Control Center Resources Scaler. Leave it blank if you want
to accept the Default setting.
astra_resources_scaler: Default

#Storageclass to be used for Astra Control Center PVCs, it must be
created before running the playbook [Leave it blank if you want the
PVCs to use default storageclass]
astra_trident_storageclass: basic

#Reclaim Policy for Astra Control Center Persistent Volumes [Allowed
values: Retain, Delete]
```

```
storageclass_reclaim_policy: Retain

#Private Registry Details
astra_registry_name: "docker.io"

#Whether the private registry requires credentials [Allowed values:
yes, no]
require_reg_creds: yes

#If require_reg_creds is yes, then define the container image
registry credentials
#Usually, the registry namespace and usernames are same for
individual users
astra_registry_namespace: "registry-user"
astra_registry_username: "registry-user"
astra_registry_password: "password"

#Kuberenets/OpenShift secret name for Astra Control Center
#This name will be assigned to the K8s secret created by the
playbook
astra_registry_secret_name: "astra-registry-credentials"

#Astra Control Center FQDN
acc_fqdn_address: astra-control-center.cie.netapp.com

#Name of the Astra Control Center instance
acc_account_name: ACC Account Name

#Administrator details for Astra Control Center
admin_email_address: admin@example.com
admin_first_name: Admin
admin_last_name: Admin
```

7. Utilisez le PlayBook pour déployer le centre de contrôle Astra. Le PlayBook requiert des privilèges root pour certaines configurations.

   Si l'utilisateur exécutant le PlayBook est root ou a configuré un sudo sans mot de passe, exécutez la commande suivante pour exécuter le PlayBook.

   ```
   ansible-playbook install_acc_playbook.yml
   ```

   Si l'accès sudo basé sur un mot de passe est configuré, exécutez la commande suivante pour exécuter le PlayBook, puis saisissez le mot de passe sudo.

   ```
   ansible-playbook install_acc_playbook.yml -K
   ```

Après l'installation

1. L'installation peut prendre plusieurs minutes. Vérifier que tous les pods et services dans le `netapp-astra-cc` les espaces de noms sont opérationnels.

```
[netapp-user@rhel7 ~]$ oc get all -n netapp-astra-cc
```

2. Vérifier le `acc-operator-controller-manager` journaux pour vérifier que l'installation est terminée.

```
[netapp-user@rhel7 ~]$ oc logs deploy/acc-operator-controller-manager -n
netapp-acc-operator -c manager -f
```

Le message suivant indique que le centre de contrôle Astra a été installé avec succès.

```
{"level":"info","ts":1624054318.029971,"logger":"controllers.AstraContro
lCenter","msg":"Successfully Reconciled AstraControlCenter in
[seconds]s","AstraControlCenter":"netapp-astra-
cc/astra","ae.Version":"[21.12.60]"}
```

3. Le nom d'utilisateur pour la connexion à Astra Control Center est l'adresse électronique de l'administrateur fournie dans le fichier CRD et le mot de passe est une chaîne `ACC-` Joint à l'UUID du centre de contrôle Astra. Exécutez la commande suivante :

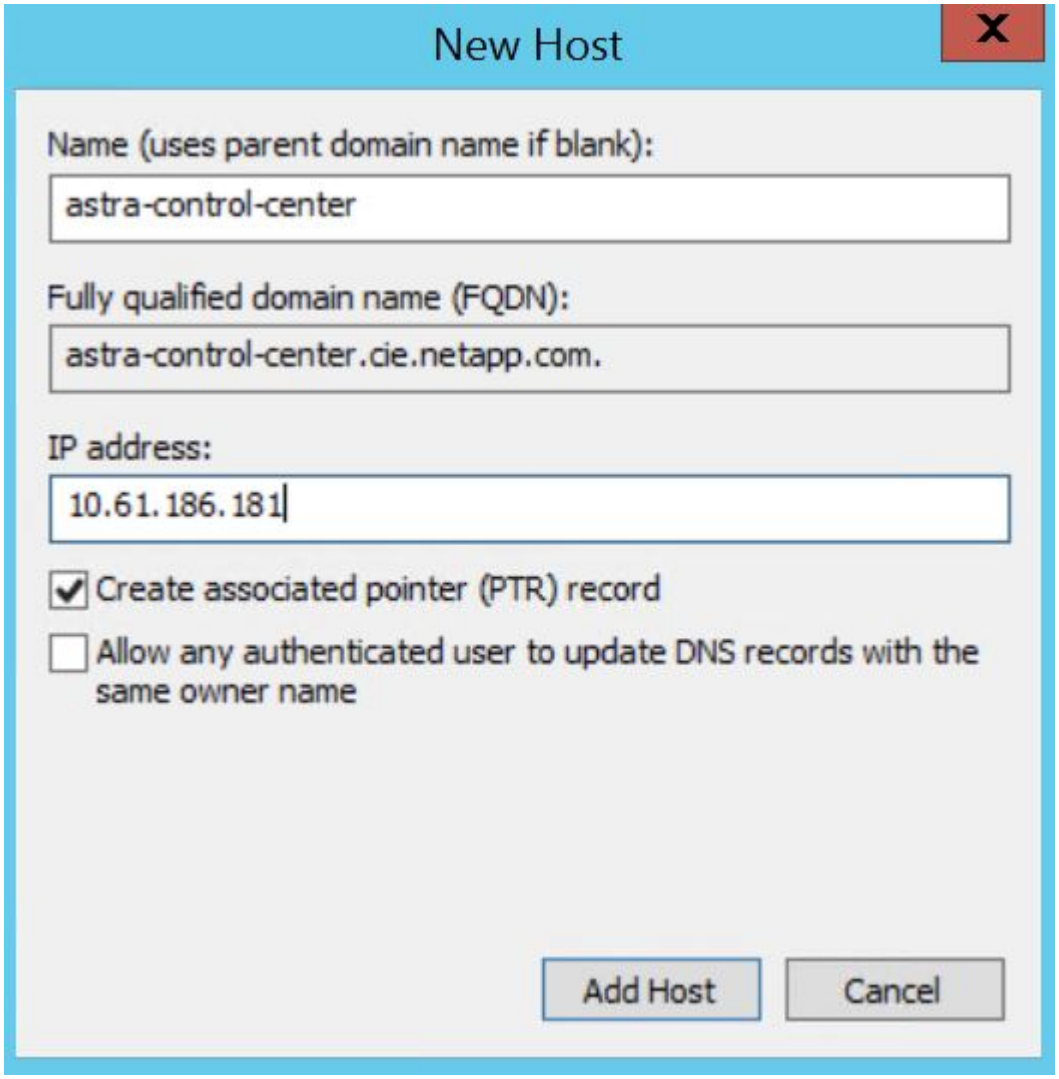
```
[netapp-user@rhel7 ~]$ oc get astracontrolcenters -n netapp-astra-cc
NAME    UUID
astra   345c55a5-bf2e-21f0-84b8-b6f2bce5e95f
```

Dans cet exemple, le mot de passe est `ACC-345c55a5-bf2e-21f0-84b8-b6f2bce5e95f`.

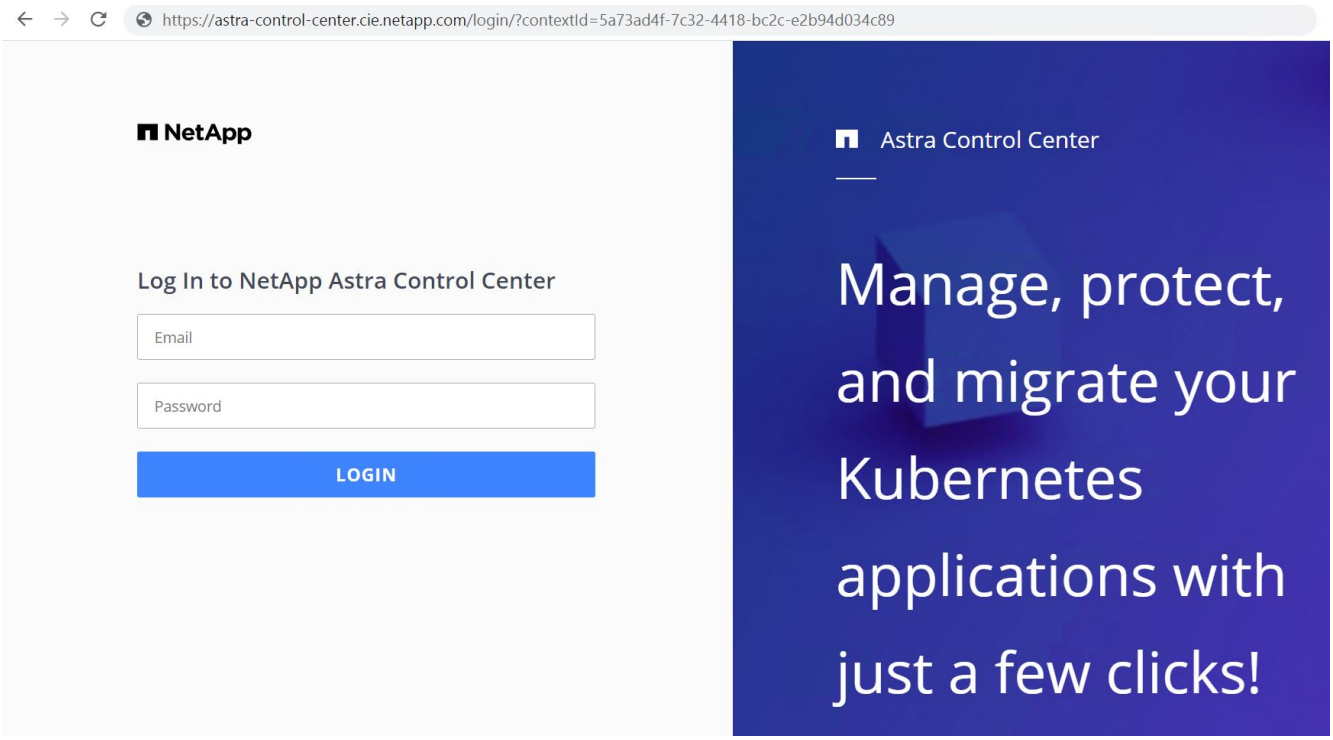
4. Procurez-vous l'IP d'équilibrage de charge du service traefik.

```
[netapp-user@rhel7 ~]$ oc get svc -n netapp-astra-cc | egrep
'EXTERNAL|traefik'

NAME                                        TYPE           CLUSTER-IP
EXTERNAL-IP     PORT(S)
AGE
traefik                                     LoadBalancer   172.30.99.142
10.61.186.181   80:30343/TCP,443:30060/TCP
16m
```

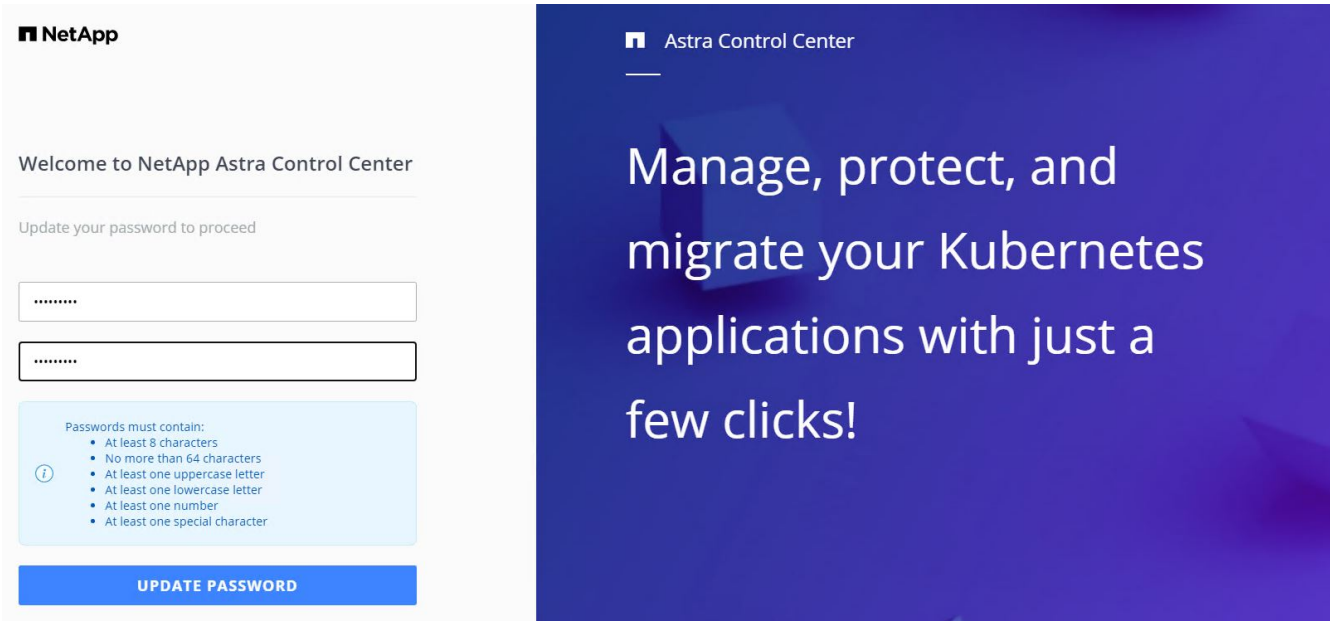
5. Ajoutez une entrée dans le serveur DNS pointant le FQDN fourni dans le fichier CRD Astra Control Center vers le `EXTERNAL-IP` du service de trafik.



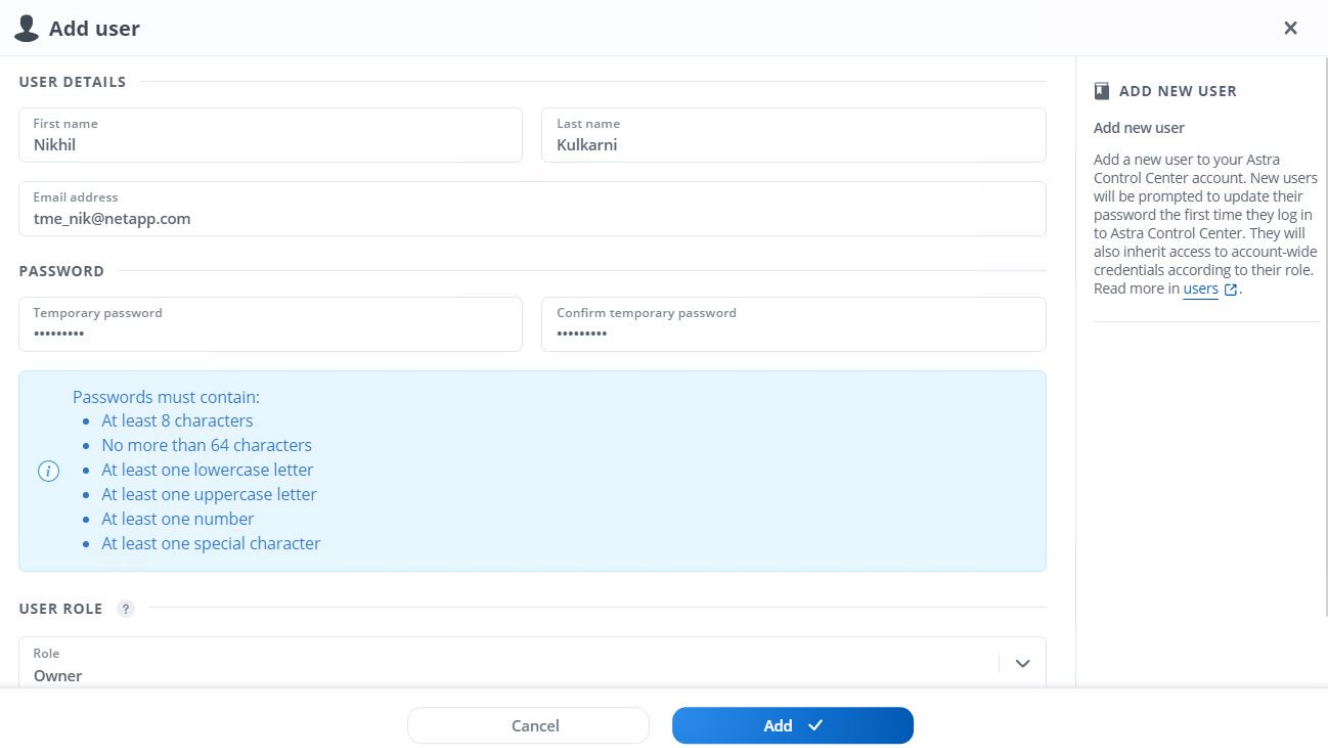
6. Connectez-vous à l'interface graphique d'Astra Control Center en parcourant son FQDN.



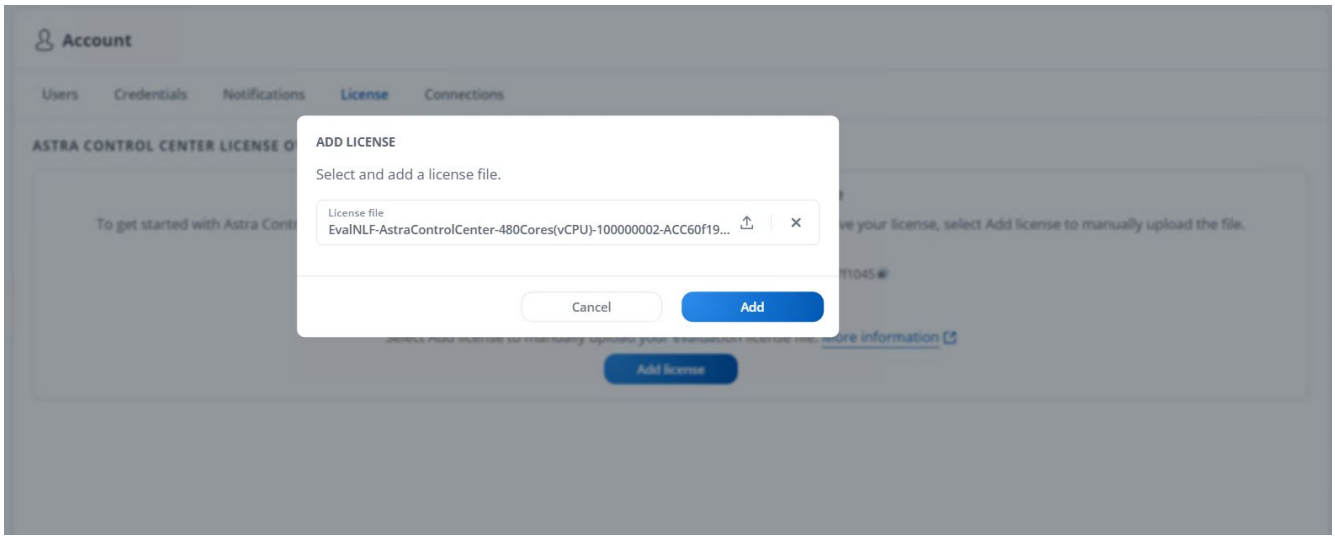
7. Lorsque vous vous connectez à l'interface graphique d'Astra Control Center pour la première fois à l'aide de l'adresse e-mail d'administration fournie dans CRD, vous devez modifier le mot de passe.



8. Si vous souhaitez ajouter un utilisateur au Centre de contrôle Astra, accédez à compte > utilisateurs, cliquez sur Ajouter, entrez les détails de l'utilisateur et cliquez sur Ajouter.



9. Astra Control Center requiert une licence pour toutes ses fonctionnalités. Pour ajouter une licence, accédez à compte > Licence, cliquez sur Ajouter une licence et téléchargez le fichier de licence.



> En cas de problème avec l'installation ou la configuration de NetApp Astra Control Center, la base de connaissances des problèmes connus est disponible "ici".

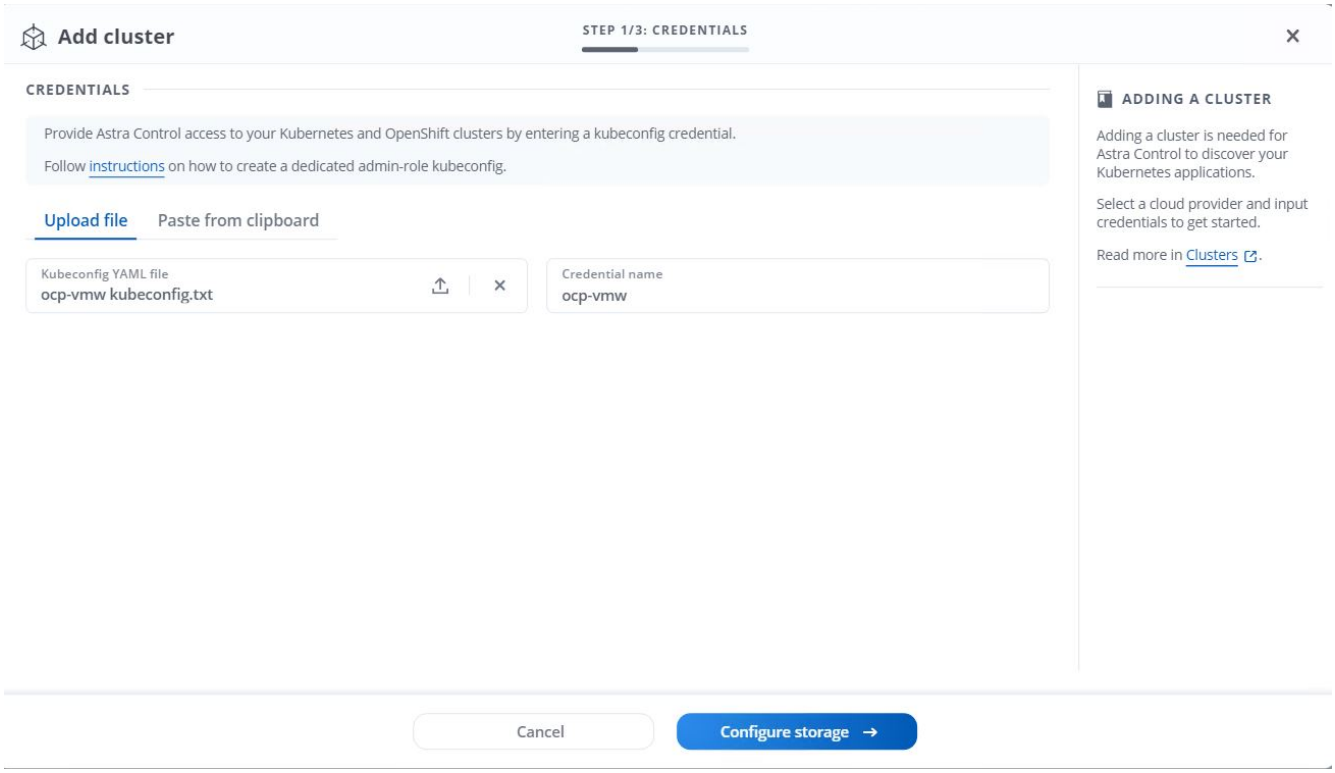
Enregistrez vos clusters Red Hat OpenShift avec Astra Control Center

Pour permettre à Astra Control Center de gérer vos charges de travail, vous devez d'abord enregistrer votre cluster Red Hat OpenShift.

Enregistrez les clusters Red Hat OpenShift

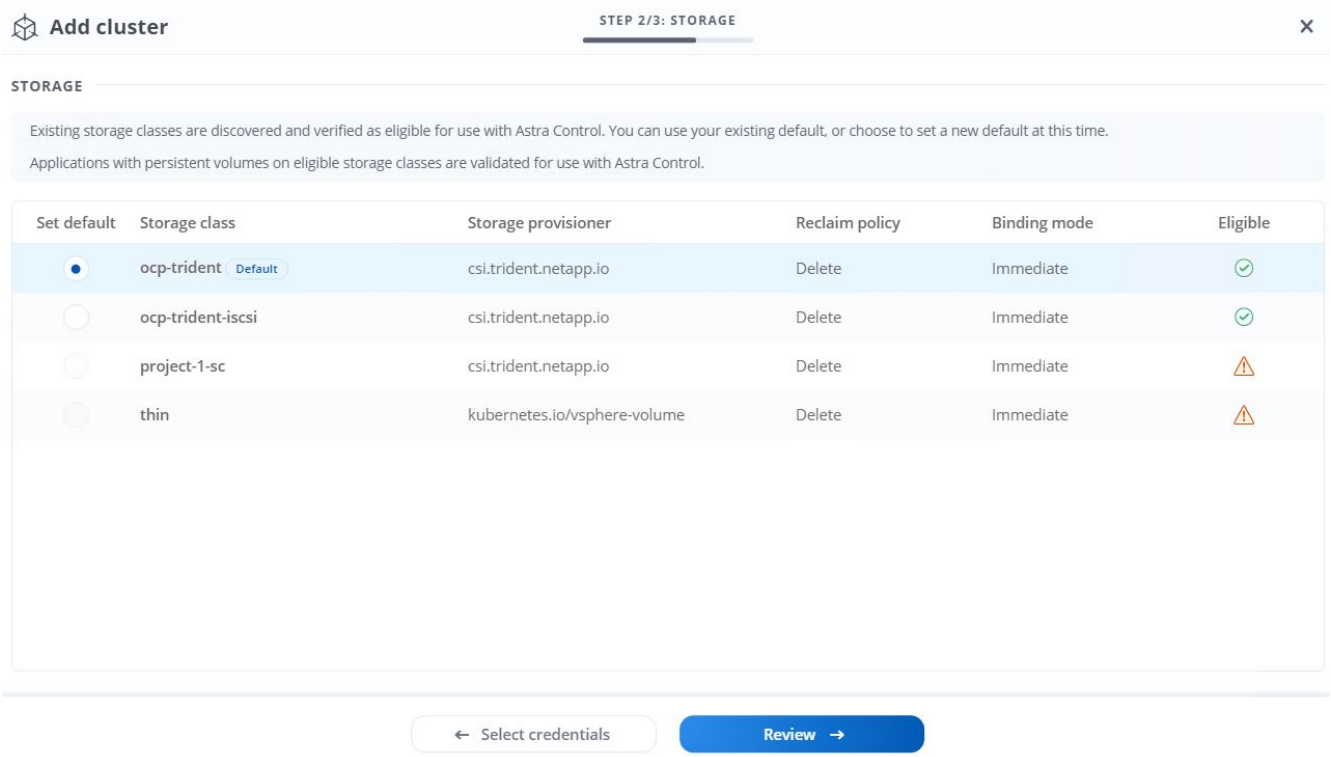
1. La première étape consiste à ajouter les clusters OpenShift au Centre de contrôle Astra et à les gérer.

Accédez aux clusters et cliquez sur Ajouter un cluster, téléchargez le fichier kubeconfig pour le cluster OpenShift, puis cliquez sur Sélectionner un stockage.

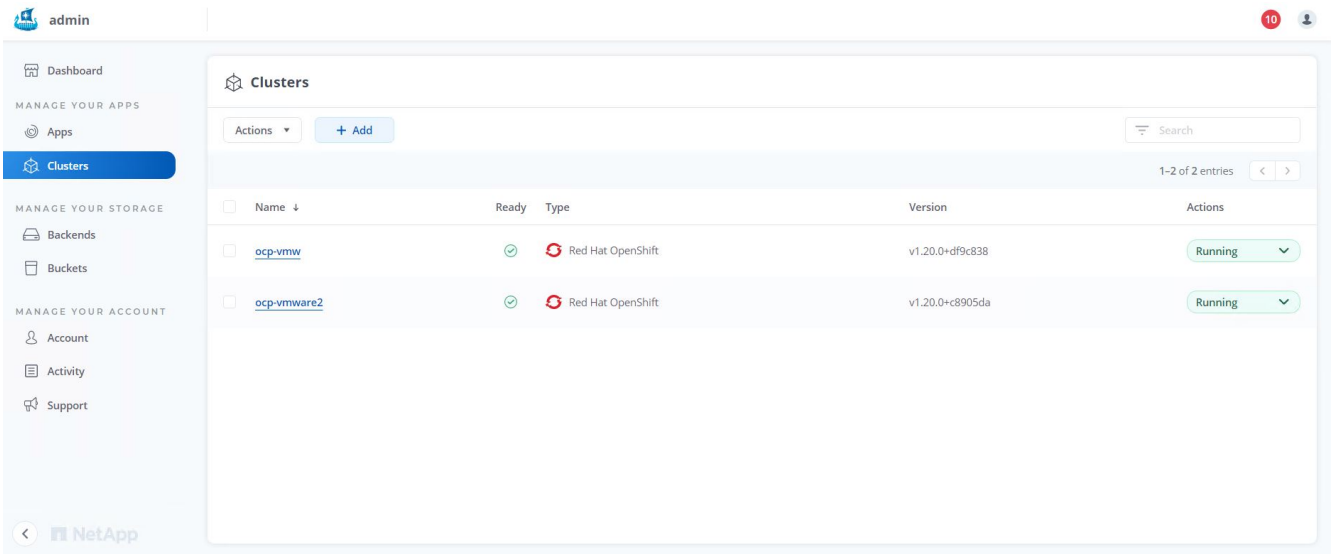


> Le fichier kubeconfig peut être généré pour s'authentifier avec un nom d'utilisateur et un mot de passe ou un jeton. Les jetons expirent après un délai limité et peuvent laisser le cluster enregistré inaccessible. NetApp recommande d'utiliser un fichier kubeconfig avec un nom d'utilisateur et un mot de passe pour enregistrer vos clusters OpenShift sur Astra Control Center.

2. Astra Control Center détecte les classes de stockage admissibles. Maintenant, sélectionnez la façon dont storageclass provisionne les volumes en utilisant Trident sauvegardé par un SVM sur NetApp ONTAP et Click Review. Dans le volet suivant, vérifiez les détails et cliquez sur Ajouter un cluster.

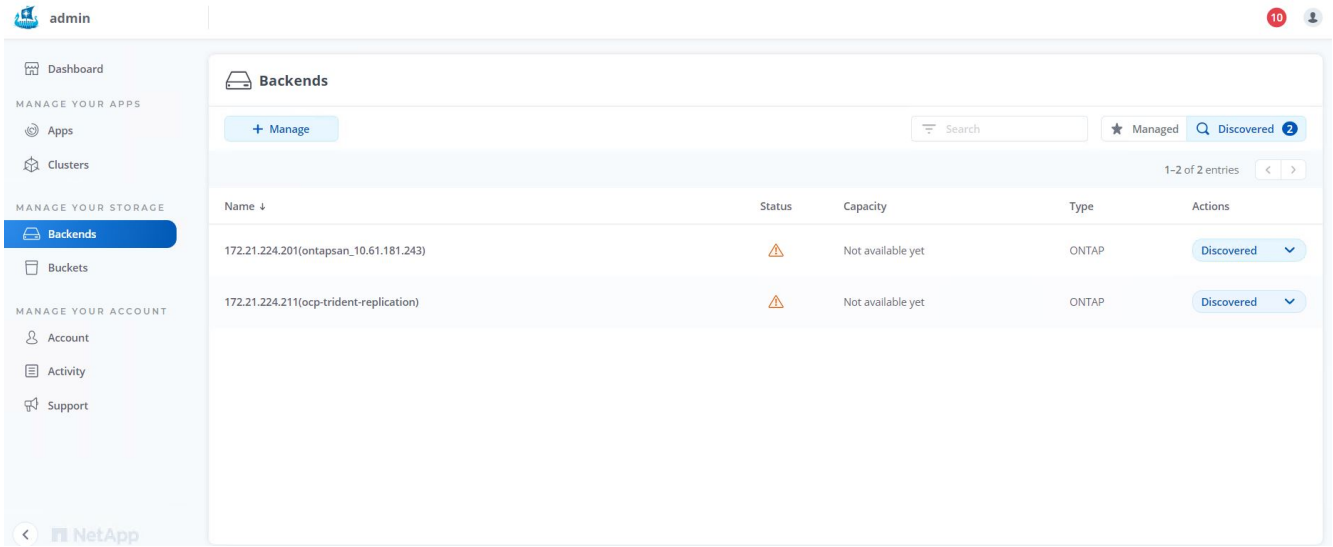


3. Enregistrez les deux clusters OpenShift comme décrit à l'étape 1. Lorsqu'elles sont ajoutées, les clusters passent à l'état découverte pendant qu'Astra Control Center les inspecte et installe les agents nécessaires. L'état du cluster est modifié en cours d'exécution après son enregistrement.

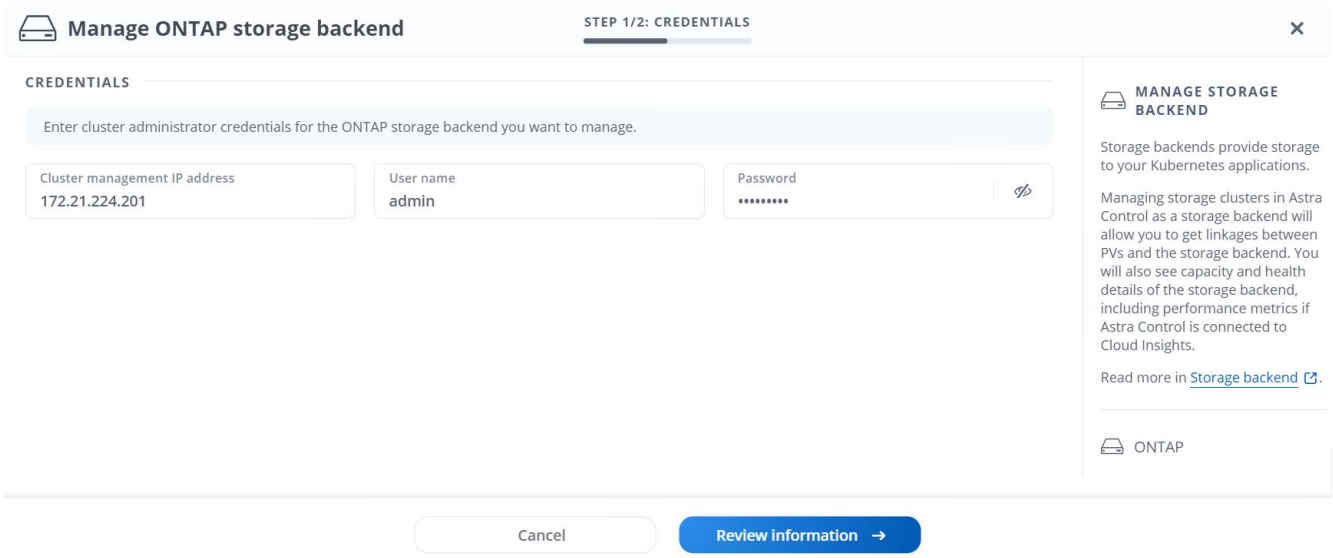


> Tous les clusters Red Hat OpenShift devant être gérés par Astra Control Center doivent avoir accès au registre d'images utilisé pour son installation lorsque les agents installés sur les clusters gérés extraient les images de ce registre.

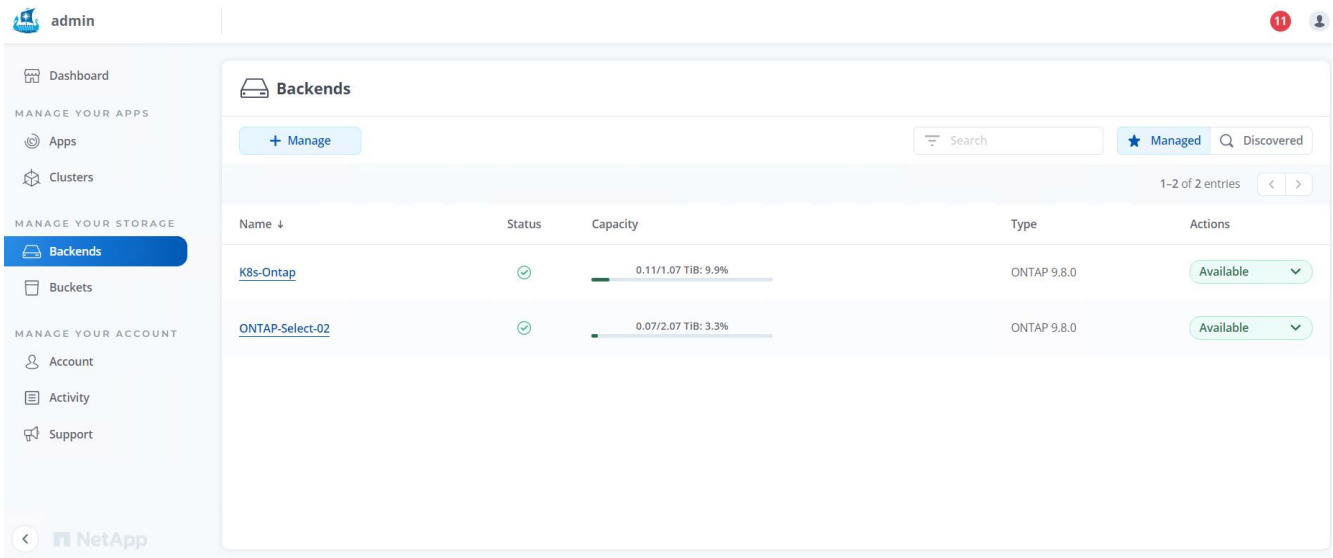
4. Importation de clusters ONTAP comme ressources de stockage à gérer en tant que système back-end par Astra Control Center. Lorsque des clusters OpenShift sont ajoutés à Astra et qu'un storageclass est configuré, il détecte et inspecte automatiquement le cluster ONTAP qui soutient le storageclass, mais ne l'importe pas dans le Control Center Astra à gérer.



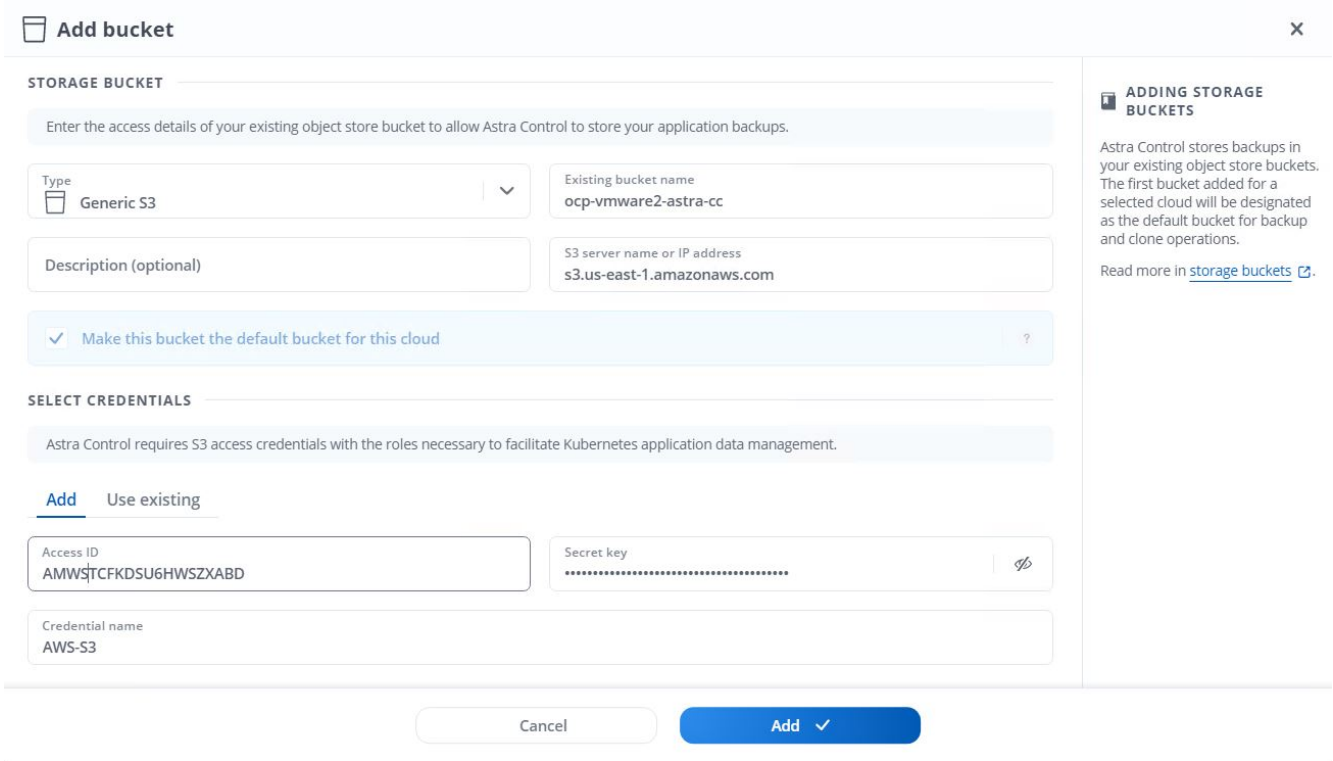
5. Pour importer les clusters ONTAP, accédez aux systèmes back-end, cliquez sur la liste déroulante et sélectionnez gérer en regard du cluster ONTAP à gérer. Entrez les informations d'identification du cluster ONTAP, cliquez sur vérifier les informations, puis sur Importer le stockage back-end.



6. Une fois que le système back-end est ajouté, le statut devient disponible. Ces systèmes back-end disposent désormais d'informations sur les volumes persistants dans le cluster OpenShift et sur les volumes correspondants sur le système ONTAP.



7. Pour la sauvegarde et la restauration entre les clusters OpenShift avec Astra Control Center, vous devez provisionner un compartiment de stockage objet qui prend en charge le protocole S3. Les options ONTAP S3, StorageGRID et AWS S3 sont actuellement prises en charge. Pour les besoins de cette installation, nous allons configurer un compartiment AWS S3. Accédez à godets, cliquez sur Ajouter un compartiment et sélectionnez Generic S3. Entrez les informations d'identification du compartiment S3 et des informations d'identification pour y accéder, cochez la case « définir ce compartiment par défaut pour le cloud », puis cliquez sur Ajouter.

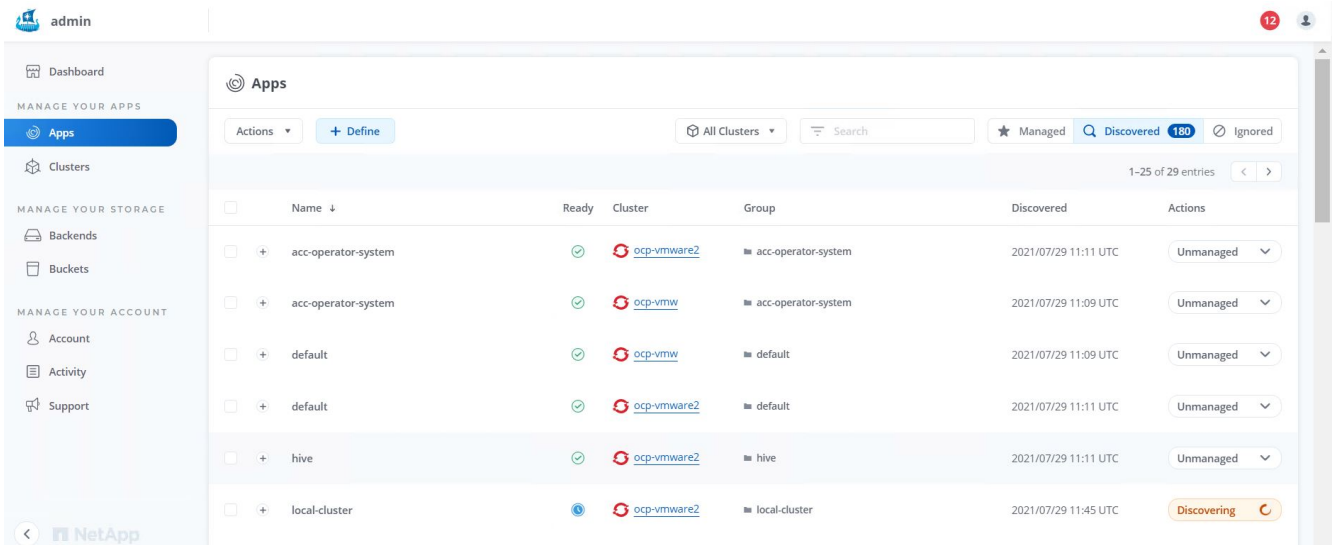


Choisissez les applications à protéger

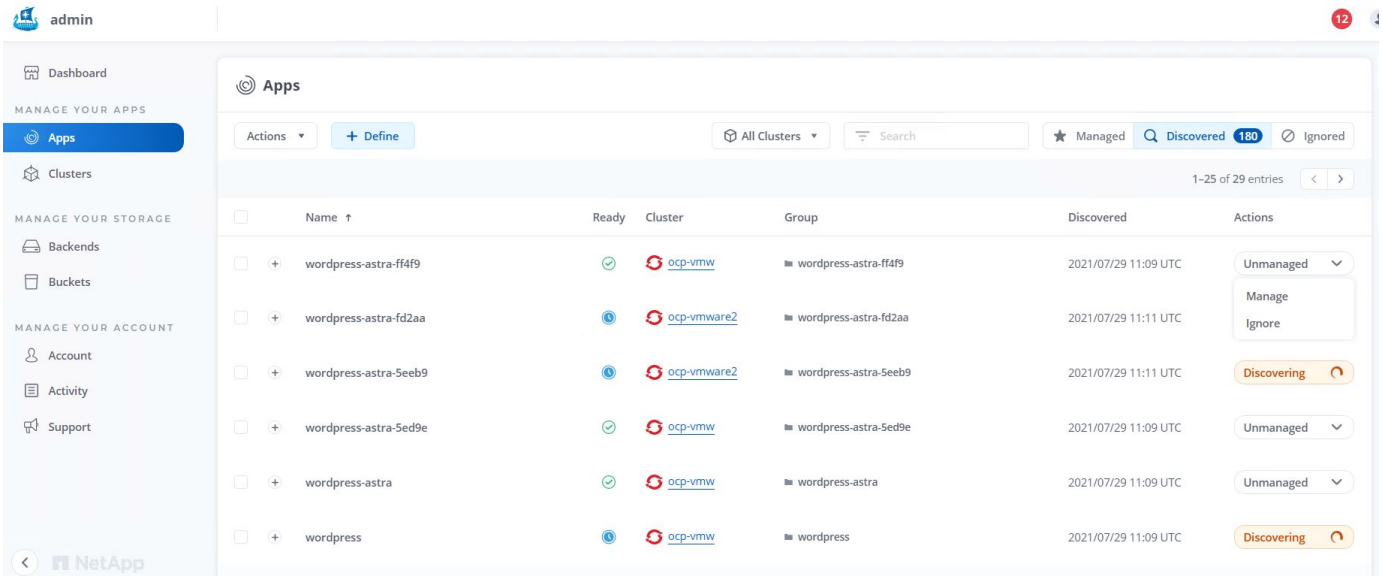
Une fois vos clusters Red Hat OpenShift enregistrés, vous pouvez détecter les applications déployées et les gérer via Astra Control Center.

Gestion des applications

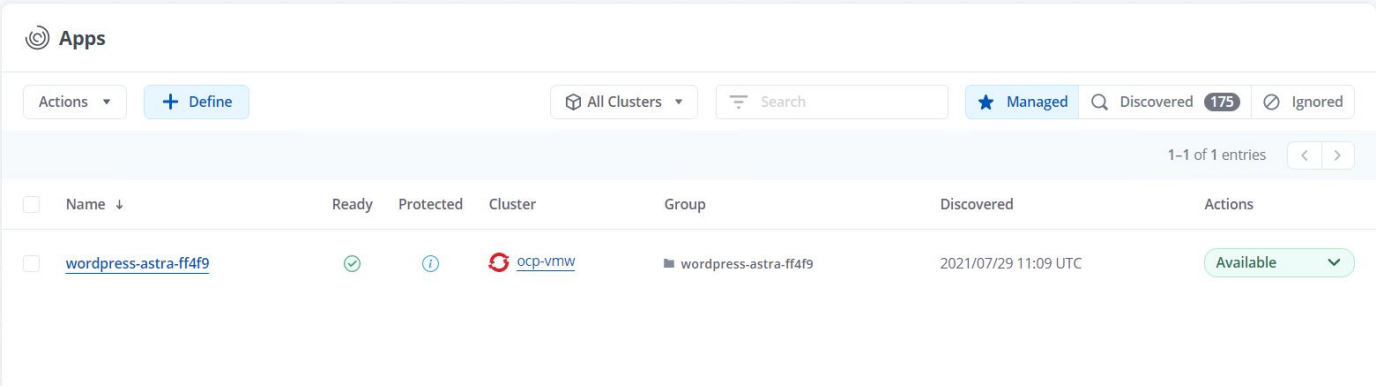
1. Une fois les clusters OpenShift et les systèmes back-end ONTAP enregistrés auprès de l'Astra Control Center, le centre de contrôle démarre automatiquement la détection des applications dans tous les namespaces qui utilisent le storageclass configuré avec le back-end ONTAP spécifié.



2. Accédez à applications > découverte et cliquez sur le menu déroulant en regard de l'application que vous souhaitez gérer à l'aide d'Astra. Cliquez ensuite sur gérer.



1. L'application passe à l'état disponible et peut être affichée sous l'onglet géré de la section applications.



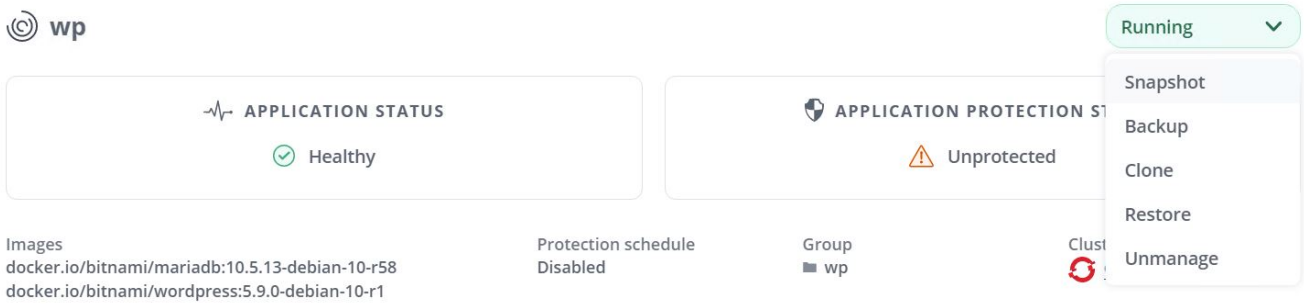
Protégez vos applications

Une fois les charges de travail applicatives gérées par Astra Control Center, vous pouvez configurer les paramètres de protection pour ces charges de travail.

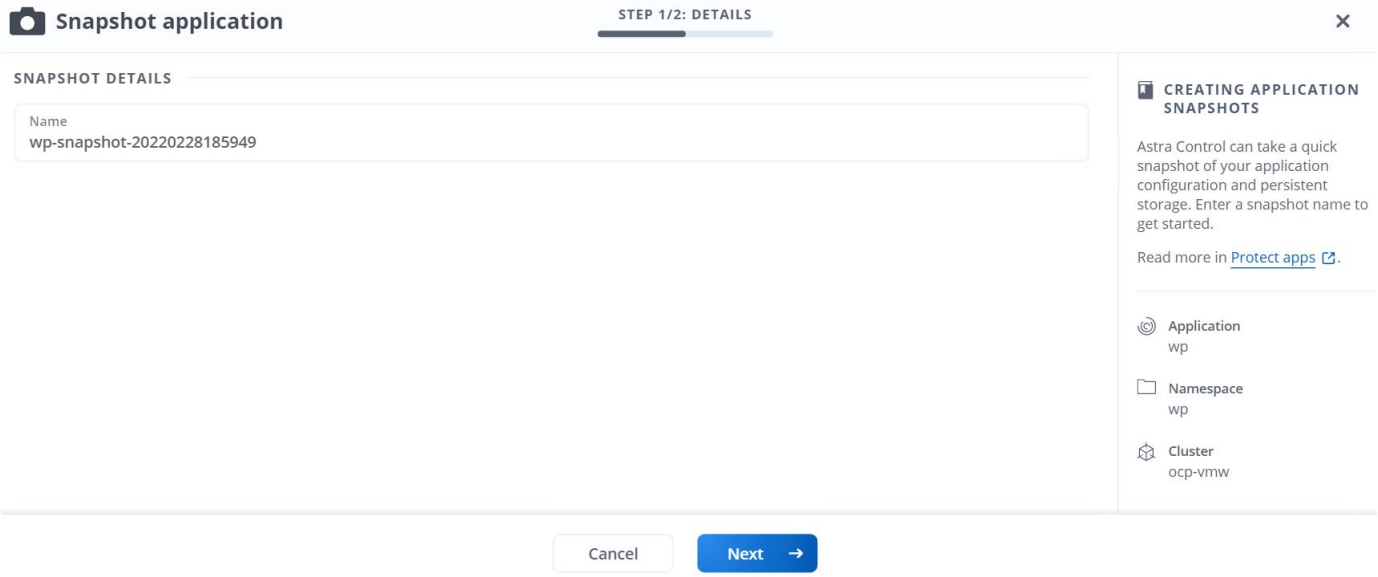
Création d'un instantané d'application

Un snapshot d'une application crée une copie Snapshot ONTAP qui peut être utilisée pour restaurer ou cloner l'application à un point dans le temps spécifique en fonction de cette copie Snapshot.

1. Pour prendre un instantané de l'application, accédez à l'onglet applications > gestion, puis cliquez sur l'application dont vous souhaitez effectuer une copie Snapshot. Cliquez sur le menu déroulant en regard du nom de l'application et cliquez sur instantané.



2. Entrez les détails du snapshot, cliquez sur Suivant, puis sur instantané. La création du Snapshot prend environ une minute et son état est disponible une fois celui-ci créé.



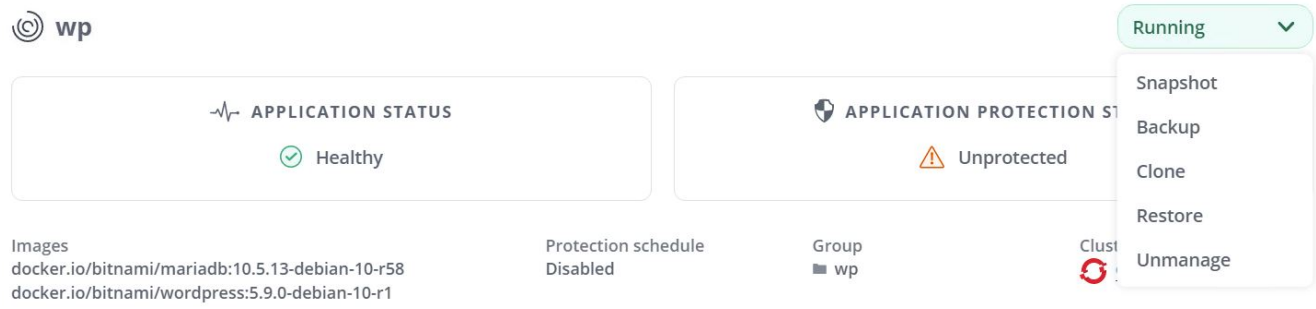
Création d'une sauvegarde d'application

Une sauvegarde d'une application capture l'état actif de l'application et la configuration des ressources informatiques, les analyse en fichiers et les stocke dans un compartiment de stockage objet distant.

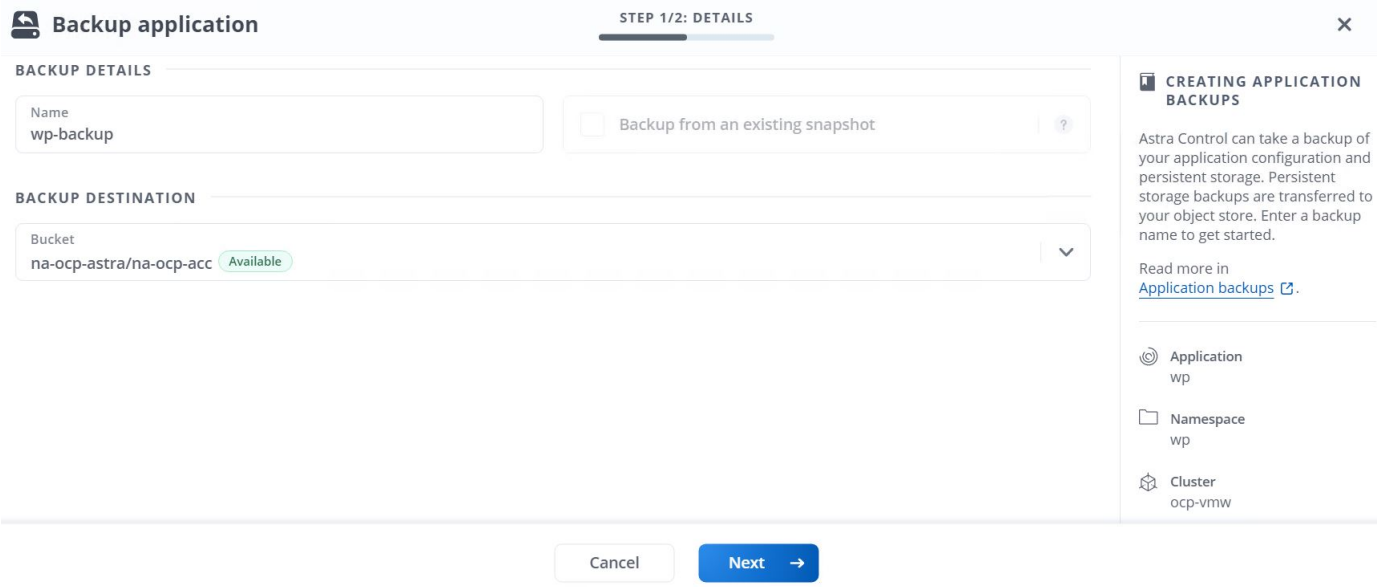
Pour la sauvegarde et la restauration des applications gérées dans le Centre de contrôle Astra, vous devez configurer les paramètres de superutilisateur des systèmes ONTAP de secours au préalable. Pour ce faire, entrez les commandes suivantes.

```
ONTAP::> export-policy rule modify -vserver ocp-trident -policyname
default -ruleindex 1 -superuser sys
ONTAP::> export-policy rule modify -policyname default -ruleindex 1 -anon
65534 -vserver ocp-trident
```

1. Pour créer une sauvegarde de l'application gérée dans Astra Control Center, accédez à l'onglet applications > géré et cliquez sur l'application dont vous souhaitez effectuer une sauvegarde. Cliquez sur le menu déroulant en regard du nom de l'application et cliquez sur Sauvegarder.



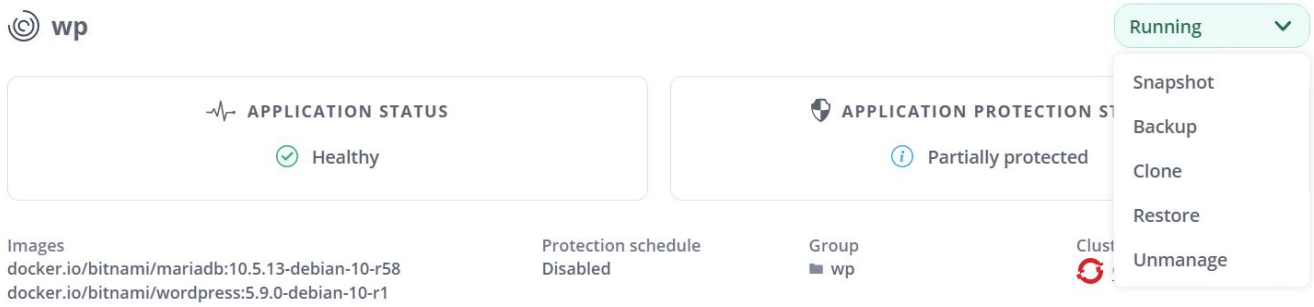
2. Entrez les détails de la sauvegarde, sélectionnez le compartiment de stockage objet pour contenir les fichiers de sauvegarde, cliquez sur Next (Suivant) et, après avoir vérifié les détails, cliquez sur Backup (Sauvegarder). Selon la taille de l'application et des données, la sauvegarde peut prendre plusieurs minutes, et l'état de la sauvegarde est disponible une fois la sauvegarde terminée.



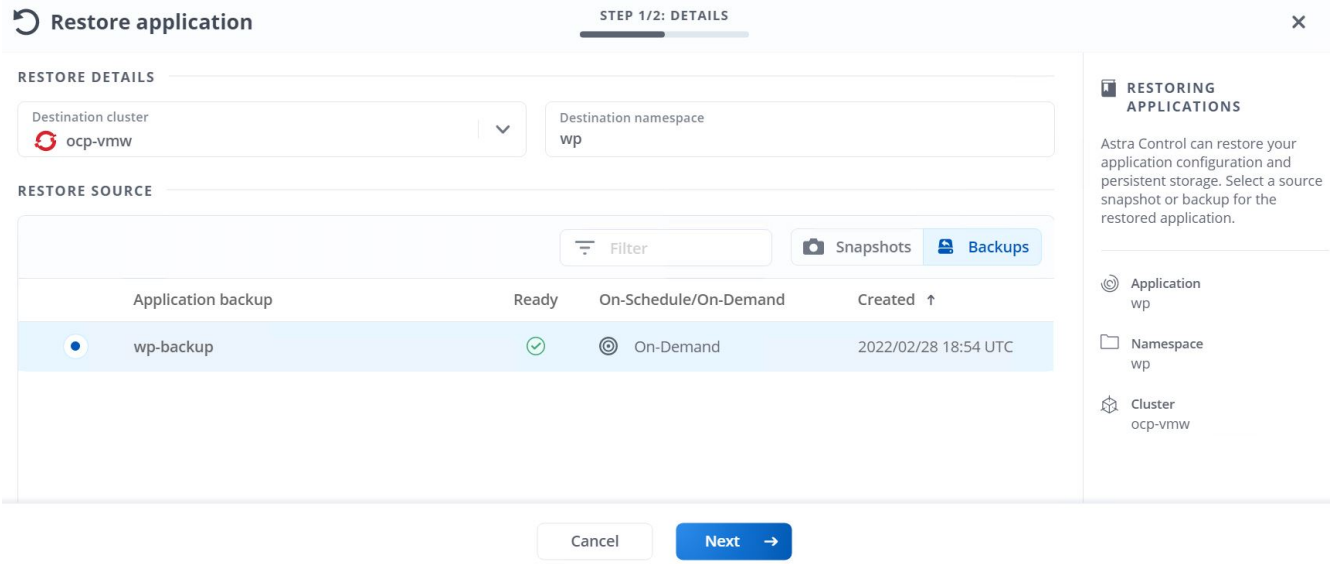
Restauration d'une application

En appuyant sur un bouton, vous pouvez restaurer une application sur l'espace de noms d'origine dans le même cluster ou sur un cluster distant afin d'assurer la protection des applications et la reprise sur incident.

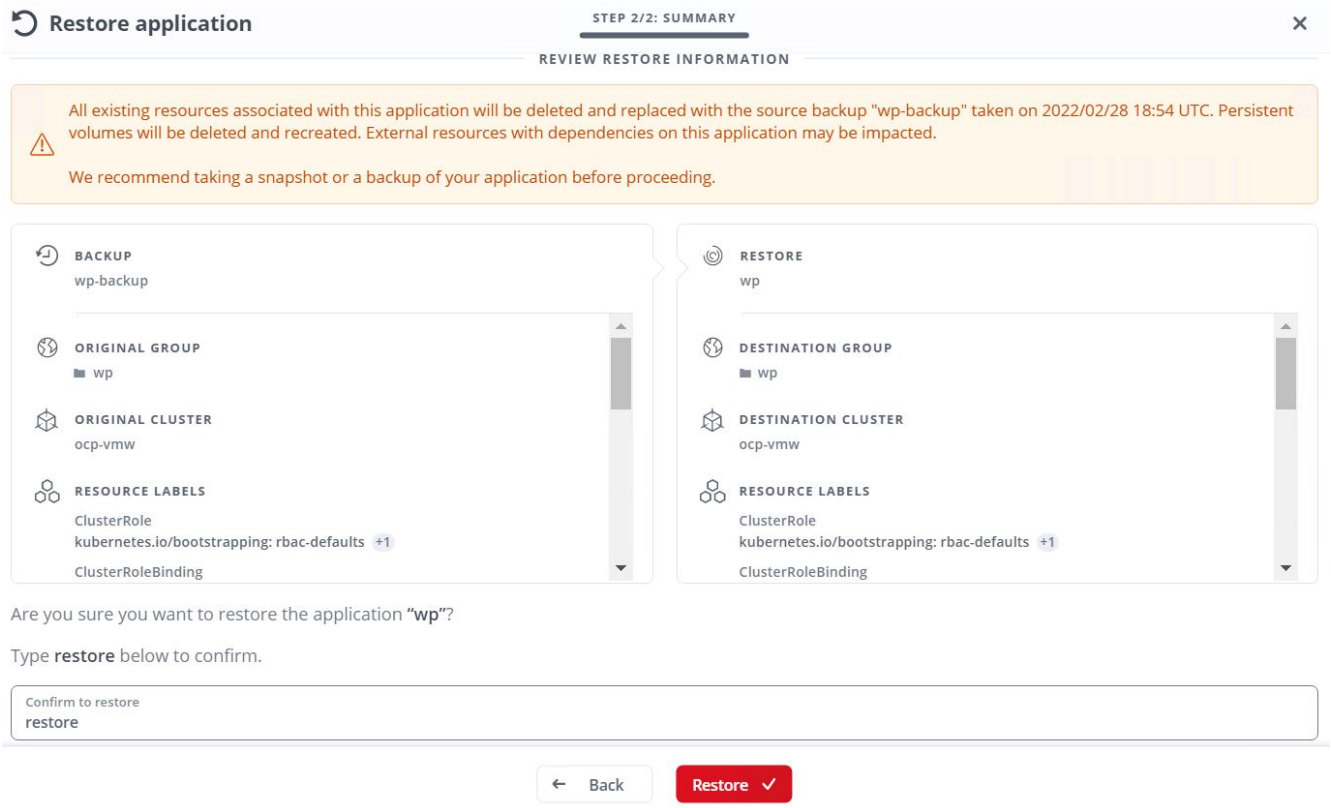
1. Pour restaurer une application, accédez à applications > onglet géré et cliquez sur l'application en question. Cliquez sur le menu déroulant en regard du nom de l'application et cliquez sur `Restore`.



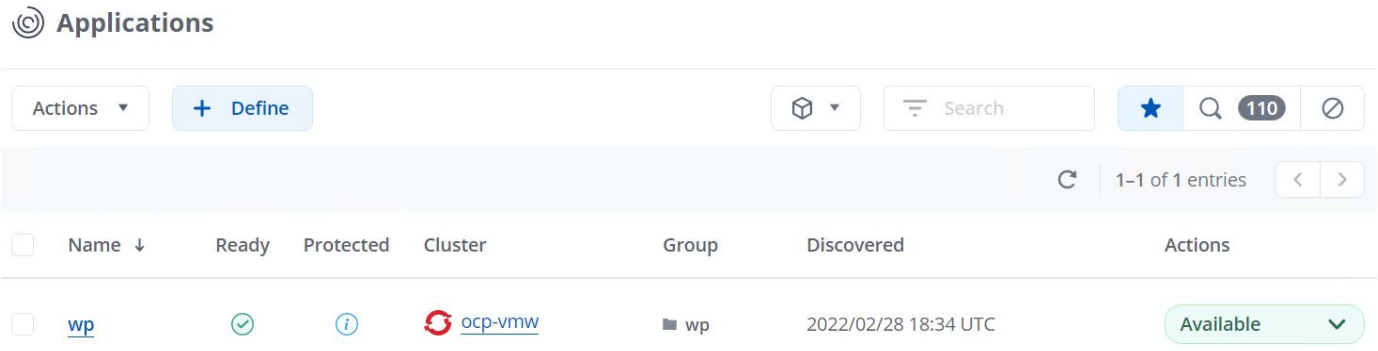
2. Entrez le nom de l'espace de noms de restauration, sélectionnez le cluster vers lequel vous souhaitez le restaurer et choisissez si vous souhaitez le restaurer à partir d'un snapshot existant ou à partir d'une sauvegarde de l'application. Cliquez sur Suivant.



3. Dans le volet de révision, entrez `restore` Puis cliquez sur Restaurer une fois que vous avez examiné les détails.



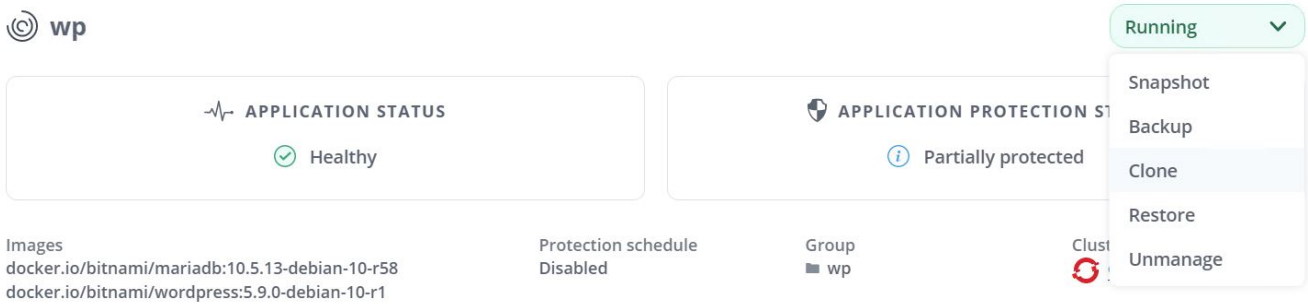
4. La nouvelle application passe à l'état de restauration tandis qu'Astra Control Center restaure l'application sur le cluster sélectionné. Une fois que toutes les ressources de l'application sont installées et détectées par Astra, l'application passe à l'état disponible.



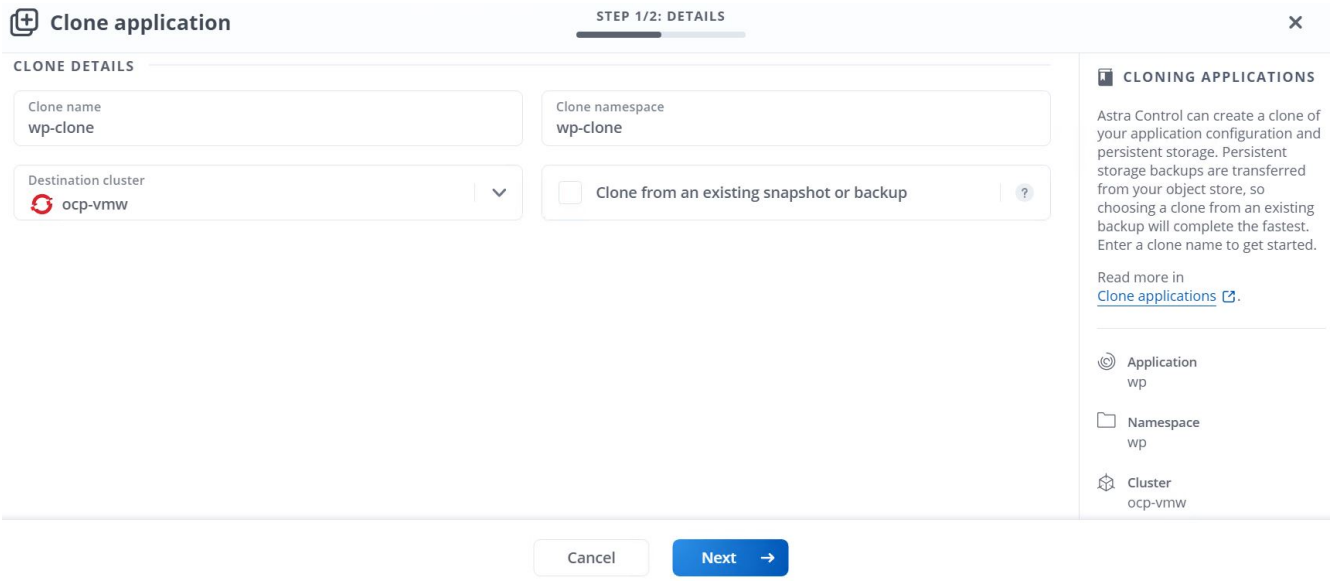
Clonage d'une application

Vous pouvez cloner une application sur le cluster d'origine ou sur un cluster distant à des fins de développement/test ou de protection des applications et de reprise sur incident. Le clonage d'une application au sein d'un même cluster sur le même système back-end utilise la technologie NetApp FlexClone, qui clonez instantanément les demandes de volume persistant et économise de l'espace de stockage.

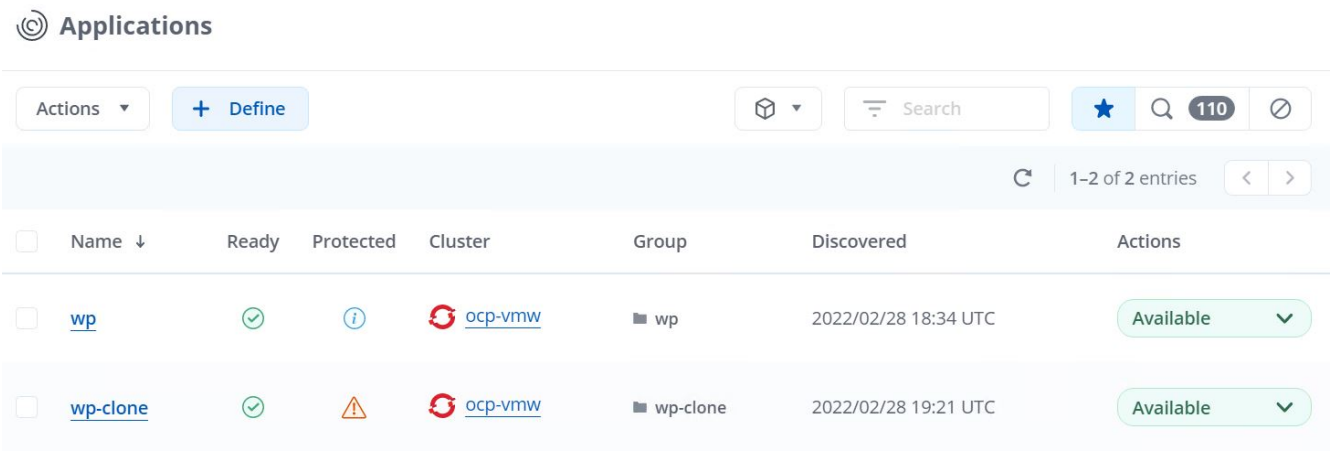
1. Pour cloner une application, accédez à l'onglet applications > gestion et cliquez sur l'application en question. Cliquez sur le menu déroulant en regard du nom de l'application, puis cliquez sur Cloner.



2. Entrez les détails du nouveau namespace, sélectionnez le cluster vers lequel vous souhaitez le cloner à partir d'un snapshot existant ou d'une sauvegarde ou de l'état actuel de l'application. Cliquez ensuite sur Suivant et sur Cloner dans le volet d'évaluation une fois que vous avez passé en revue les détails.



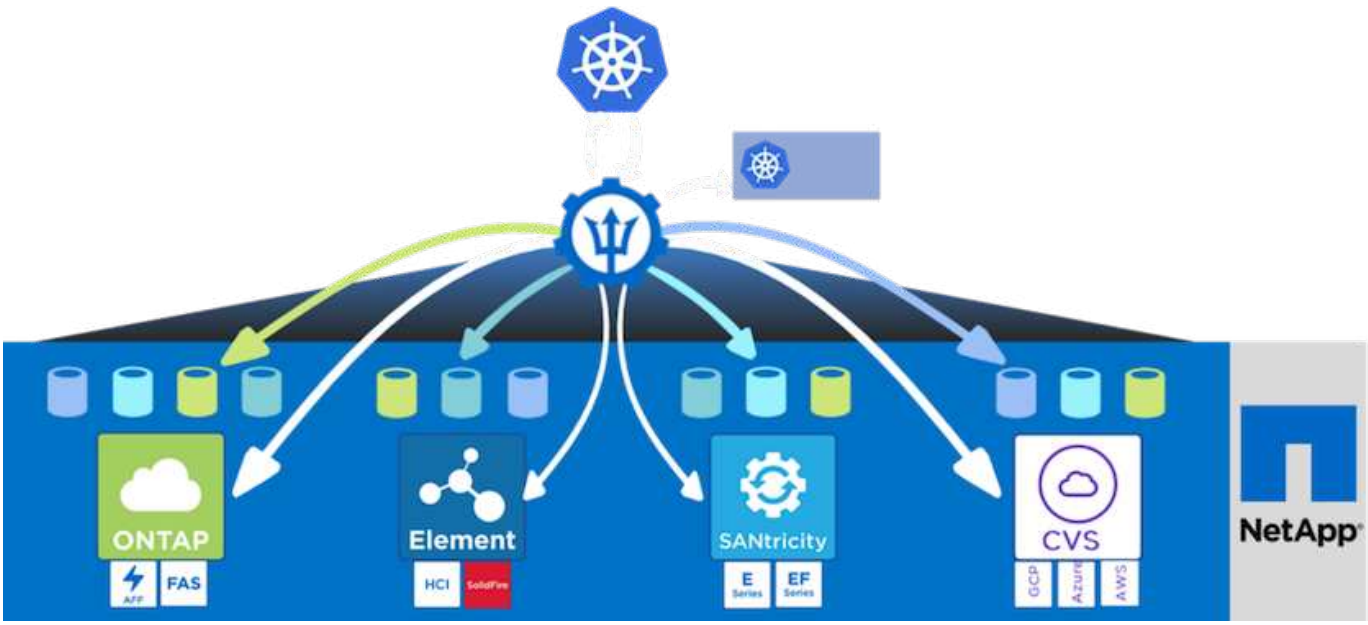
3. La nouvelle application passe à l'état découverte tandis que Astra Control Center crée l'application sur le cluster sélectionné. Une fois que toutes les ressources de l'application sont installées et détectées par Astra, l'application passe à l'état disponible.



Présentation d'Astra Trident

Astra Trident est un orchestrateur de stockage open source entièrement pris en charge, y compris Red Hat OpenShift, pour les conteneurs et les distributions Kubernetes. Trident fonctionne avec l'ensemble de la gamme de solutions de stockage NetApp, notamment les systèmes de stockage NetApp ONTAP et Element, et prend également en charge les connexions NFS et iSCSI. Trident accélère le workflow DevOps en permettant aux utilisateurs d'approvisionner et de gérer le stockage à partir de leurs systèmes de stockage NetApp, sans intervention de l'administrateur de stockage.

Un administrateur peut configurer plusieurs systèmes de stockage back-end en fonction des besoins des projets et des modèles de système de stockage. Ces fonctionnalités permettent notamment la compression, des types de disques spécifiques ou des niveaux de QoS garantissant un certain niveau de performance. Une fois définis, ces systèmes back-end peuvent être utilisés par les développeurs dans leurs projets pour créer des demandes de volume persistant et connecter le stockage persistant à la demande dans leurs conteneurs.



Astra Trident a un cycle de développement rapide, et comme Kubernetes, est lancé quatre fois par an.

La dernière version d'Astra Trident est 22.01 publiée en janvier 2022. Une matrice de prise en charge pour quelle version de Trident a été testée avec laquelle une distribution Kubernetes est disponible "ici".

Depuis la version 20.04, l'opérateur Trident effectue la configuration de Trident. L'opérateur facilite les déploiements à grande échelle et offre un support supplémentaire, notamment l'auto-rétablissement des pods déployés dans le cadre de l'installation de Trident.

Avec la version 21.01, un graphique Helm a été disponible pour faciliter l'installation de l'opérateur Trident.

Téléchargez Astra Trident

Pour installer Trident sur le cluster utilisateur déployé et provisionner un volume persistant, procédez comme suit :

1. Téléchargez l'archive d'installation sur la station de travail d'administration et extrayez son contenu. La version actuelle de Trident est la version 22.01, que vous pouvez télécharger "ici".

```
[netapp-user@rhel7 ~]$ wget
https://github.com/NetApp/trident/releases/download/v22.01.0/trident-
installer-22.01.0.tar.gz
--2021-05-06 15:17:30--
https://github.com/NetApp/trident/releases/download/v22.01.0/trident-
installer-22.01.0.tar.gz
Resolving github.com (github.com)... 140.82.114.3
Connecting to github.com (github.com)|140.82.114.3|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://github-
releases.githubusercontent.com/77179634/a4fa9f00-a9f2-11eb-9053-
98e8e573d4ae?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-
Credential=AKIAIWNJYAX4CSVEH53A%2F20210506%2Fus-east-
1%2Fs3%2Faws4_request&X-Amz-Date=20210506T191643Z&X-Amz-Expires=300&X-
Amz-
Signature=8a49a2a1e08c147d1ddd8149ce45a5714f9853fee19bb1c507989b9543eb36
30&X-Amz-
SignedHeaders=host&actor_id=0&key_id=0&repo_id=77179634&response-
content-disposition=attachment%3B%20filename%3Dtrident-installer-
22.01.0.tar.gz&response-content-type=application%2Foctet-stream
[following]
--2021-05-06 15:17:30--  https://github-
releases.githubusercontent.com/77179634/a4fa9f00-a9f2-11eb-9053-
98e8e573d4ae?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-
Credential=AKIAIWNJYAX4CSVEH53A%2F20210506%2Fus-east-
1%2Fs3%2Faws4_request&X-Amz-Date=20210506T191643Z&X-Amz-Expires=300&X-
Amz-
Signature=8a49a2a1e08c147d1ddd8149ce45a5714f9853fee19bb1c507989b9543eb36
30&X-Amz-
SignedHeaders=host&actor_id=0&key_id=0&repo_id=77179634&response-
content-disposition=attachment%3B%20filename%3Dtrident-installer-
```

```
22.01.0.tar.gz&response-content-type=application%2Foctet-stream
Resolving github-releases.githubusercontent.com (github-
releases.githubusercontent.com)... 185.199.108.154, 185.199.109.154,
185.199.110.154, ...
Connecting to github-releases.githubusercontent.com (github-
releases.githubusercontent.com)|185.199.108.154|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 38349341 (37M) [application/octet-stream]
Saving to: 'trident-installer-22.01.0.tar.gz'

100%[===================================================================
==============================================>] 38,349,341  88.5MB/s
in 0.4s

2021-05-06 15:17:30 (88.5 MB/s) - 'trident-installer-22.01.0.tar.gz'
saved [38349341/38349341]
```

2. Extrayez l'installation de Trident du bundle téléchargé.

```
[netapp-user@rhel7 ~]$ tar -xzf trident-installer-22.01.0.tar.gz
[netapp-user@rhel7 ~]$ cd trident-installer/
[netapp-user@rhel7 trident-installer]$
```

Installer l'opérateur Trident avec Helm

1. Définissez tout d'abord l'emplacement du cluster utilisateur `kubeconfig` Fichier en tant que variable d'environnement pour que vous n'ayez pas à le référencer, car Trident n'a pas d'option pour transmettre ce fichier.

```
[netapp-user@rhel7 trident-installer]$ export KUBECONFIG=~/ocp-
install/auth/kubeconfig
```

2. Lancer la commande Helm pour installer l'opérateur Trident à partir du tarball dans le répertoire Helm lors de la création du namespace trident dans le cluster utilisateur.

```
[netapp-user@rhel7 trident-installer]$ helm install trident
helm/trident-operator-22.01.0.tgz --create-namespace --namespace trident
NAME: trident
LAST DEPLOYED: Fri May  7 12:54:25 2021
NAMESPACE: trident
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
Thank you for installing trident-operator, which will deploy and manage
NetApp's Trident CSI
storage provisioner for Kubernetes.

Your release is named 'trident' and is installed into the 'trident'
namespace.
Please note that there must be only one instance of Trident (and
trident-operator) in a Kubernetes cluster.

To configure Trident to manage storage resources, you will need a copy
of tridentctl, which is
available in pre-packaged Trident releases.  You may find all Trident
releases and source code
online at https://github.com/NetApp/trident.

To learn more about the release, try:

  $ helm status trident
  $ helm get all trident
```

3. Vous pouvez vérifier que Trident est correctement installé en vérifiant les pods qui s’exécutent dans l’espace de noms ou en utilisant le binaire tridentctl pour vérifier la version installée.

```
[netapp-user@rhel7 trident-installer]$ oc get pods -n trident
NAME                                READY   STATUS    RESTARTS   AGE
trident-csi-5z45l                   1/2     Running   2          30s
trident-csi-696b685cf8-htdb2        6/6     Running   0          30s
trident-csi-b74p2                   2/2     Running   0          30s
trident-csi-lrw4n                   2/2     Running   0          30s
trident-operator-7c748d957-gr2gw    1/1     Running   0          36s

[netapp-user@rhel7 trident-installer]$ ./tridentctl -n trident version
+----------------+----------------+
| SERVER VERSION | CLIENT VERSION |
+----------------+----------------+
| 22.01.0          | 22.01.0          |
+----------------+----------------+
```

Dans certains cas, il est possible que les environnements client nécessitent la personnalisation du déploiement Trident. Dans ce cas, il est également possible d'installer manuellement l'opérateur Trident et de mettre à jour les manifestes inclus pour personnaliser le déploiement.

Installez manuellement l'opérateur Trident

1. Commencez par définir l'emplacement du cluster utilisateur `kubeconfig` Fichier en tant que variable d'environnement pour que vous n'ayez pas à le référencer, car Trident n'a pas d'option pour transmettre ce fichier.

```
[netapp-user@rhel7 trident-installer]$ export KUBECONFIG=~/ocp-
install/auth/kubeconfig
```

2. Le `trident-installer` le répertoire contient des manifestes pour définir toutes les ressources requises. À l'aide des manifestes appropriés, créer le `TridentOrchestrator` définition de ressource personnalisée.

```
[netapp-user@rhel7 trident-installer]$ oc create -f
deploy/crds/trident.netapp.io_tridentorchestrators_crd_post1.16.yaml
customresourcedefinition.apiextensions.k8s.io/tridentorchestrators.tride
nt.netapp.io created
```

3. Si aucun n'existe, créez un espace de nom Trident dans le cluster à l'aide du manifeste fourni.

```
[netapp-user@rhel7 trident-installer]$ oc apply -f deploy/namespace.yaml
namespace/trident created
```

4. Créez les ressources requises pour le déploiement par un opérateur Trident, par exemple un

`ServiceAccount` pour l'opérateur, un `ClusterRole` et `ClusterRoleBinding` à la `ServiceAccount`, un dédié `PodSecurityPolicy`, ou l'opérateur lui-même.

```
[netapp-user@rhel7 trident-installer]$ oc create -f deploy/bundle.yaml
serviceaccount/trident-operator created
clusterrole.rbac.authorization.k8s.io/trident-operator created
clusterrolebinding.rbac.authorization.k8s.io/trident-operator created
deployment.apps/trident-operator created
podsecuritypolicy.policy/tridentoperatorpods created
```

5. Vous pouvez vérifier l'état de l'opérateur après son déploiement à l'aide des commandes suivantes :

```
[netapp-user@rhel7 trident-installer]$ oc get deployment -n trident
NAME               READY   UP-TO-DATE   AVAILABLE   AGE
trident-operator   1/1     1            1           23s
[netapp-user@rhel7 trident-installer]$ oc get pods -n trident
NAME                                READY   STATUS    RESTARTS   AGE
trident-operator-66f48895cc-lzczk   1/1     Running   0          41s
```

6. Une fois l'opérateur déployé, nous pouvons maintenant l'utiliser pour installer Trident. Cela nécessite la création d'un `TridentOrchestrator`.

```
[netapp-user@rhel7 trident-installer]$ oc create -f
deploy/crds/tridentorchestrator_cr.yaml
tridentorchestrator.trident.netapp.io/trident created
[netapp-user@rhel7 trident-installer]$ oc describe torc trident
Name:         trident
Namespace:
Labels:       <none>
Annotations:  <none>
API Version:  trident.netapp.io/v1
Kind:         TridentOrchestrator
Metadata:
  Creation Timestamp:  2021-05-07T17:00:28Z
  Generation:          1
  Managed Fields:
    API Version:  trident.netapp.io/v1
    Fields Type:  FieldsV1
    fieldsV1:
      f:spec:
        .:
        f:debug:
        f:namespace:
    Manager:      kubectl-create
```

```
    Operation:    Update
    Time:         2021-05-07T17:00:28Z
    API Version:  trident.netapp.io/v1
    Fields Type:  FieldsV1
    fieldsV1:
      f:status:
        .:
        f:currentInstallationParams:
          .:
          f:IPv6:
          f:autosupportHostname:
          f:autosupportImage:
          f:autosupportProxy:
          f:autosupportSerialNumber:
          f:debug:
          f:enableNodePrep:
          f:imagePullSecrets:
          f:imageRegistry:
          f:k8sTimeout:
          f:kubeletDir:
          f:logFormat:
          f:silenceAutosupport:
          f:tridentImage:
        f:message:
        f:namespace:
        f:status:
        f:version:
    Manager:         trident-operator
    Operation:       Update
    Time:            2021-05-07T17:00:28Z
  Resource Version:  931421
  Self Link:
/apis/trident.netapp.io/v1/tridentorchestrators/trident
  UID:               8a26a7a6-dde8-4d55-9b66-a7126754d81f
Spec:
  Debug:      true
  Namespace:  trident
Status:
  Current Installation Params:
    IPv6:                       false
    Autosupport Hostname:
    Autosupport Image:          netapp/trident-autosupport:21.01
    Autosupport Proxy:
    Autosupport Serial Number:
    Debug:                      true
    Enable Node Prep:           false
```

```
    Image Pull Secrets:
    Image Registry:
    k8sTimeout:           30
    Kubelet Dir:          /var/lib/kubelet
    Log Format:           text
    Silence Autosupport:  false
    Trident Image:        netapp/trident:22.01.0
  Message:                Trident installed
  Namespace:              trident
  Status:                 Installed
  Version:                v22.01.0
Events:
  Type    Reason      Age   From                        Message
  ----    ------      ----  ----                        -------
  Normal  Installing  80s   trident-operator.netapp.io  Installing
Trident
  Normal  Installed   68s   trident-operator.netapp.io  Trident
installed
```

7. Vous pouvez vérifier que Trident est correctement installé en vérifiant les pods qui s'exécutent dans l'espace de noms ou en utilisant le binaire tridentctl pour vérifier la version installée.

```
[netapp-user@rhel7 trident-installer]$ oc get pods -n trident
NAME                                READY   STATUS    RESTARTS   AGE
trident-csi-bb64c6cb4-lmd6h         6/6     Running   0          82s
trident-csi-gn59q                   2/2     Running   0          82s
trident-csi-m4szj                   2/2     Running   0          82s
trident-csi-sb9k9                   2/2     Running   0          82s
trident-operator-66f48895cc-lzczk   1/1     Running   0          2m39s

[netapp-user@rhel7 trident-installer]$ ./tridentctl -n trident version
+----------------+----------------+
| SERVER VERSION | CLIENT VERSION |
+----------------+----------------+
| 22.01.0          | 22.01.0          |
+----------------+----------------+
```

Préparez les nœuds workers pour le stockage

NFS

La plupart des distributions Kubernetes sont fournies avec des packages et des utilitaires permettant de monter les systèmes back-end NFS installés par défaut, y compris Red Hat OpenShift.

Cependant, pour NFSv3, il n'existe aucun mécanisme pour négocier la simultanéité entre le client et le serveur. Par conséquent, le nombre maximal d'entrées de la table d'emplacements sunrpc côté client doit être

synchronisé manuellement avec la valeur prise en charge sur le serveur pour assurer les meilleures performances de la connexion NFS sans que le serveur n'ait à diminuer la taille de la fenêtre de la connexion.

Pour ONTAP, le nombre maximal d'entrées de la table des emplacements sunrpc pris en charge est de 128, c'est-à-dire que ONTAP peut traiter 128 requêtes NFS simultanées à la fois. Cependant, par défaut, Red Hat CoreOS/Red Hat Enterprise Linux possède au maximum 65,536 entrées de table sunrpc par connexion. Nous devons définir cette valeur sur 128 et cela peut être fait à l'aide de l'opérateur de configuration machine (MCO) d'OpenShift.

Pour modifier le nombre maximal d'entrées de la table d'emplacements sunrpc dans les nœuds de travail OpenShift, procédez comme suit :

1. Connectez-vous à la console Web OCP et accédez à Compute > machine configurations. Cliquez sur Créer une configuration de machine. Copiez et collez le fichier YAML, puis cliquez sur Créer.

```
apiVersion: machineconfiguration.openshift.io/v1
kind: MachineConfig
metadata:
  name: 98-worker-nfs-rpc-slot-tables
  labels:
    machineconfiguration.openshift.io/role: worker
spec:
  config:
    ignition:
      version: 3.2.0
    storage:
      files:
        - contents:
            source: data:text/plain;charset=utf-
8; base64,b3B0aW9ucyBzdW5ycGMgdGNwX21heF9zbG90X3RhYmxlX2VudHJpZXM9MTI4Cg=
=
          filesystem: root
          mode: 420
          path: /etc/modprobe.d/sunrpc.conf
```

2. Une fois le MCO créé, la configuration doit être appliquée à tous les nœuds workers et redémarrée un par un. Le processus prend entre 20 et 30 minutes environ. Vérifiez si la configuration de la machine est appliquée à l'aide de `oc get mcp` et assurez-vous que le pool de configuration de la machine pour les employés est mis à jour.

```
[netapp-user@rhel7 openshift-deploy]$ oc get mcp
NAME     CONFIG                                     UPDATED   UPDATING
DEGRADED
master   rendered-master-a520ae930e1d135e0dee7168   True      False
False
worker   rendered-worker-de321b36eeba62df41feb7bc   True      False
False
```

ISCSI

Pour préparer les nœuds workers afin de permettre le mappage des volumes de stockage en mode bloc via le protocole iSCSI, vous devez installer les packages nécessaires pour prendre en charge cette fonctionnalité.

Dans Red Hat OpenShift, ces opérations sont gérées via l'application d'un MCO (opérateur de configuration de machine) à votre cluster après son déploiement.

Pour configurer les nœuds workers pour exécuter des services iSCSI, procédez comme suit :

1. Connectez-vous à la console Web OCP et accédez à Compute > machine configurations. Cliquez sur Créer une configuration de machine. Copiez et collez le fichier YAML, puis cliquez sur Créer.

   Lorsque vous n'utilisez pas les chemins d'accès multiples :

   ```
   apiVersion: machineconfiguration.openshift.io/v1
   kind: MachineConfig
   metadata:
     labels:
       machineconfiguration.openshift.io/role: worker
     name: 99-worker-element-iscsi
   spec:
     config:
       ignition:
         version: 3.2.0
       systemd:
         units:
           - name: iscsid.service
             enabled: true
             state: started
     osImageURL: ""
   ```

   Lorsque vous utilisez les chemins d'accès multiples :

```
apiVersion: machineconfiguration.openshift.io/v1
kind: MachineConfig
metadata:
  name: 99-worker-ontap-iscsi
  labels:
    machineconfiguration.openshift.io/role: worker
spec:
  config:
    ignition:
      version: 3.2.0
    storage:
      files:
      - contents:
          source: data:text/plain;charset=utf-
8; base64,ZGVmYXVsdHMgewogICAgICAgIHVzZXJfZnJpZW5kbHlfbmFtZXMgbm8KICAgICA
gICBmaW5kX211bHRpcGF0aHMgbm8KfQoKYmxhY2tsaXN0X2V4Y2VwdGlvbnMgewogICAgICA
gIHByb3BlcnR5ICIoU0NTSV9JREVOVF98SURfV1dOKSIKfQoKYmxhY2tsaXN0IHsKfQoK
          verification: {}
        filesystem: root
        mode: 400
        path: /etc/multipath.conf
    systemd:
      units:
        - name: iscsid.service
          enabled: true
          state: started
        - name: multipathd.service
          enabled: true
          state: started
  osImageURL: ""
```

2. Une fois la configuration créée, il faut environ 20 à 30 minutes pour appliquer la configuration aux nœuds worker et les recharger. Vérifiez si la configuration de la machine est appliquée à l'aide de `oc get mcp` et assurez-vous que le pool de configuration de la machine pour les employés est mis à jour. Vous pouvez également vous connecter aux nœuds workers pour vérifier que le service iscsid est en cours d'exécution (et que le service multipathd est exécuté en cas d'utilisation de chemins d'accès multiples).

```
[netapp-user@rhel7 openshift-deploy]$ oc get mcp
NAME     CONFIG                                     UPDATED   UPDATING
DEGRADED
master   rendered-master-a520ae930e1d135e0dee7168   True      False
False
worker   rendered-worker-de321b36eeba62df41feb7bc   True      False
False

[netapp-user@rhel7 openshift-deploy]$ ssh core@10.61.181.22 sudo
systemctl status iscsid
● iscsid.service - Open-iSCSI
   Loaded: loaded (/usr/lib/systemd/system/iscsid.service; enabled;
vendor preset: disabled)
   Active: active (running) since Tue 2021-05-26 13:36:22 UTC; 3 min ago
     Docs: man:iscsid(8)
           man:iscsiadm(8)
 Main PID: 1242 (iscsid)
   Status: "Ready to process requests"
    Tasks: 1
   Memory: 4.9M
      CPU: 9ms
   CGroup: /system.slice/iscsid.service
           └─1242 /usr/sbin/iscsid -f

[netapp-user@rhel7 openshift-deploy]$ ssh core@10.61.181.22 sudo
systemctl status multipathd
 ● multipathd.service - Device-Mapper Multipath Device Controller
   Loaded: loaded (/usr/lib/systemd/system/multipathd.service; enabled;
vendor preset: enabled)
   Active: active (running) since Tue 2021-05-26 13:36:22 UTC; 3 min ago
  Main PID: 918 (multipathd)
    Status: "up"
    Tasks: 7
    Memory: 13.7M
    CPU: 57ms
    CGroup: /system.slice/multipathd.service
            └─918 /sbin/multipathd -d -s
```

> Il est également possible de confirmer que la MachineConfig a été appliquée avec succès et que les services ont été lancés comme prévu en exécutant le `oc debug` commande avec les indicateurs appropriés.

Création de systèmes back-end de stockage

Une fois l'installation d'Astra Trident Operator, vous devez configurer le système back-end pour la plateforme

de stockage NetApp spécifique que vous utilisez. Suivre les liens ci-dessous pour poursuivre l'installation et la configuration d'Astra Trident.

- "NetApp ONTAP NFS"
- "ISCSI NetApp ONTAP"
- "ISCSI NetApp Element"

Configuration NetApp ONTAP NFS

Pour activer l'intégration de Trident avec le système de stockage NetApp ONTAP, il faut créer un back-end permettant la communication avec le système de stockage.

1. Des exemples de fichiers backend sont disponibles dans l'archive d'installation téléchargée dans le `sample-input` hiérarchie des dossiers. Pour les systèmes NetApp ONTAP servant de protocole NFS, copiez le `backend-ontap-nas.json` dans votre répertoire de travail et modifiez le fichier.

```
[netapp-user@rhel7 trident-installer]$ cp sample-input/backends-
samples/ontap-nas/backend-ontap-nas.json ./
[netapp-user@rhel7 trident-installer]$ vi backend-ontap-nas.json
```

2. Modifier le backendName, la gestion LIF, dataLIF, svm, nom d'utilisateur, et les valeurs de mot de passe dans ce fichier.

```
{
    "version": 1,
    "storageDriverName": "ontap-nas",
    "backendName": "ontap-nas+10.61.181.221",
    "managementLIF": "172.21.224.201",
    "dataLIF": "10.61.181.221",
    "svm": "trident_svm",
    "username": "cluster-admin",
    "password": "password"
}
```

> Il est recommandé de définir la valeur backendName personnalisée comme combinaison du storageDriverName et de la dataLIF qui sert NFS pour une identification facile.

3. Lorsque ce fichier backend est en place, exécutez la commande suivante pour créer votre premier back-end.

```
[netapp-user@rhel7 trident-installer]$ ./tridentctl -n trident create
backend -f backend-ontap-nas.json
+------------------------+----------------
+--------------------------------------+--------+---------+
|          NAME          | STORAGE DRIVER |                  UUID
| STATE  | VOLUMES |
+------------------------+----------------
+--------------------------------------+--------+---------+
| ontap-nas+10.61.181.221 | ontap-nas      | be7a619d-c81d-445c-b80c-
5c87a73c5b1e | online |       0 |
+------------------------+----------------
+--------------------------------------+--------+---------+
```

4. Lorsque le back-end est créé, vous devez ensuite créer une classe de stockage. Tout comme pour le back-end, il existe un exemple de fichier de classe de stockage qui peut être modifié pour l'environnement disponible dans le dossier des échantillons-entrées. Copiez-le dans le répertoire de travail et apportez les modifications nécessaires pour refléter le back-end créé.

```
[netapp-user@rhel7 trident-installer]$ cp sample-input/storage-class-
samples/storage-class-csi.yaml.templ ./storage-class-basic.yaml
[netapp-user@rhel7 trident-installer]$ vi storage-class-basic.yaml
```

5. La seule modification à effectuer dans ce fichier consiste à définir le `backendType` valeur du nom du pilote de stockage du back-end nouvellement créé. Notez également la valeur nom-champ, qui doit être référencée ultérieurement.

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: basic-csi
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-nas"
```

> Il y a un champ facultatif appelé `fsType` qui est défini dans ce fichier. Cette ligne peut être supprimée dans les systèmes back-end NFS.

6. Exécutez le `oc` pour créer la classe de stockage.

```
[netapp-user@rhel7 trident-installer]$ oc create -f storage-class-
basic.yaml
storageclass.storage.k8s.io/basic-csi created
```

7. Une fois la classe de stockage créée, vous devez ensuite créer la première demande de volume persistant. Il y a un échantillon `pvc-basic.yaml` fichier qui peut être utilisé pour effectuer cette action également située dans les entrées d'échantillons.

```
[netapp-user@rhel7 trident-installer]$ cp sample-input/pvc-samples/pvc-
basic.yaml ./
[netapp-user@rhel7 trident-installer]$ vi pvc-basic.yaml
```

8. La seule modification à effectuer dans ce fichier est de s'assurer que `storageClassName` le champ correspond à celui que vous venez de créer. La définition du volume persistant peut être personnalisée davantage selon les besoins de la charge de travail à provisionner.

```
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: basic
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 1Gi
  storageClassName: basic-csi
```

9. Créez le PVC en émettant le `oc` commande. La création peut prendre un certain temps en fonction de la taille du volume de sauvegarde en cours de création, de sorte que vous pouvez regarder le processus au fur et à mesure qu'il se termine.

```
[netapp-user@rhel7 trident-installer]$ oc create -f pvc-basic.yaml
persistentvolumeclaim/basic created

[netapp-user@rhel7 trident-installer]$ oc get pvc
NAME    STATUS   VOLUME                                     CAPACITY
ACCESS MODES   STORAGECLASS   AGE
basic   Bound    pvc-b4370d37-0fa4-4c17-bd86-94f96c94b42d   1Gi
RWO            basic-csi      7s
```

Configuration ONTAP iSCSI de NetApp

Pour activer l'intégration de Trident avec le système de stockage NetApp ONTAP, il faut créer un back-end permettant la communication avec le système de stockage.

1. Des exemples de fichiers backend sont disponibles dans l'archive d'installation téléchargée dans le `sample-input` hiérarchie des dossiers. Pour les systèmes NetApp ONTAP servant iSCSI, copiez le `backend-ontap-san.json` dans votre répertoire de travail et modifiez le fichier.

```
[netapp-user@rhel7 trident-installer]$ cp sample-input/backends-
samples/ontap-san/backend-ontap-san.json ./
[netapp-user@rhel7 trident-installer]$ vi backend-ontap-san.json
```

2. Modifiez les valeurs LIF, dataLIF, svm, nom d'utilisateur et mot de passe dans ce fichier.

```
{
  "version": 1,
  "storageDriverName": "ontap-san",
  "managementLIF": "172.21.224.201",
  "dataLIF": "10.61.181.240",
  "svm": "trident_svm",
  "username": "admin",
  "password": "password"
}
```

3. Lorsque ce fichier backend est en place, exécutez la commande suivante pour créer votre premier back-end.

```
[netapp-user@rhel7 trident-installer]$ ./tridentctl -n trident create
backend -f backend-ontap-san.json
+------------------------+----------------
+--------------------------------------+--------+---------+
|          NAME          | STORAGE DRIVER |                  UUID
| STATE  | VOLUMES |
+------------------------+----------------
+--------------------------------------+--------+---------+
| ontapsan_10.61.181.241 | ontap-san      | 6788533c-7fea-4a35-b797-
fb9bb3322b91 | online |       0 |
+------------------------+----------------
+--------------------------------------+--------+---------+
```

4. Lorsque le back-end est créé, vous devez ensuite créer une classe de stockage. Tout comme pour le back-end, il existe un exemple de fichier de classe de stockage qui peut être modifié pour l'environnement disponible dans le dossier des échantillons-entrées. Copiez-le dans le répertoire de travail et apportez les modifications nécessaires pour refléter le back-end créé.

```
[netapp-user@rhel7 trident-installer]$ cp sample-input/storage-class-
samples/storage-class-csi.yaml.templ ./storage-class-basic.yaml
[netapp-user@rhel7 trident-installer]$ vi storage-class-basic.yaml
```

5. La seule modification à effectuer dans ce fichier consiste à définir le `backendType` valeur du nom du pilote de stockage du back-end nouvellement créé. Notez également la valeur nom-champ, qui doit être

référencée ultérieurement.

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: basic-csi
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-san"
```

> Il y a un champ facultatif appelé `fsType` qui est défini dans ce fichier. Dans les systèmes back-end iSCSI, cette valeur peut être définie sur un type de système de fichiers Linux spécifique (XFS, ext4, etc.) ou peut être supprimée pour permettre à OpenShift de décider du système de fichiers à utiliser.

6. Exécutez le `oc` pour créer la classe de stockage.

```
[netapp-user@rhel7 trident-installer]$ oc create -f storage-class-
basic.yaml
storageclass.storage.k8s.io/basic-csi created
```

7. Une fois la classe de stockage créée, vous devez ensuite créer la première demande de volume persistant. Il y a un échantillon `pvc-basic.yaml` fichier qui peut être utilisé pour effectuer cette action également située dans les entrées d'échantillons.

```
[netapp-user@rhel7 trident-installer]$ cp sample-input/pvc-samples/pvc-
basic.yaml ./
[netapp-user@rhel7 trident-installer]$ vi pvc-basic.yaml
```

8. La seule modification à effectuer dans ce fichier est de s'assurer que `storageClassName` le champ correspond à celui que vous venez de créer. La définition du volume persistant peut être personnalisée davantage selon les besoins de la charge de travail à provisionner.

```
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: basic
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 1Gi
  storageClassName: basic-csi
```

9. Créez le PVC en émettant le `oc` commande. La création peut prendre un certain temps en fonction de la taille du volume de sauvegarde en cours de création, de sorte que vous pouvez regarder le processus au fur et à mesure qu'il se termine.

```
[netapp-user@rhel7 trident-installer]$ oc create -f pvc-basic.yaml
persistentvolumeclaim/basic created

[netapp-user@rhel7 trident-installer]$ oc get pvc
NAME    STATUS   VOLUME                                     CAPACITY
ACCESS MODES   STORAGECLASS   AGE
basic   Bound    pvc-7ceac1ba-0189-43c7-8f98-094719f7956c   1Gi
RWO            basic-csi      3s
```

Configuration iSCSI de NetApp Element

Pour activer l'intégration de Trident avec le système de stockage NetApp Element, vous devez créer un back-end permettant la communication avec le système de stockage via le protocole iSCSI.

1. Des exemples de fichiers backend sont disponibles dans l'archive d'installation téléchargée dans le `sample-input` hiérarchie des dossiers. Pour les systèmes NetApp Element servant iSCSI, copiez le `backend-solidfire.json` dans votre répertoire de travail et modifiez le fichier.

```
[netapp-user@rhel7 trident-installer]$ cp sample-input/backends-
samples/solidfire/backend-solidfire.json ./
[netapp-user@rhel7 trident-installer]$ vi ./backend-solidfire.json
```

   a. Modifiez l'utilisateur, le mot de passe et la valeur MVIP sur le `EndPoint` ligne.

   b. Modifiez le `SVIP` valeur.

```
 {
    "version": 1,
    "storageDriverName": "solidfire-san",
    "Endpoint": "https://trident:password@172.21.224.150/json-
rpc/8.0",
    "SVIP": "10.61.180.200:3260",
    "TenantName": "trident",
    "Types": [{"Type": "Bronze", "Qos": {"minIOPS": 1000, "maxIOPS":
2000, "burstIOPS": 4000}},
              {"Type": "Silver", "Qos": {"minIOPS": 4000, "maxIOPS":
6000, "burstIOPS": 8000}},
              {"Type": "Gold", "Qos": {"minIOPS": 6000, "maxIOPS":
8000, "burstIOPS": 10000}}]
}
```

2. Avec ce fichier back-end en place, exécutez la commande suivante pour créer votre premier back-end.

```
[netapp-user@rhel7 trident-installer]$ ./tridentctl -n trident create
backend -f backend-solidfire.json
+------------------------+----------------
+--------------------------------------+--------+---------+
|          NAME          | STORAGE DRIVER |                  UUID
| STATE  | VOLUMES |
+------------------------+----------------
+--------------------------------------+--------+---------+
| solidfire_10.61.180.200 | solidfire-san  | b90783ee-e0c9-49af-8d26-
3ea87ce2efdf | online |       0 |
+------------------------+----------------
+--------------------------------------+--------+---------+
```

3. Lorsque le back-end est créé, vous devez ensuite créer une classe de stockage. Tout comme pour le back-end, il existe un exemple de fichier de classe de stockage qui peut être modifié pour l'environnement disponible dans le dossier des échantillons-entrées. Copiez-le dans le répertoire de travail et apportez les modifications nécessaires pour refléter le back-end créé.

```
[netapp-user@rhel7 trident-installer]$ cp sample-input/storage-class-
samples/storage-class-csi.yaml.templ ./storage-class-basic.yaml
[netapp-user@rhel7 trident-installer]$ vi storage-class-basic.yaml
```

4. La seule modification à effectuer dans ce fichier consiste à définir le `backendType` valeur du nom du pilote de stockage du back-end nouvellement créé. Notez également la valeur nom-champ, qui doit être référencée ultérieurement.

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: basic-csi
provisioner: csi.trident.netapp.io
parameters:
  backendType: "solidfire-san"
```

> Il y a un champ facultatif appelé `fsType` qui est défini dans ce fichier. Dans les systèmes back-end iSCSI, cette valeur peut être définie sur un type de système de fichiers Linux spécifique (XFS, ext4, etc.), ou elle peut être supprimée pour permettre à OpenShift de décider du système de fichiers à utiliser.

5. Exécutez le `oc` pour créer la classe de stockage.

```
[netapp-user@rhel7 trident-installer]$ oc create -f storage-class-
basic.yaml
storageclass.storage.k8s.io/basic-csi created
```

6. Une fois la classe de stockage créée, vous devez ensuite créer la première demande de volume persistant. Il y a un échantillon `pvc-basic.yaml` fichier qui peut être utilisé pour effectuer cette action également située dans les entrées d'échantillons.

```
[netapp-user@rhel7 trident-installer]$ cp sample-input/pvc-samples/pvc-
basic.yaml ./
[netapp-user@rhel7 trident-installer]$ vi pvc-basic.yaml
```

7. La seule modification à effectuer dans ce fichier est de s'assurer que `storageClassName` le champ correspond à celui que vous venez de créer. La définition du volume persistant peut être personnalisée davantage selon les besoins de la charge de travail à provisionner.

```
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: basic
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 1Gi
  storageClassName: basic-csi
```

8. Créez le PVC en émettant le `oc` commande. La création peut prendre un certain temps en fonction de la taille du volume de sauvegarde en cours de création, de sorte que vous pouvez regarder le processus au fur et à mesure qu'il se termine.

```
[netapp-user@rhel7 trident-installer]$ oc create -f pvc-basic.yaml
persistentvolumeclaim/basic created

[netapp-user@rhel7 trident-installer]$ oc get pvc
NAME    STATUS   VOLUME                                     CAPACITY
ACCESS MODES   STORAGECLASS   AGE
basic   Bound    pvc-3445b5cc-df24-453d-a1e6-b484e874349d   1Gi
RWO            basic-csi      5s
```

Options de configuration avancées

Exploration des options d'équilibreur de charge avec Red Hat OpenShift avec NetApp

Dans la plupart des cas, Red Hat OpenShift met les applications à la disposition du monde extérieur via des routes. Un service est exposé en lui donnant un nom d'hôte accessible en externe. La route définie et les points de terminaison identifiés par son service peuvent être utilisés par un routeur OpenShift pour fournir cette connectivité nommée aux clients externes.

Cependant, dans certains cas, les applications nécessitent le déploiement et la configuration d'équilibreurs de charge personnalisés pour exposer les services appropriés. Il s'agit notamment du centre de contrôle NetApp Astra, Pour répondre à ce besoin, nous avons évalué un certain nombre d'options d'équilibrage de charge personnalisé. Leur installation et leur configuration sont décrites dans cette section.

Les pages suivantes présentent des informations supplémentaires sur les options de équilibreur de charge validées dans la solution Red Hat OpenShift avec NetApp :

- "MetalLB"
- "F5 BIG-IP"

Installation d'équilibreurs de charge MetalLB : Red Hat OpenShift avec NetApp

Cette page répertorie les instructions d'installation et de configuration de l'équilibreur de charge MetalLB.

MetalLB est un équilibreur de charge réseau hébergé automatiquement sur votre cluster OpenShift qui permet la création de services OpenShift d'équilibreur de charge dans les clusters qui ne s'exécutent pas sur un fournisseur cloud. Les deux principales caractéristiques de MetalLB qui fonctionnent ensemble pour prendre en charge les services LoadBalancer sont l'allocation d'adresses et l'annonce externe.

Options de configuration MetalLB

D'après la façon dont MetalLB annonce l'adresse IP attribuée aux services LoadBalancer en dehors du cluster OpenShift, elle fonctionne selon deux modes :

- **Mode de couche 2.** dans ce mode, un nœud du cluster OpenShift est propriétaire du service et répond aux demandes ARP pour cette IP pour la rendre accessible en dehors du cluster OpenShift. Seul le nœud

annonce l'IP, il présente un goulot d'étranglement au niveau de la bande passante et des limitations de basculement lentes. Pour plus d'informations, reportez-vous à la documentation "ici".

- **Mode BGP.** dans ce mode, tous les nœuds du cluster OpenShift établissent des sessions de peering BGP avec un routeur et annoncent les routes pour transférer le trafic vers les adresses IP du service. La condition préalable est d'intégrer MetalLB à un routeur de ce réseau. En raison du mécanisme de hachage dans BGP, il possède une certaine limite lors du mappage d'IP à nœud pour les modifications de service. Pour plus d'informations, reportez-vous à la documentation "ici".

> Pour les besoins de ce document, nous allons configurer MetalLB en mode couche 2.

Installation de l'équilibreur de charge MetalLB

1. Téléchargez les ressources MetalLB.

```
[netapp-user@rhel7 ~]$ wget
https://raw.githubusercontent.com/metallb/metallb/v0.10.2/manifests/name
space.yaml
[netapp-user@rhel7 ~]$ wget
https://raw.githubusercontent.com/metallb/metallb/v0.10.2/manifests/meta
llb.yaml
```

2. Modifier le fichier `metallb.yaml` et déposer `spec.template.spec.securityContext` À partir du déploiement du contrôleur et de l'ensemble des haut-parleurs.

   Lignes à supprimer :

```
securityContext:
  runAsNonRoot: true
  runAsUser: 65534
```

3. Créer le `metallb-system` espace de noms.

```
[netapp-user@rhel7 ~]$ oc create -f namespace.yaml
namespace/metallb-system created
```

4. Créer la CR du MetalLB.

```
[netapp-user@rhel7 ~]$ oc create -f metallb.yaml
podsecuritypolicy.policy/controller created
podsecuritypolicy.policy/speaker created
serviceaccount/controller created
serviceaccount/speaker created
clusterrole.rbac.authorization.k8s.io/metallb-system:controller created
clusterrole.rbac.authorization.k8s.io/metallb-system:speaker created
role.rbac.authorization.k8s.io/config-watcher created
role.rbac.authorization.k8s.io/pod-lister created
role.rbac.authorization.k8s.io/controller created
clusterrolebinding.rbac.authorization.k8s.io/metallb-system:controller
created
clusterrolebinding.rbac.authorization.k8s.io/metallb-system:speaker
created
rolebinding.rbac.authorization.k8s.io/config-watcher created
rolebinding.rbac.authorization.k8s.io/pod-lister created
rolebinding.rbac.authorization.k8s.io/controller created
daemonset.apps/speaker created
deployment.apps/controller created
```

5. Avant de configurer le haut-parleur MetalLB, accordez à l'intervenant DemonSet des privilèges élevés afin qu'il puisse effectuer la configuration réseau requise pour que les équilibreurs de charge fonctionnent.

```
[netapp-user@rhel7 ~]$ oc adm policy add-scc-to-user privileged -n
metallb-system -z speaker
clusterrole.rbac.authorization.k8s.io/system:openshift:scc:privileged
added: "speaker"
```

6. Configurez MetalLB en créant un `ConfigMap` dans le `metallb-system` espace de noms.

```
[netapp-user@rhel7 ~]$ vim metallb-config.yaml

apiVersion: v1
kind: ConfigMap
metadata:
  namespace: metallb-system
  name: config
data:
  config: |
    address-pools:
    - name: default
      protocol: layer2
      addresses:
      - 10.63.17.10-10.63.17.200

[netapp-user@rhel7 ~]$ oc create -f metallb-config.yaml
configmap/config created
```

7. Maintenant que des services loadBALB sont créés, MetalLB attribue un IP externe aux services et annonce l'adresse IP en répondant aux demandes ARP.

> Si vous souhaitez configurer MetalLB en mode BGP, ignorez l'étape 6 ci-dessus et suivez la procédure décrite dans la documentation MetalLB "ici".

Installation des presses à balles F5 BIG-IP Load Balancers

F5 BIG-IP est un contrôleur de distribution d'applications (ADC) qui offre un large éventail de services avancés de gestion du trafic et de sécurité de niveau production, tels que L4-L7 d'équilibrage de charge, d'allègement de la charge SSL/TLS, de DNS, de pare-feu et bien d'autres. Ces services augmentent considérablement la disponibilité, la sécurité et les performances de vos applications.

F5 BIG-IP peut être déployé et utilisé de différentes façons, sur un matériel dédié, dans le cloud ou comme appliance virtuelle sur site. Reportez-vous à la documentation ici pour explorer et déployer F5 BIG-IP selon les besoins.

Pour une intégration efficace des services F5 BIG-IP avec Red Hat OpenShift, F5 propose UN service CIS (BIG-IP Container Ingress Service). CIS est installé en tant que pod de contrôleur qui surveille l'API OpenShift pour certaines définitions de ressources personnalisées (CRD) et gère la configuration système F5 BIG-IP. F5 BIG-IP CIS peut être configuré pour contrôler les types de service LoadBalancers et les routes dans OpenShift.

En outre, pour l'allocation automatique d'adresses IP pour le traitement du type LoadBalancer, vous pouvez utiliser le contrôleur F5 IPAM. Le contrôleur F5 IPAM est installé comme un pod de contrôleur qui surveille l'API OpenShift pour les services LoadBalancer avec une annotation ipamLabel afin d'allouer l'adresse IP à partir d'un pool préconfiguré.

Cette page répertorie les instructions d'installation et de configuration pour F5 BIG-IP CIS et contrôleur IPAM. Un système F5 BIG-IP doit être déployé et sous licence. Il doit également être concédé sous licence pour les services SDN, qui sont inclus par défaut avec la licence de base BIG-IP VE.

F5 BIG-IP peut être déployé en mode autonome ou cluster. Aux fins de cette validation, F5 BIG-IP a été déployé en mode autonome, mais pour la production, il est préférable d'avoir un cluster de BIG-IP pour éviter un seul point de défaillance.

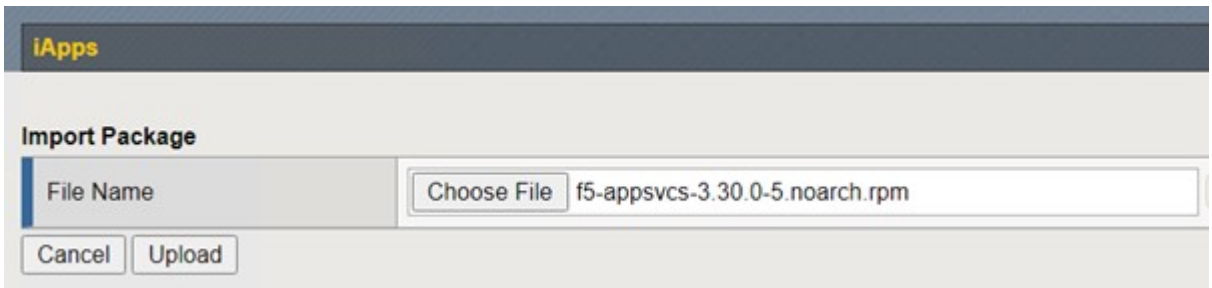
Un système F5 BIG-IP peut être déployé sur un matériel dédié, dans le cloud ou en tant qu'appliance virtuelle sur site avec des versions supérieures à 12.x pour une intégration avec F5 CIS. Dans le cadre de ce document, le système F5 BIG-IP a été validé en tant qu'appliance virtuelle, par exemple en utilisant L'édition BIG-IP VE.

Versions validées

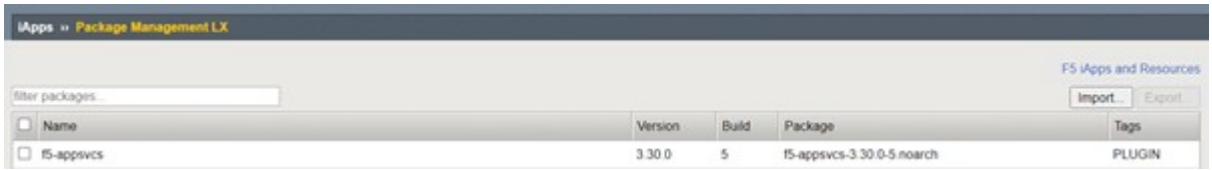
De déduplication	Version logicielle
Red Hat OpenShift	4.6 EUS, 4.7
F5 BIG-IP VE EDITION	16.1.0
Service F5 d'entrée de conteneur	2.5.1
Contrôleur F5 IPAM	0.1.4
AS3 F5	3.30.0

Installation

1. Installez l'extension F5 application Services 3 pour permettre aux systèmes BIG-IP d'accepter les configurations au format JSON au lieu de commandes impérative. Accédez à "Référentiel GitHub F5 AS3"Et téléchargez le dernier fichier RPM.
2. Connectez-vous au système F5 BIG-IP, accédez à iApps > Package Management LX et cliquez sur Importer.
3. Cliquez sur choisir un fichier et sélectionnez le fichier RPM AS3 téléchargé, cliquez sur OK, puis cliquez sur Télécharger.

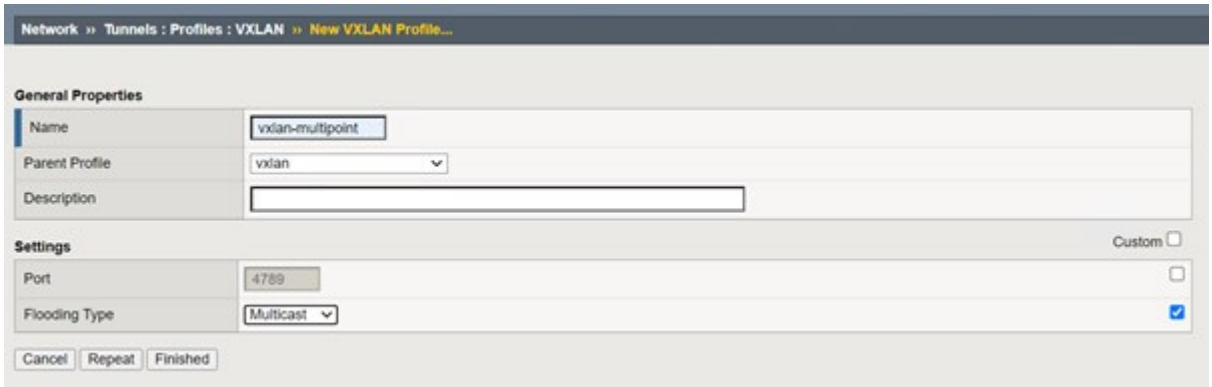


4. Vérifiez que l'extension AS3 est correctement installée.

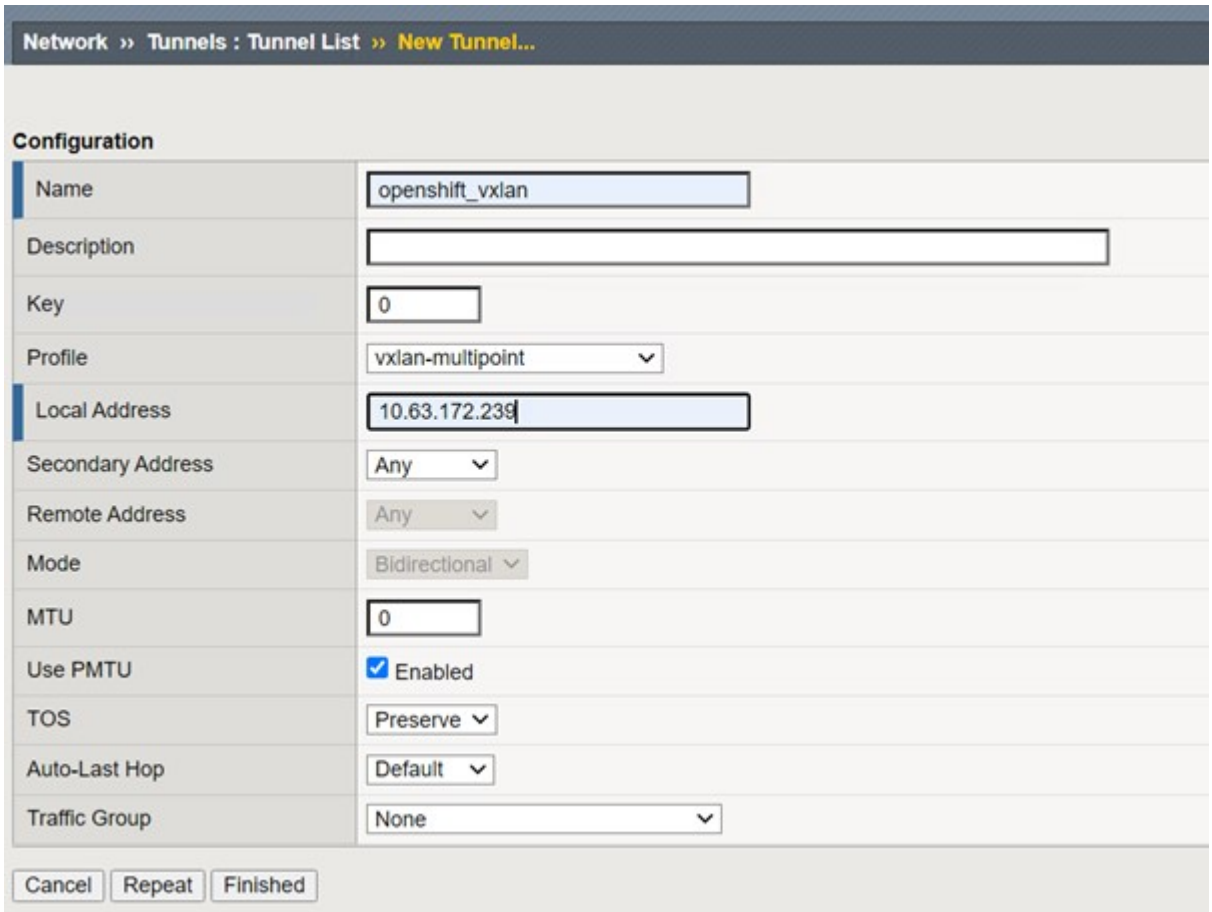


5. Configurez ensuite les ressources requises pour la communication entre les systèmes OpenShift et BIG-IP. Commencez par créer un tunnel entre OpenShift et LE serveur BIG-IP en créant une interface de tunnel VXLAN sur le système BIG-IP pour OpenShift SDN. Naviguez jusqu'à réseau > tunnels > profils, cliquez sur Créer, puis définissez le profil parent sur vxlan et le type d'inondation sur Multicast. Entrez un nom pour

le profil et cliquez sur terminé.



6. Naviguez jusqu'à réseau > tunnels > liste de tunnels, cliquez sur Créer, puis entrez le nom et l'adresse IP locale du tunnel. Sélectionnez le profil de tunnel créé à l'étape précédente et cliquez sur terminé.



7. Connectez-vous au cluster Red Hat OpenShift avec les privilèges cluster-admin.
8. Créez un sous-réseau d'hôtes sur OpenShift pour le serveur F5 BIG-IP, qui étend le sous-réseau du cluster OpenShift au serveur F5 BIG-IP. Téléchargez la définition YAML du sous-réseau hôte.

```
wget https://github.com/F5Networks/k8s-bigip-
ctlr/blob/master/docs/config_examples/openshift/f5-kctlr-openshift-
hostsubnet.yaml
```

9. Modifiez le fichier de sous-réseau de l'hôte et ajoutez l'IP VTEP (VXLAN tunnel) BIG-IP pour le SDN OpenShift.

```
apiVersion: v1
kind: HostSubnet
metadata:
  name: f5-server
  annotations:
    pod.network.openshift.io/fixed-vnid-host: "0"
    pod.network.openshift.io/assign-subnet: "true"
# provide a name for the node that will serve as BIG-IP's entry into the
cluster
host: f5-server
# The hostIP address will be the BIG-IP interface address routable to
the
# OpenShift Origin nodes.
# This address is the BIG-IP VTEP in the SDN's VXLAN.
hostIP: 10.63.172.239
```

Modifiez l'adresse IP de l'hôte et d'autres détails applicables à votre environnement.

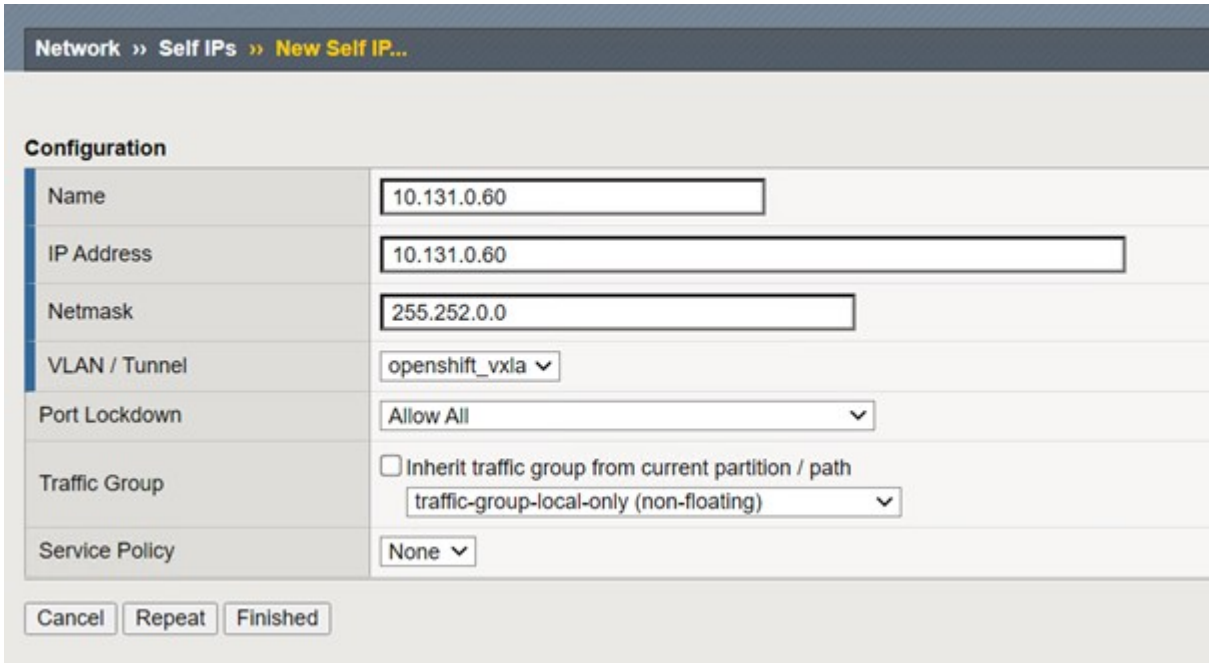
10. Créez la ressource HostSubnet.

```
[admin@rhel-7 ~]$ oc create -f f5-kctlr-openshift-hostsubnet.yaml

hostsubnet.network.openshift.io/f5-server created
```

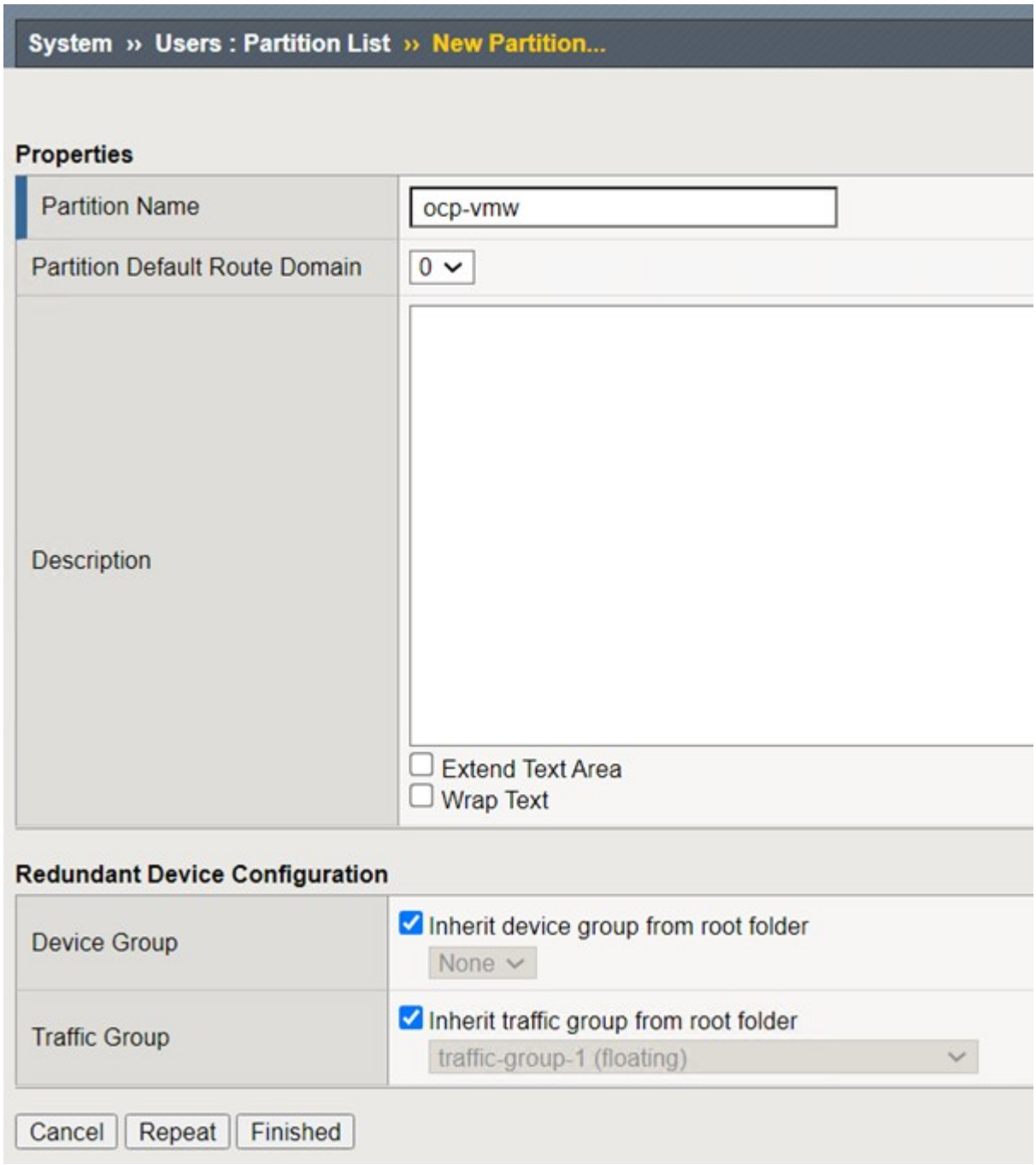
11. Obtenez la plage de sous-réseau IP du cluster pour le sous-réseau hôte créé pour le serveur F5 BIG-IP.

```
[admin@rhel-7 ~]$ oc get hostsubnet

NAME                          HOST                          HOST IP
SUBNET          EGRESS CIDRS   EGRESS IPS
f5-server                     f5-server                     10.63.172.239
10.131.0.0/23
ocp-vmw-nszws-master-0        ocp-vmw-nszws-master-0        10.63.172.44
10.128.0.0/23
ocp-vmw-nszws-master-1        ocp-vmw-nszws-master-1        10.63.172.47
10.130.0.0/23
ocp-vmw-nszws-master-2        ocp-vmw-nszws-master-2        10.63.172.48
10.129.0.0/23
ocp-vmw-nszws-worker-r8fh4    ocp-vmw-nszws-worker-r8fh4    10.63.172.7
10.130.2.0/23
ocp-vmw-nszws-worker-tvr46    ocp-vmw-nszws-worker-tvr46    10.63.172.11
10.129.2.0/23
ocp-vmw-nszws-worker-wdxhg    ocp-vmw-nszws-worker-wdxhg    10.63.172.24
10.128.2.0/23
ocp-vmw-nszws-worker-wg8r4    ocp-vmw-nszws-worker-wg8r4    10.63.172.15
10.131.2.0/23
ocp-vmw-nszws-worker-wtgfw    ocp-vmw-nszws-worker-wtgfw    10.63.172.17
10.128.4.0/23
```

12. Créez un auto-IP sur OpenShift VXLAN avec un IP dans la plage de sous-réseau hôte d'OpenShift correspondant au serveur F5 BIG-IP. Connectez-vous au système F5 BIG-IP, accédez à réseau > Auto-IP et cliquez sur Créer. Entrez une adresse IP à partir du sous-réseau IP du cluster créé pour le sous-réseau hôte F5 BIG-IP, sélectionnez le tunnel VXLAN et entrez les autres détails. Cliquez ensuite sur terminé.



13. Créez une partition dans le système F5 BIG-IP à configurer et à utiliser avec CIS. Accédez à système > utilisateurs > liste de partitions, cliquez sur Créer et entrez les détails. Cliquez ensuite sur terminé.



> F5 recommande de ne pas effectuer de configuration manuelle sur la partition gérée par CIS.

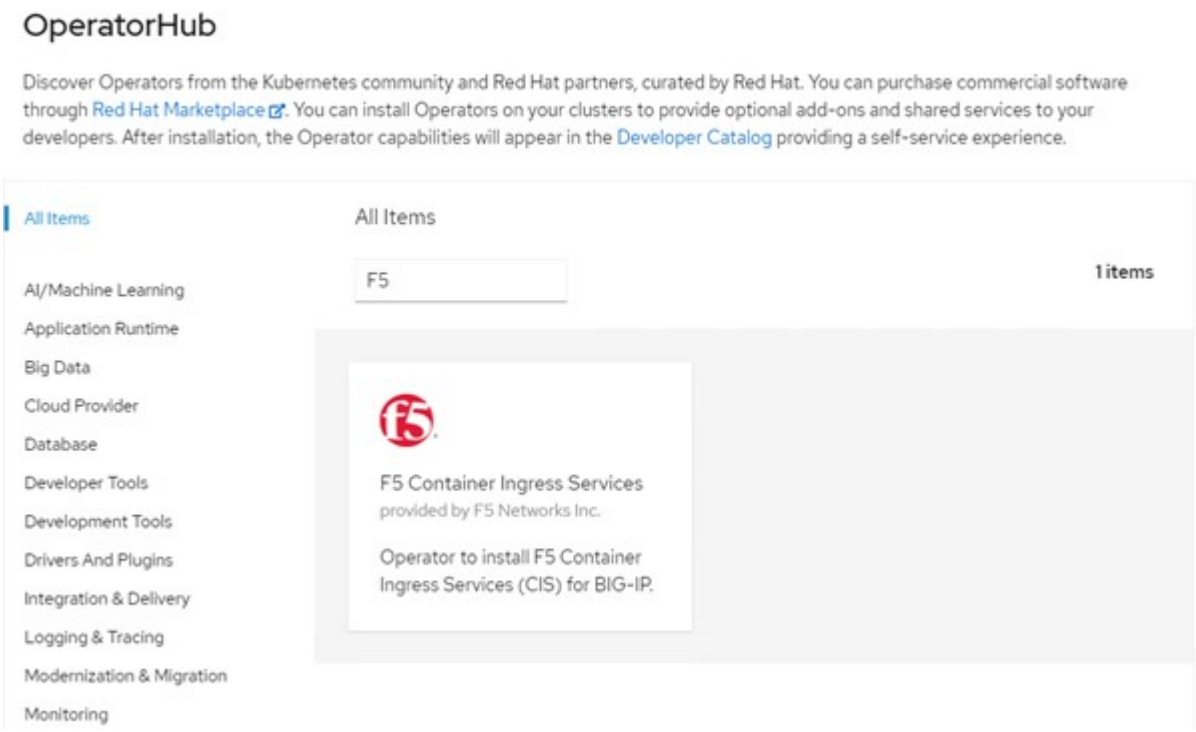
14. Installez F5 BIG-IP CIS à l'aide de l'opérateur depuis OperatorHub. Connectez-vous au cluster Red Hat OpenShift avec des privilèges cluster-admin et créez un secret avec les identifiants de connexion du système F5 BIG-IP. Il est indispensable pour l'opérateur.

```
[admin@rhel-7 ~]$ oc create secret generic bigip-login -n kube-system
--from-literal=username=admin --from-literal=password=admin

secret/bigip-login created
```

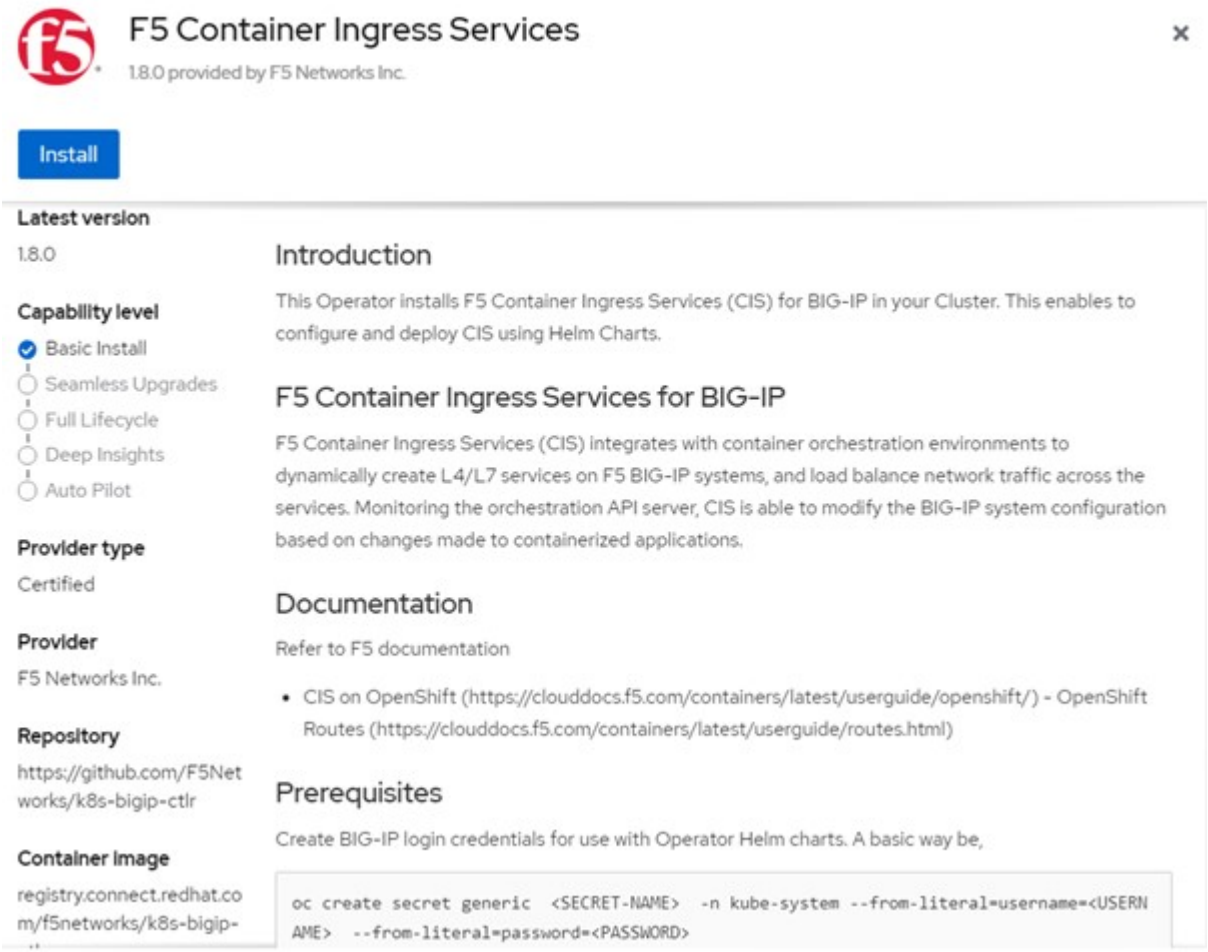
15. Installez les CRD F5 CIS.

```
[admin@rhel-7 ~]$ oc apply -f
https://raw.githubusercontent.com/F5Networks/k8s-bigip-
ctlr/master/docs/config_examples/crd/Install/customresourcedefinitions.y
ml

customresourcedefinition.apiextensions.k8s.io/virtualservers.cis.f5.com
created
customresourcedefinition.apiextensions.k8s.io/tlsprofiles.cis.f5.com
created
customresourcedefinition.apiextensions.k8s.io/transportservers.cis.f5.co
m created
customresourcedefinition.apiextensions.k8s.io/externaldnss.cis.f5.com
created
customresourcedefinition.apiextensions.k8s.io/ingresslinks.cis.f5.com
created
```

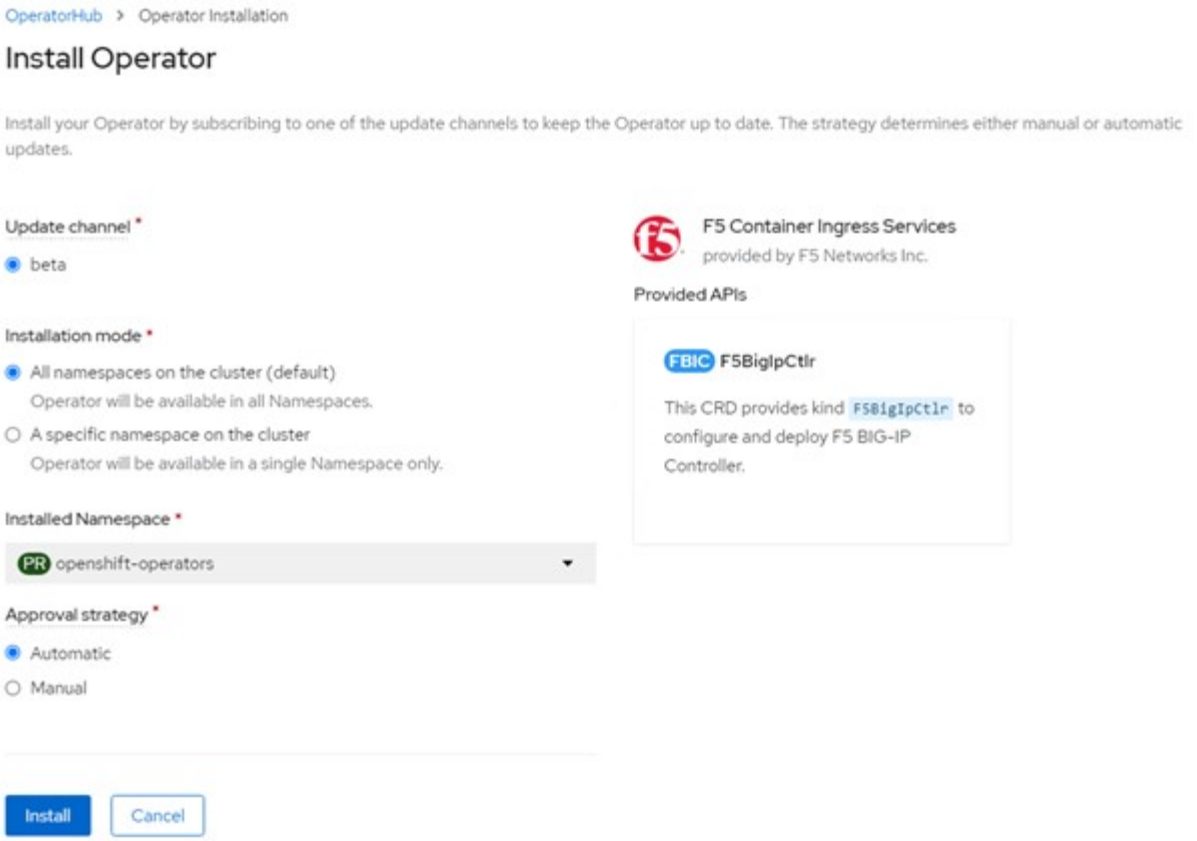
16. Accédez à Operators > OperatorHub, recherchez le mot-clé F5, puis cliquez sur la mosaïque F5 Container Ingress Service.



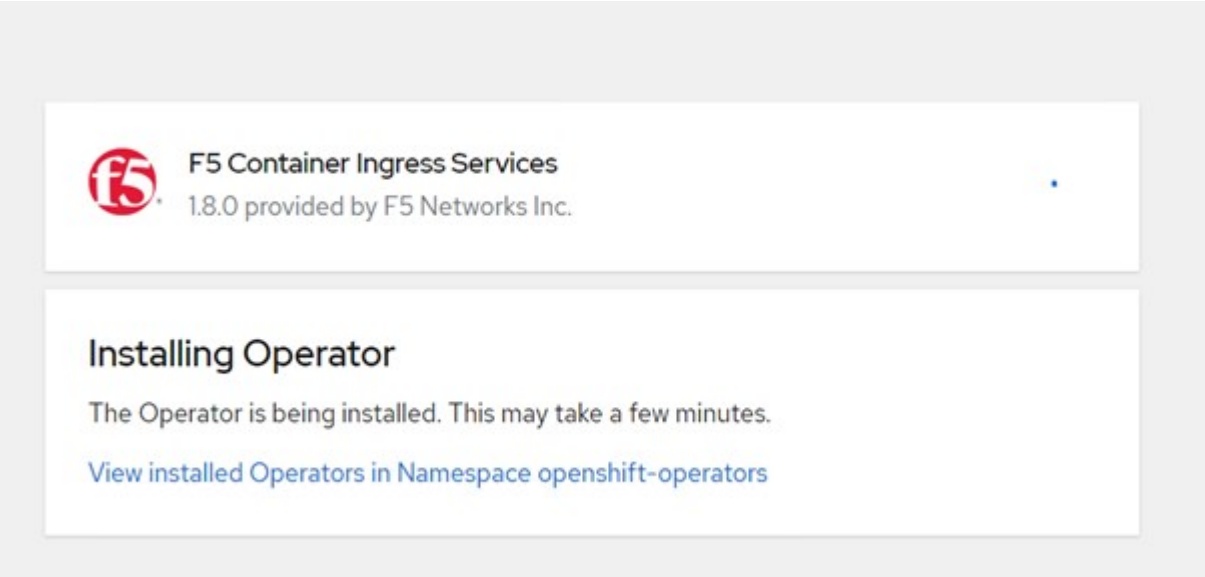
17. Lisez les informations de l'opérateur et cliquez sur installer.



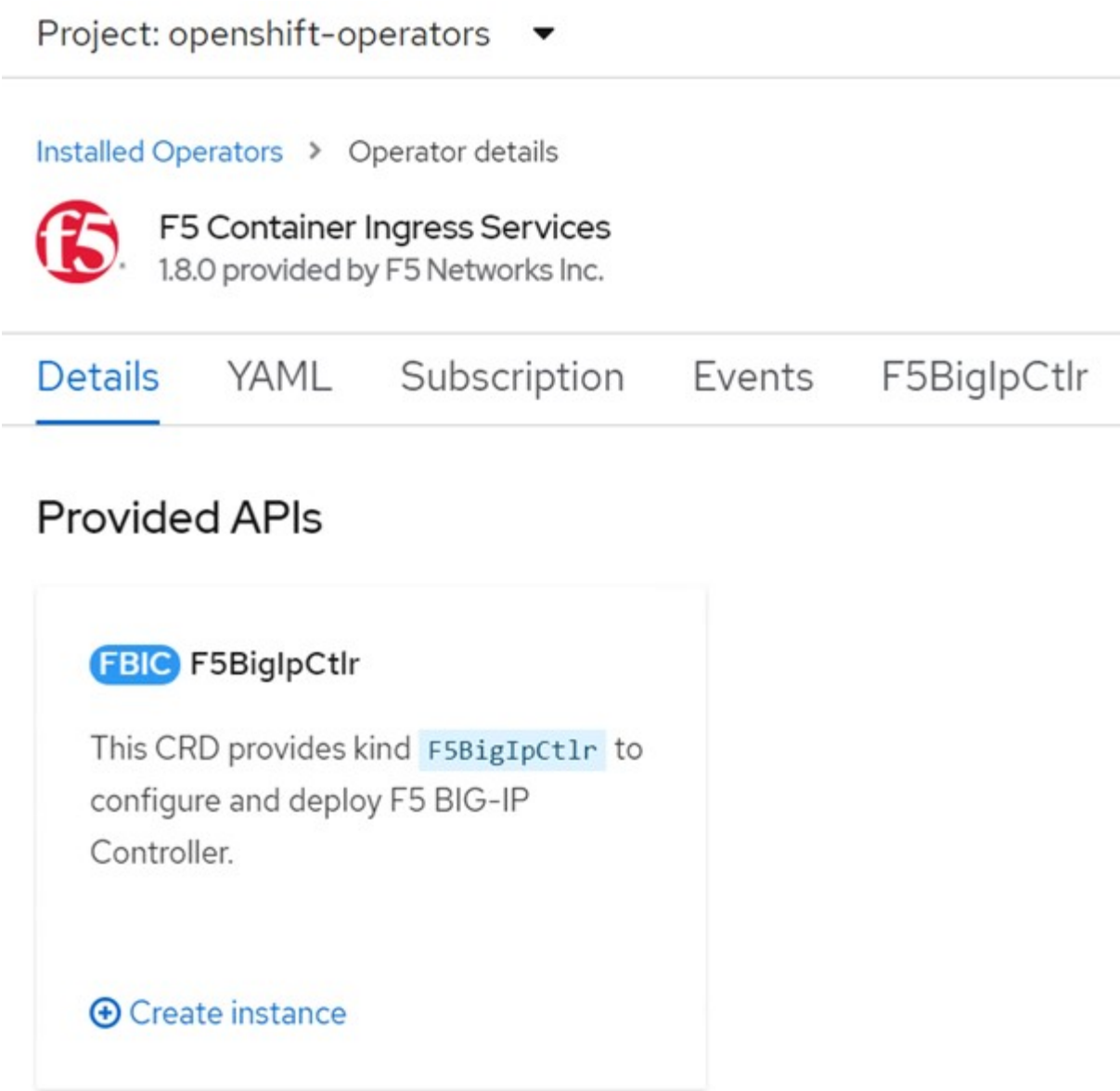
18. Sur l'écran de l'opérateur d'installation, conservez tous les paramètres par défaut, puis cliquez sur installer.



19. L'installation de l'opérateur prend un certain temps.



20. Une fois l'opérateur installé, le message installation réussie s'affiche.
21. Accédez à opérateurs > opérateurs installés, cliquez sur F5 Container Ingress Service, puis cliquez sur Créer une instance sous la mosaïque F5BigIpCtlr.

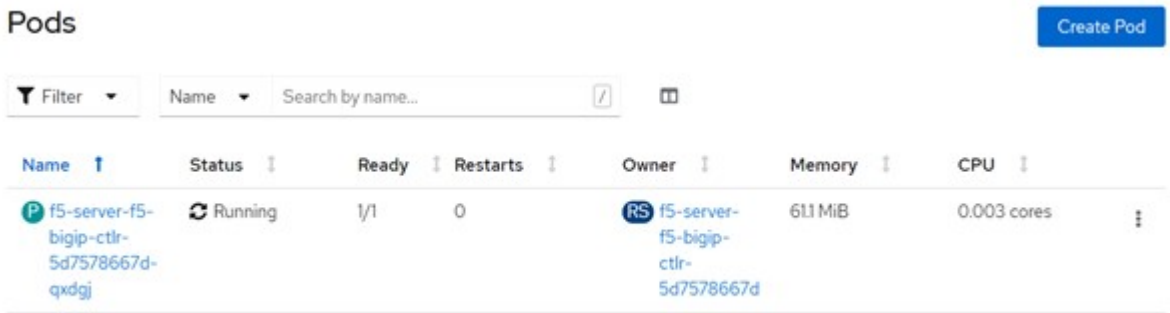


22. Cliquez sur vue YAML et collez le contenu suivant après la mise à jour des paramètres nécessaires.

> Mettre à jour les paramètres `bigip_partition`, ` openshift_sdn_name`, `bigip_url` et `bigip_login_secret` ci-dessous pour refléter les valeurs de votre configuration avant de copier le contenu.

```
apiVersion: cis.f5.com/v1
kind: F5BigIpCtlr
metadata:
  name: f5-server
  namespace: openshift-operators
spec:
  args:
    log_as3_response: true
    agent: as3
    log_level: DEBUG
    bigip_partition: ocp-vmw
    openshift_sdn_name: /Common/openshift_vxlan
    bigip_url: 10.61.181.19
    insecure: true
    pool-member-type: cluster
    custom_resource_mode: true
    as3_validation: true
    ipam: true
    manage_configmaps: true
  bigip_login_secret: bigip-login
  image:
    pullPolicy: Always
    repo: f5networks/cntr-ingress-svcs
    user: registry.connect.redhat.com
  namespace: kube-system
  rbac:
    create: true
  resources: {}
  serviceAccount:
    create: true
  version: latest
```

23. Après avoir collé ce contenu, cliquez sur Créer. Cela installe les modules CIS dans l'espace de noms du système kube.



Par défaut, Red Hat OpenShift permet d'exposer les services via des routes pour l'équilibrage de charge L7. Un routeur OpenShift intégré est chargé de la publicité et du traitement du trafic pour ces routes. Cependant, vous pouvez également configurer F5 CIS pour prendre en charge les routes via un système F5 BIG-IP externe, qui peut s'exécuter soit en tant que routeur auxiliaire, soit en remplacement du routeur OpenShift auto-hébergé. CIS crée un serveur virtuel dans le système BIG-IP qui sert de routeur pour les routes OpenShift, et BIG-IP gère la publicité et le routage du trafic. Pour plus d'informations sur les paramètres permettant d'activer cette fonctionnalité, reportez-vous à la documentation ci-dessous. Notez que ces paramètres sont définis pour la ressource OpenShift Deployment dans l'API apps/v1. Par conséquent, lors de l'utilisation de ces traits avec l'API F5BigIpCtlr ressource cis.f5.com/v1, remplacer les traits d'Union (-) par des traits de soulignement (_) pour les noms de paramètres.

24. Les arguments qui sont transmis à la création de ressources CIS sont notamment `ipam: true` et `custom_resource_mode: true`. Ces paramètres sont nécessaires pour activer l'intégration CIS avec un contrôleur IPAM. Vérifiez que le CIS a activé l'intégration IPAM en créant la ressource IP F5.

```
[admin@rhel-7 ~]$ oc get f5ipam -n kube-system

NAMESPACE   NAME                              AGE
kube-system   ipam.10.61.181.19.ocp-vmw      43s
```

25. Créez le compte de service, le rôle et la liaison en liaison rolerequises pour le contrôleur F5 IPAM. Créez un fichier YAML et collez le contenu suivant.

```
[admin@rhel-7 ~]$ vi f5-ipam-rbac.yaml

kind: ClusterRole
apiVersion: rbac.authorization.k8s.io/v1
metadata:
  name: ipam-ctlr-clusterrole
rules:
  - apiGroups: ["fic.f5.com"]
    resources: ["ipams","ipams/status"]
    verbs: ["get", "list", "watch", "update", "patch"]
---
kind: ClusterRoleBinding
apiVersion: rbac.authorization.k8s.io/v1
metadata:
  name: ipam-ctlr-clusterrole-binding
  namespace: kube-system
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: ClusterRole
  name: ipam-ctlr-clusterrole
subjects:
  - apiGroup: ""
    kind: ServiceAccount
    name: ipam-ctlr
    namespace: kube-system
---
apiVersion: v1
kind: ServiceAccount
metadata:
  name: ipam-ctlr
  namespace: kube-system
```

26. Créez les ressources.

```
[admin@rhel-7 ~]$ oc create -f f5-ipam-rbac.yaml

clusterrole.rbac.authorization.k8s.io/ipam-ctlr-clusterrole created
clusterrolebinding.rbac.authorization.k8s.io/ipam-ctlr-clusterrole-
binding created
serviceaccount/ipam-ctlr created
```

27. Créez un fichier YAML et collez la définition de déploiement IPAM F5 indiquée ci-dessous.

Mettez à jour le paramètre de plage ip dans spec.template.spec.containers[0].args ci-dessous pour refléter les plages d'adresses IP et ipamLabels correspondant à votre configuration.

IpamLabels [`range1` et `range2` Dans l'exemple ci-dessous] sont nécessaires pour être annotés pour les services de type LoadBalancer pour le contrôleur IPAM afin de détecter et d'affecter une adresse IP à partir de la plage définie.

```
[admin@rhel-7 ~]$ vi f5-ipam-deployment.yaml

apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    name: f5-ipam-controller
  name: f5-ipam-controller
  namespace: kube-system
spec:
  replicas: 1
  selector:
    matchLabels:
      app: f5-ipam-controller
  template:
    metadata:
      creationTimestamp: null
      labels:
        app: f5-ipam-controller
    spec:
      containers:
      - args:
        - --orchestration=openshift
        - --ip-range='{"range1":"10.63.172.242-10.63.172.249",
"range2":"10.63.170.111-10.63.170.129"}'
        - --log-level=DEBUG
        command:
        - /app/bin/f5-ipam-controller
        image: registry.connect.redhat.com/f5networks/f5-ipam-
controller:latest
        imagePullPolicy: IfNotPresent
        name: f5-ipam-controller
      dnsPolicy: ClusterFirst
      restartPolicy: Always
      schedulerName: default-scheduler
      securityContext: {}
      serviceAccount: ipam-ctlr
      serviceAccountName: ipam-ctlr
```

28. Créer le déploiement du contrôleur F5 IPAM.

```
[admin@rhel-7 ~]$ oc create -f f5-ipam-deployment.yaml

deployment/f5-ipam-controller created
```

29. Vérifiez que les modules de contrôleur F5 IPAM sont en cours d'exécution.

```
[admin@rhel-7 ~]$ oc get pods -n kube-system

NAME                                        READY   STATUS    RESTARTS
AGE
f5-ipam-controller-5986cff5bd-2bvn6         1/1     Running   0
30s
f5-server-f5-bigip-ctlr-5d7578667d-qxdgj    1/1     Running   0
14m
```

30. Créez le schéma F5 IPAM.

```
[admin@rhel-7 ~]$ oc create -f
https://raw.githubusercontent.com/F5Networks/f5-ipam-
controller/main/docs/_static/schemas/ipam_schema.yaml

customresourcedefinition.apiextensions.k8s.io/ipams.fic.f5.com
```

Vérification

1. Créez un service de type LoadBalancer

```
[admin@rhel-7 ~]$ vi example_svc.yaml

apiVersion: v1
kind: Service
metadata:
  annotations:
    cis.f5.com/ipamLabel: range1
  labels:
    app: f5-demo-test
  name: f5-demo-test
  namespace: default
spec:
  ports:
  - name: f5-demo-test
    port: 80
    protocol: TCP
    targetPort: 80
  selector:
    app: f5-demo-test
  sessionAffinity: None
  type: LoadBalancer
```

```
[admin@rhel-7 ~]$ oc create -f example_svc.yaml

service/f5-demo-test created
```

2. Vérifiez si le contrôleur IPAM lui attribue une adresse IP externe.

```
[admin@rhel-7 ~]$ oc get svc

NAME           TYPE           CLUSTER-IP       EXTERNAL-IP
PORT(S)        AGE
f5-demo-test   LoadBalancer   172.30.210.108   10.63.172.242
80:32605/TCP   27s
```

3. Créez un déploiement et utilisez le service LoadBalancer qui a été créé.

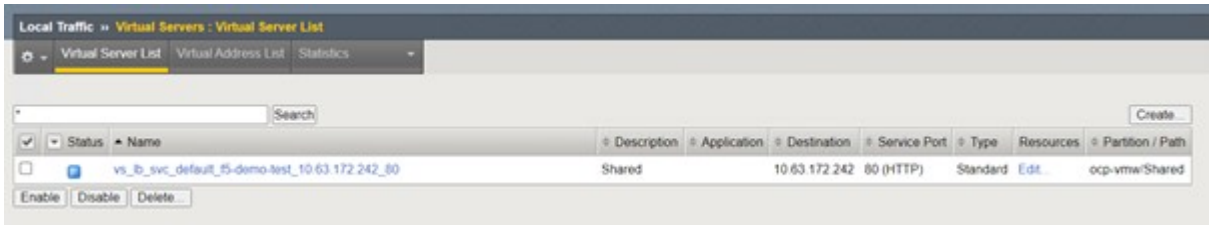
```
[admin@rhel-7 ~]$ vi example_deployment.yaml

apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    app: f5-demo-test
  name: f5-demo-test
spec:
  replicas: 2
  selector:
    matchLabels:
      app: f5-demo-test
  template:
    metadata:
      labels:
        app: f5-demo-test
    spec:
      containers:
      - env:
        - name: service_name
          value: f5-demo-test
        image: nginx
        imagePullPolicy: Always
        name: f5-demo-test
        ports:
        - containerPort: 80
          protocol: TCP
```

```
[admin@rhel-7 ~]$ oc create -f example_deployment.yaml

deployment/f5-demo-test created
```

4. Vérifiez que les modules sont en cours d'exécution.

```
[admin@rhel-7 ~]$ oc get pods

NAME                            READY   STATUS    RESTARTS   AGE
f5-demo-test-57c46f6f98-47wwp   1/1     Running   0          27s
f5-demo-test-57c46f6f98-cl2m8   1/1     Running   0          27s
```

5. Vérifiez si le serveur virtuel correspondant est créé dans LE système BIG-IP pour le service de type LoadBalancer dans OpenShift. Accédez à trafic local > serveurs virtuels > liste de serveurs virtuels.



Création de registres d'images privées

Pour la plupart des déploiements de Red Hat OpenShift, à l'aide d'un registre public comme "Quay.io" ou "DockerHub" répond à la plupart des besoins des clients. Cependant, il se peut qu'un client souhaite héberger ses propres images privées ou personnalisées.

Cette procédure décrit la création d'un registre d'images privées, sauvegardé par un volume persistant fourni par Astra Trident et NetApp ONTAP.

> Astra Control Center requiert un registre pour héberger les images dont les conteneurs Astra ont besoin. La section suivante décrit les étapes de configuration d'un registre privé sur un cluster Red Hat OpenShift et l'envoi des images requises pour prendre en charge l'installation d'Astra Control Center.

Création d'un registre d'images privé

1. Supprimez l'annotation par défaut de la classe de stockage par défaut actuelle et annoter la classe de stockage sauvegardée par Trident par défaut pour le cluster OpenShift.

```
[netapp-user@rhel7 ~]$ oc patch storageclass thin -p '{"metadata":
{"annotations": {"storageclass.kubernetes.io/is-default-class":
"false"}}}'
storageclass.storage.k8s.io/thin patched

[netapp-user@rhel7 ~]$ oc patch storageclass ocp-trident -p
'{"metadata": {"annotations": {"storageclass.kubernetes.io/is-default-
class": "true"}}}'
storageclass.storage.k8s.io/ocp-trident patched
```

2. Modifiez l'opérateur imageistry en saisissant les paramètres de stockage suivants dans le `spec` section.

```
[netapp-user@rhel7 ~]$ oc edit
configs.imageregistry.operator.openshift.io

storage:
  pvc:
    claim:
```

3. Entrez les paramètres suivants dans le `spec` Section permettant de créer une route OpenShift avec un nom d'hôte personnalisé. Enregistrer et quitter.

```
routes:
  - hostname: astra-registry.apps.ocp-vmw.cie.netapp.com
    name: netapp-astra-route
```

> La configuration de route ci-dessus est utilisée lorsque vous voulez un nom d'hôte personnalisé pour votre itinéraire. Si vous souhaitez qu'OpenShift crée une route avec un nom d'hôte par défaut, vous pouvez ajouter les paramètres suivants à l' `spec` section : `defaultRoute: true`.

Certificats TLS personnalisés

Lorsque vous utilisez un nom d'hôte personnalisé pour la route, il utilise par défaut la configuration TLS par défaut de l'opérateur OpenShift Ingress. Cependant, vous pouvez ajouter une configuration TLS personnalisée à la route. Pour ce faire, procédez comme suit.

a. Créez un secret avec les certificats TLS et la clé de la route.

```
[netapp-user@rhel7 ~]$ oc create secret tls astra-route-tls -n
openshift-image-registry –cert/home/admin/netapp-astra/tls.crt
--key=/home/admin/netapp-astra/tls.key
```

b. Modifiez l'opérateur imageistry et ajoutez les paramètres suivants à la `spec` section.

```
[netapp-user@rhel7 ~]$ oc edit
configs.imageregistry.operator.openshift.io

routes:
  - hostname: astra-registry.apps.ocp-vmw.cie.netapp.com
    name: netapp-astra-route
    secretName: astra-route-tls
```

4. Modifiez à nouveau l'opérateur imageistry et modifiez l'état de gestion de l'opérateur sur `Managed` état. Enregistrer et quitter.

```
oc edit configs.imageregistry/cluster

managementState: Managed
```

5. Si toutes les conditions préalables sont remplies, les ESV, les pods et les services sont créés pour le registre d'images privées. Dans quelques minutes, le registre devrait être mis en service.

```
[netapp-user@rhel7 ~]$oc get all -n openshift-image-registry

NAME                                                    READY   STATUS
RESTARTS   AGE
pod/cluster-image-registry-operator-74f6d954b6-rb7zr   1/1     Running
3          90d
pod/image-pruner-1627257600-f5cpj                       0/1     Completed
0          2d9h
pod/image-pruner-1627344000-swqx9                       0/1     Completed
0          33h
pod/image-pruner-1627430400-rv5nt                       0/1     Completed
0          9h
pod/image-registry-6758b547f-6pnj8                      1/1     Running
0          76m
pod/node-ca-bwb5r                                       1/1     Running
0          90d
pod/node-ca-f8w54                                       1/1     Running
0          90d
pod/node-ca-gjx7h                                       1/1     Running
0          90d
pod/node-ca-lcx4k                                       1/1     Running
0          33d
pod/node-ca-v7zmx                                       1/1     Running
0          7d21h
pod/node-ca-xpppp                                       1/1     Running
0          89d

NAME                                TYPE        CLUSTER-IP       EXTERNAL-
IP   PORT(S)     AGE
service/image-registry              ClusterIP   172.30.196.167   <none>
5000/TCP    15h
service/image-registry-operator     ClusterIP   None             <none>
60000/TCP   90d

NAME                       DESIRED   CURRENT   READY   UP-TO-DATE
AVAILABLE   NODE SELECTOR            AGE
daemonset.apps/node-ca   6         6         6       6            6
kubernetes.io/os=linux   90d

NAME                                                 READY   UP-TO-DATE
AVAILABLE   AGE
deployment.apps/cluster-image-registry-operator   1/1     1            1
90d
deployment.apps/image-registry                    1/1     1            1
15h
```

```
NAME                                                                DESIRED
CURRENT   READY   AGE
replicaset.apps/cluster-image-registry-operator-74f6d954b6   1         1
1       90d
replicaset.apps/image-registry-6758b547f                     1         1
1       76m
replicaset.apps/image-registry-78bfbd7f59                    0         0
0       15h
replicaset.apps/image-registry-7fcc8d6cc8                    0         0
0       80m
replicaset.apps/image-registry-864f88f5b                     0         0
0       15h
replicaset.apps/image-registry-cb47fffb                      0         0
0       10h

NAME                                 COMPLETIONS   DURATION   AGE
job.batch/image-pruner-1627257600   1/1           10s        2d9h
job.batch/image-pruner-1627344000   1/1           6s         33h
job.batch/image-pruner-1627430400   1/1           5s         9h

NAME                         SCHEDULE    SUSPEND   ACTIVE   LAST
SCHEDULE   AGE
cronjob.batch/image-pruner   0 0 * * *   False     0        9h
90d

NAME                                        HOST/PORT
PATH   SERVICES         PORT    TERMINATION   WILDCARD
route.route.openshift.io/public-routes   astra-registry.apps.ocp-
vmw.cie.netapp.com          image-registry   <all>   reencrypt     None
```

6. Si vous utilisez les certificats TLS par défaut pour la route de registre OpenShift de l'opérateur d'entrée, vous pouvez récupérer les certificats TLS à l'aide de la commande suivante.

```
[netapp-user@rhel7 ~]$ oc extract secret/router-ca --keys=tls.crt -n
openshift-ingress-operator
```

7. Pour permettre aux nœuds OpenShift d'accéder aux images et de les extraire du registre, ajoutez les certificats au client docker sur les nœuds OpenShift. Créez une configuration dans le `openshift-config` Espace de noms à l'aide des certificats TLS et le patch dans la configuration d'image du cluster pour que le certificat soit fiable.

```
[netapp-user@rhel7 ~]$ oc create configmap astra-ca -n openshift-config
--from-file=astra-registry.apps.ocp-vmw.cie.netapp.com=tls.crt

[netapp-user@rhel7 ~]$ oc patch image.config.openshift.io/cluster
--patch '{"spec":{"additionalTrustedCA":{"name":"astra-ca"}}}'
--type=merge
```

8. Le registre interne OpenShift est contrôlé par une authentification. Tous les utilisateurs OpenShift peuvent accéder au registre OpenShift, mais les opérations que l'utilisateur connecté peut exécuter dépendent des autorisations des utilisateurs.
   a. Pour permettre à un utilisateur ou à un groupe d'utilisateurs d'extraire des images du registre, le rôle du visualiseur de registre doit être affecté à l'utilisateur.

   ```
   [netapp-user@rhel7 ~]$ oc policy add-role-to-user registry-viewer
   ocp-user

   [netapp-user@rhel7 ~]$ oc policy add-role-to-group registry-viewer
   ocp-user-group
   ```

   b. Pour permettre à un utilisateur ou à un groupe d'utilisateurs d'écrire ou de diffuser des images, le rôle de l'éditeur de registre doit être affecté.

   ```
   [netapp-user@rhel7 ~]$ oc policy add-role-to-user registry-editor
   ocp-user

   [netapp-user@rhel7 ~]$ oc policy add-role-to-group registry-editor
   ocp-user-group
   ```

9. Pour que les nœuds OpenShift puissent accéder au registre et envoyer ou extraire les images, vous devez configurer un secret Pull.

   ```
   [netapp-user@rhel7 ~]$ oc create secret docker-registry astra-registry-
   credentials --docker-server=astra-registry.apps.ocp-vmw.cie.netapp.com
   --docker-username=ocp-user --docker-password=password
   ```

10. Ce secret Pull peut ensuite être corrigé aux comptes de service ou être référencé dans la définition de pod correspondante.
    a. Pour le corriger aux comptes de service, exécutez la commande suivante.

    ```
    [netapp-user@rhel7 ~]$ oc secrets link <service_account_name> astra-
    registry-credentials --for=pull
    ```

b. Pour référencer le secret Pull dans la définition du pod, ajoutez le paramètre suivant à l' `spec` section.

```
imagePullSecrets:
  - name: astra-registry-credentials
```

11. Pour pousser ou extraire une image des postes de travail en dehors du nœud OpenShift, procédez comme suit.

a. Ajoutez les certificats TLS au client docker.

```
[netapp-user@rhel7 ~]$ sudo mkdir /etc/docker/certs.d/astra-
registry.apps.ocp-vmw.cie.netapp.com

[netapp-user@rhel7 ~]$ sudo cp /path/to/tls.crt
/etc/docker/certs.d/astra-registry.apps.ocp-vmw.cie.netapp.com
```

b. Connectez-vous à OpenShift à l'aide de la commande oc login.

```
[netapp-user@rhel7 ~]$ oc login --token=sha256~D49SpB_lesSrJYwrM0LIO
-VRcjWHu0a27vKa0 --server=https://api.ocp-vmw.cie.netapp.com:6443
```

c. Connectez-vous au registre à l'aide des informations d'identification de l'utilisateur OpenShift avec la commande podman/docker.

podman

```
[netapp-user@rhel7 ~]$ podman login astra-registry.apps.ocp-
vmw.cie.netapp.com -u kubeadmin -p $(oc whoami -t) --tls
-verify=false
```

+ REMARQUE : si vous utilisez `kubeadmin` l'utilisateur doit se connecter au registre privé, puis utiliser un jeton au lieu du mot de passe.

docker

```
[netapp-user@rhel7 ~]$ docker login astra-registry.apps.ocp-
vmw.cie.netapp.com -u kubeadmin -p $(oc whoami -t)
```

+ REMARQUE : si vous utilisez `kubeadmin` l'utilisateur doit se connecter au registre privé, puis utiliser un jeton au lieu du mot de passe.

d. Pousser ou extraire les images.

podman

```
[netapp-user@rhel7 ~]$ podman push astra-registry.apps.ocp-
vmw.cie.netapp.com/netapp-astra/vault-controller:latest
[netapp-user@rhel7 ~]$ podman pull astra-registry.apps.ocp-
vmw.cie.netapp.com/netapp-astra/vault-controller:latest
```

docker

```
[netapp-user@rhel7 ~]$ docker push astra-registry.apps.ocp-
vmw.cie.netapp.com/netapp-astra/vault-controller:latest
[netapp-user@rhel7 ~]$ docker pull astra-registry.apps.ocp-
vmw.cie.netapp.com/netapp-astra/vault-controller:latest
```

Validation et utilisations de la solution : Red Hat OpenShift avec NetApp

Les exemples présentés sur cette page sont les validations et les utilisations de Red Hat OpenShift avec NetApp.

- "Déployez un pipeline ci/CD Jenkins avec le stockage persistant"
- "Configurez la colocation sur Red Hat OpenShift avec NetApp"
- "Red Hat OpenShift Virtualization avec NetApp ONTAP"
- "Solution NetApp de gestion avancée des clusters pour Kubernetes sur Red Hat OpenShift"

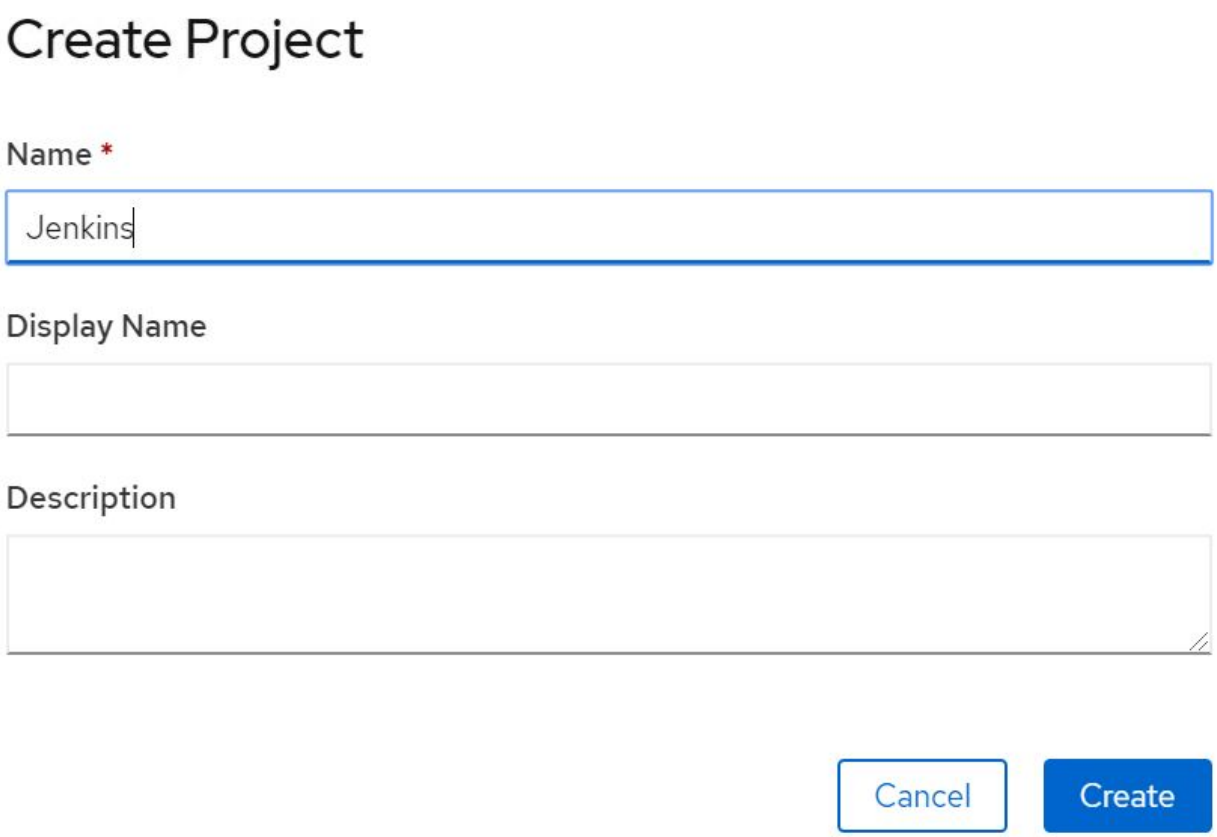
Déployez un pipeline ci/CD Jenkins avec le stockage persistant : Red Hat OpenShift avec NetApp

Cette section explique comment déployer un pipeline d'intégration/livraison continues ou de déploiement avec Jenkins pour valider le fonctionnement de la solution.

Créez les ressources requises pour le déploiement de Jenkins

Pour créer les ressources nécessaires au déploiement de l'application Jenkins, procédez comme suit :

1. Créez un nouveau projet appelé Jenkins.



2. Dans cet exemple, nous avons déployé Jenkins avec du stockage persistant. Pour prendre en charge la construction Jenkins, créez le PVC. Accédez à stockage > demandes de volume persistant et cliquez sur Créer une demande de volume persistant. Sélectionnez la classe de stockage créée, vérifiez que le nom de la demande de volume persistant est jenkins, sélectionnez la taille et le mode d'accès appropriés, puis cliquez sur Créer.

Project: jenkins

Create Persistent Volume Claim

Edit YAML

Storage Class

SC basic

Storage class for the new claim.

Persistent Volume Claim Name *

jenkins

A unique name for the storage claim within the project.

Access Mode *

Single User (RWO) Shared Access (RWX) Read Only (ROX)

Permissions to the mounted drive.

Size *

100 GiB

Desired storage capacity.

Use label selectors to request storage

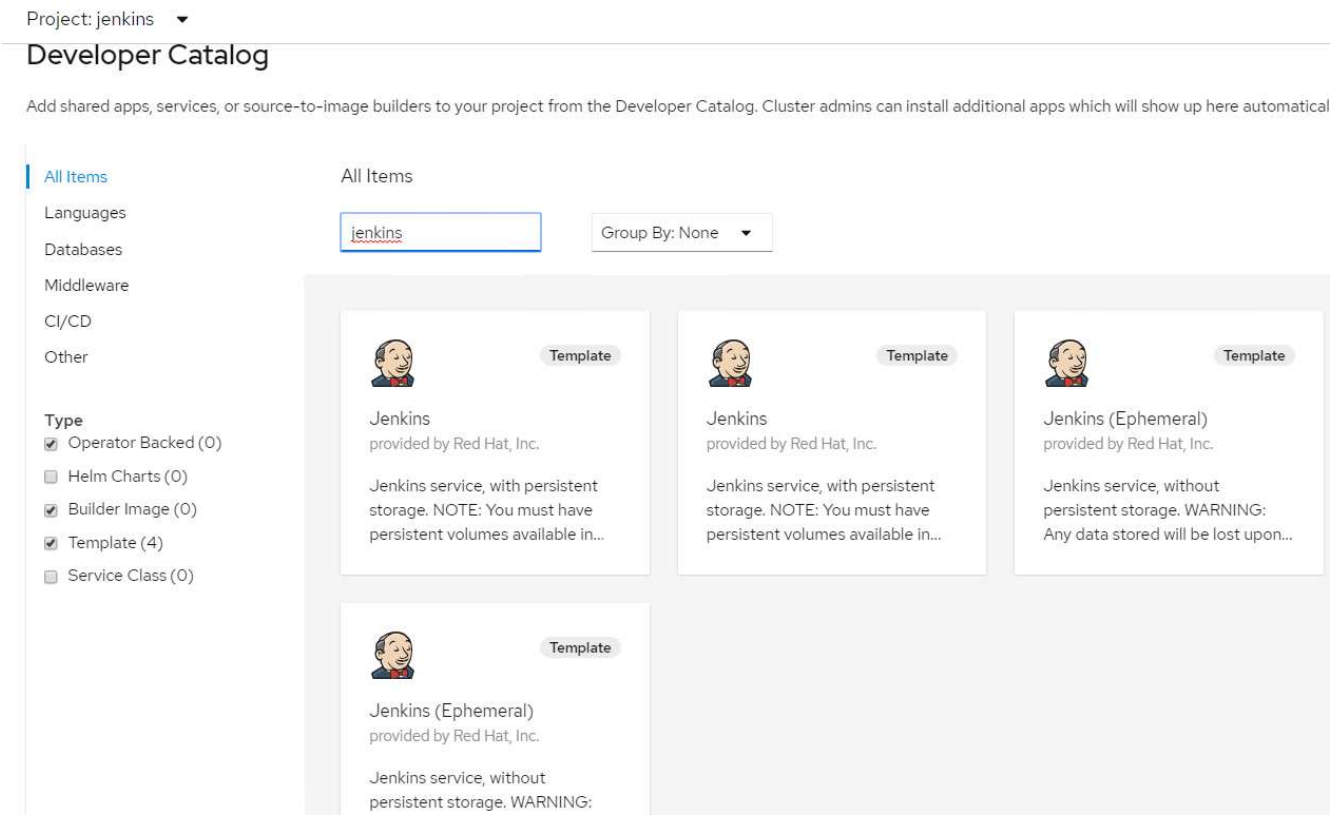
Use label selectors to define how storage is created.

Create Cancel

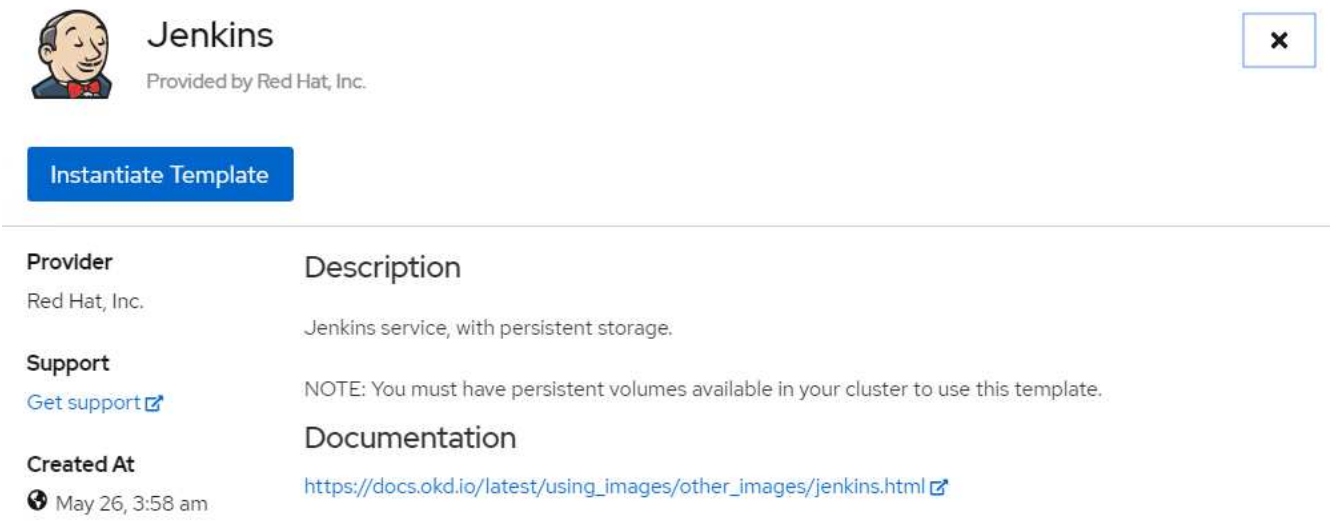
Déployez Jenkins avec le stockage persistant

Pour déployer Jenkins avec le stockage persistant, procédez comme suit :

1. Dans le coin supérieur gauche, modifiez le rôle de Administrateur à Développeur. Cliquez sur +Ajouter et sélectionnez à partir du catalogue. Dans la barre filtre par mot-clé, recherchez jenkins. Sélectionnez le service Jenkins avec le stockage persistant.



2. Cliquez sur `Instantiate Template`.



3. Par défaut, les détails de l'application Jenkins sont renseignés. En fonction de vos besoins, modifiez les paramètres et cliquez sur Créer. Ce processus crée toutes les ressources nécessaires pour prendre en

charge Jenkins sur OpenShift.

Instantiate Template

Namespace *

PR jenkins

Jenkins Service Name

jenkins

The name of the OpenShift Service exposed for the Jenkins container.

Jenkins JNLP Service Name

jenkins-jnlp

The name of the service used for master/slave communication.

Enable OAuth in Jenkins

true

Whether to enable OAuth OpenShift integration. If false, the static account 'admin' will be initialized with the password 'password'.

Memory Limit

Gi

Maximum amount of memory the container can use.

Volume Capacity *

50Gi

Volume space available for data, e.g. 512Mi, 2Gi.

Jenkins ImageStream Namespace

openshift

The OpenShift Namespace where the Jenkins ImageStream resides.

Disable memory intensive administrative monitors

false

Whether to perform memory intensive, possibly slow, synchronization with the Jenkins Update Center on start. If true, the Jenkins core update monitor and site warnings monitor are disabled.

Jenkins ImageStreamTag

jenkins:2

Name of the ImageStreamTag to be used for the Jenkins image.

Fatal Error Log File

false

When a fatal error occurs, an error log is created with information and the state obtained at the time of the fatal error.

Allows use of Jenkins Update Center repository with invalid SSL certificate

false

Whether to allow use of a Jenkins Update Center that uses invalid certificate (self-signed, unknown CA). If any value other than 'false', certificate check is bypassed. By default, certificate check is enforced.

Create Cancel

Jenkins

INSTANT-APP JENKINS

View documentation Get support

Jenkins service, with persistent storage.

NOTE: You must have persistent volumes available in your cluster to use this template.

The following resources will be created:

- DeploymentConfig
- PersistentVolumeClaim
- RoleBinding
- Route
- Service
- ServiceAccount

4. Les modules Jenkins prennent environ 10 à 12 minutes pour entrer en état « prêt ».

Project: jenkins

Pods

Create Pod

Filter by name...

1 Running | 0 Pending | 0 Terminating | 0 CrashLoopBackOff | 1 Completed | 0 Failed | 0 Unknown

Select all filters

1 of 2 Items

Name	Namespace	Status	Ready	Owner	Memory	CPU
P jenkins-1-c77n9	NS jenkins	Running	1/1	RC jenkins-1	-	0.004 cores

5. Une fois les pods instanciés, accédez à réseau > routes. Pour ouvrir la page Web Jenkins, cliquez sur l'URL fournie pour la route jenkins.

Project: jenkins

Routes

Create Route

Filter by name...

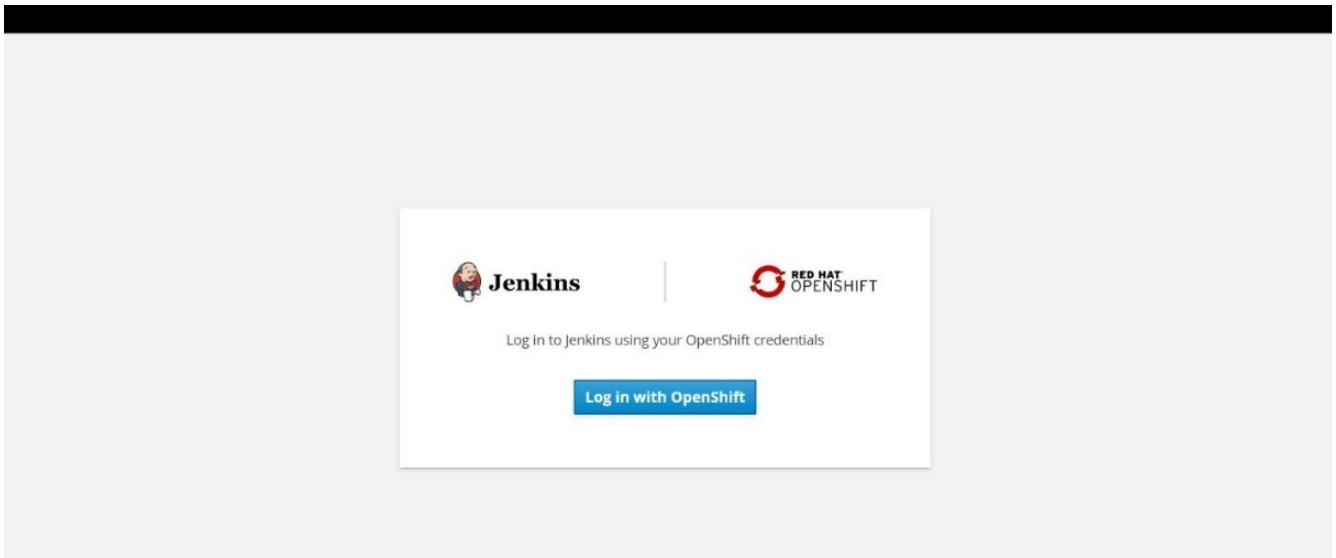
1 Accepted | 0 Rejected | 0 Pending

Select all filters

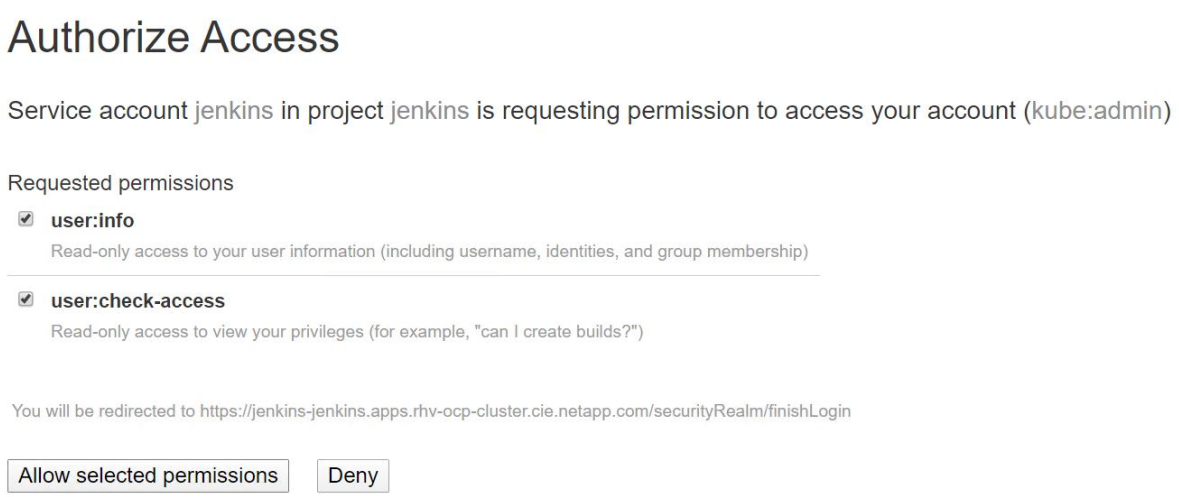
1 Item

Name	Namespace	Status	Location	Service
RT jenkins	NS jenkins	Accepted	https://jenkins-jenkins.apps.rhv-ocp-cluster.cie.netapp.com	S jenkins

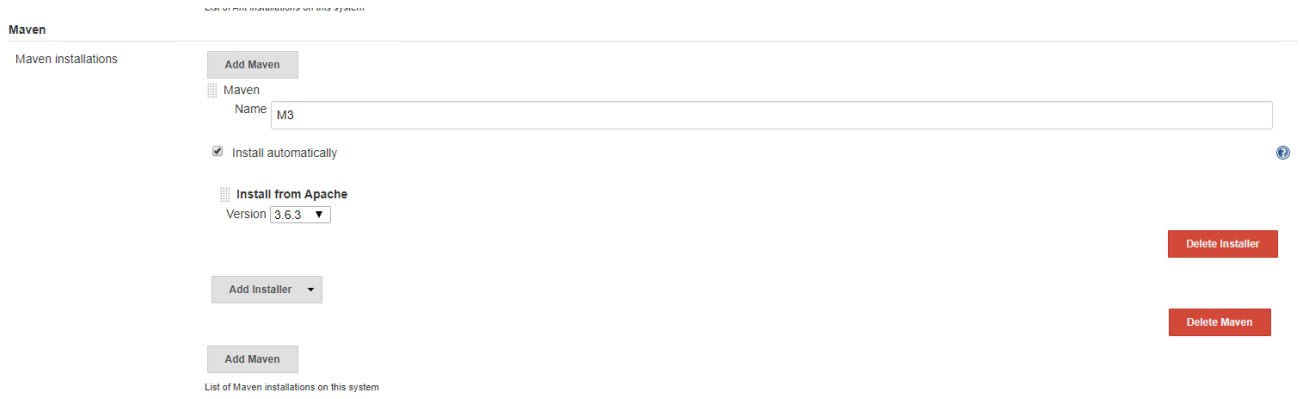
6. OpenShift OAuth a été utilisé lors de la création de l'application Jenkins, cliquez sur « se connecter avec OpenShift ».



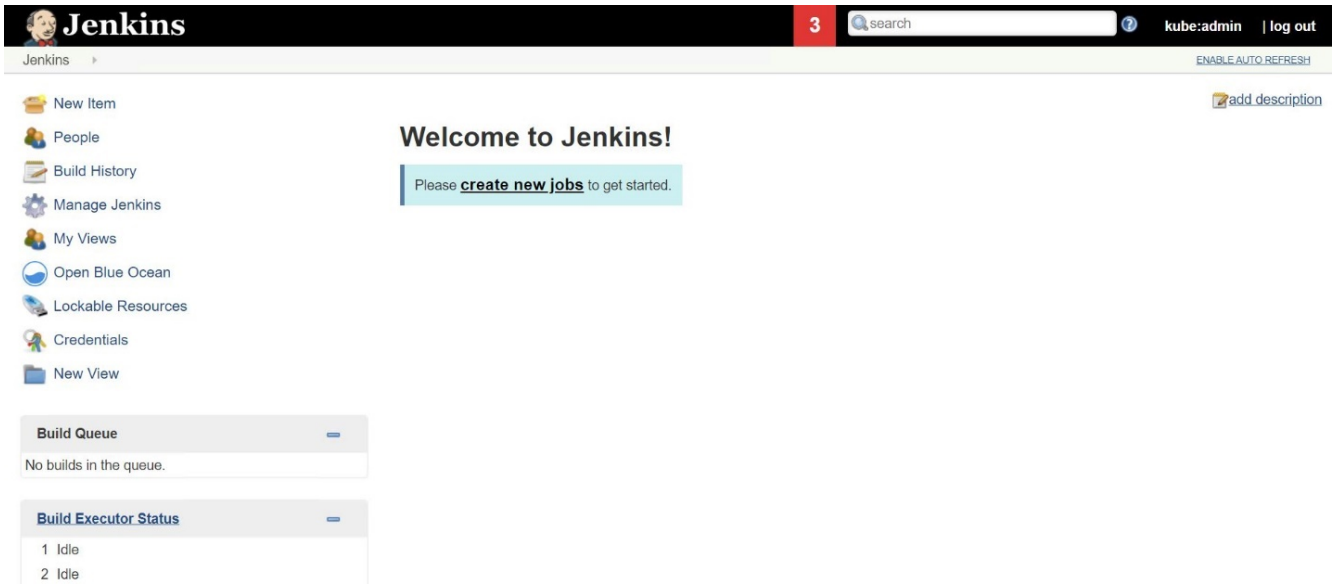
7. Autoriser le compte de service Jenkins à accéder aux utilisateurs OpenShift.



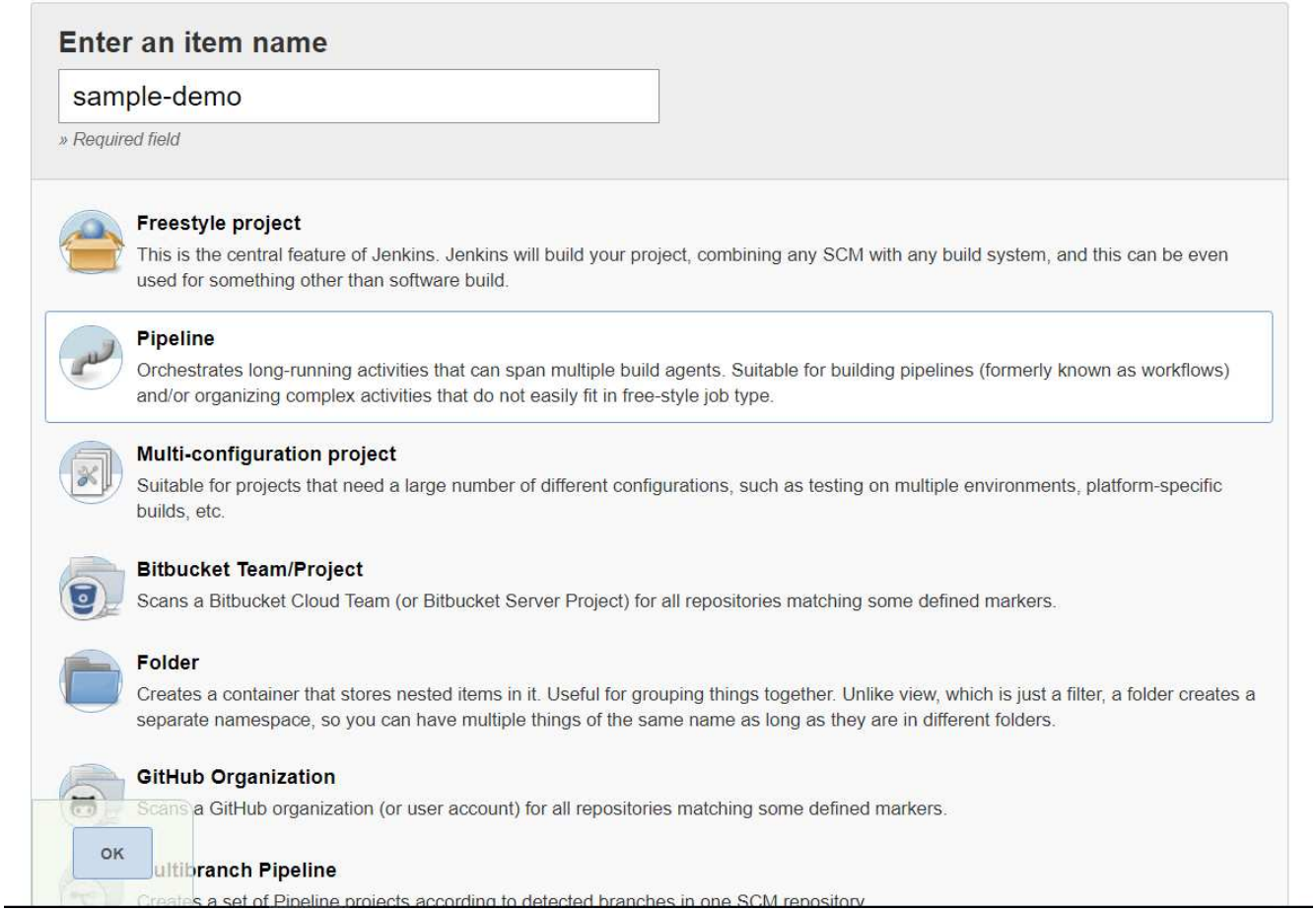
8. La page d’accueil de Jenkins s’affiche. Parce que nous utilisons une construction Maven, terminez d’abord l’installation Maven. Accédez à Manage Jenkins > Global Tool Configuration, puis, dans le sous-titre Maven, cliquez sur Add Maven. Entrez le nom de votre choix et assurez-vous que l’option installer automatiquement est sélectionnée. Cliquez sur Enregistrer.



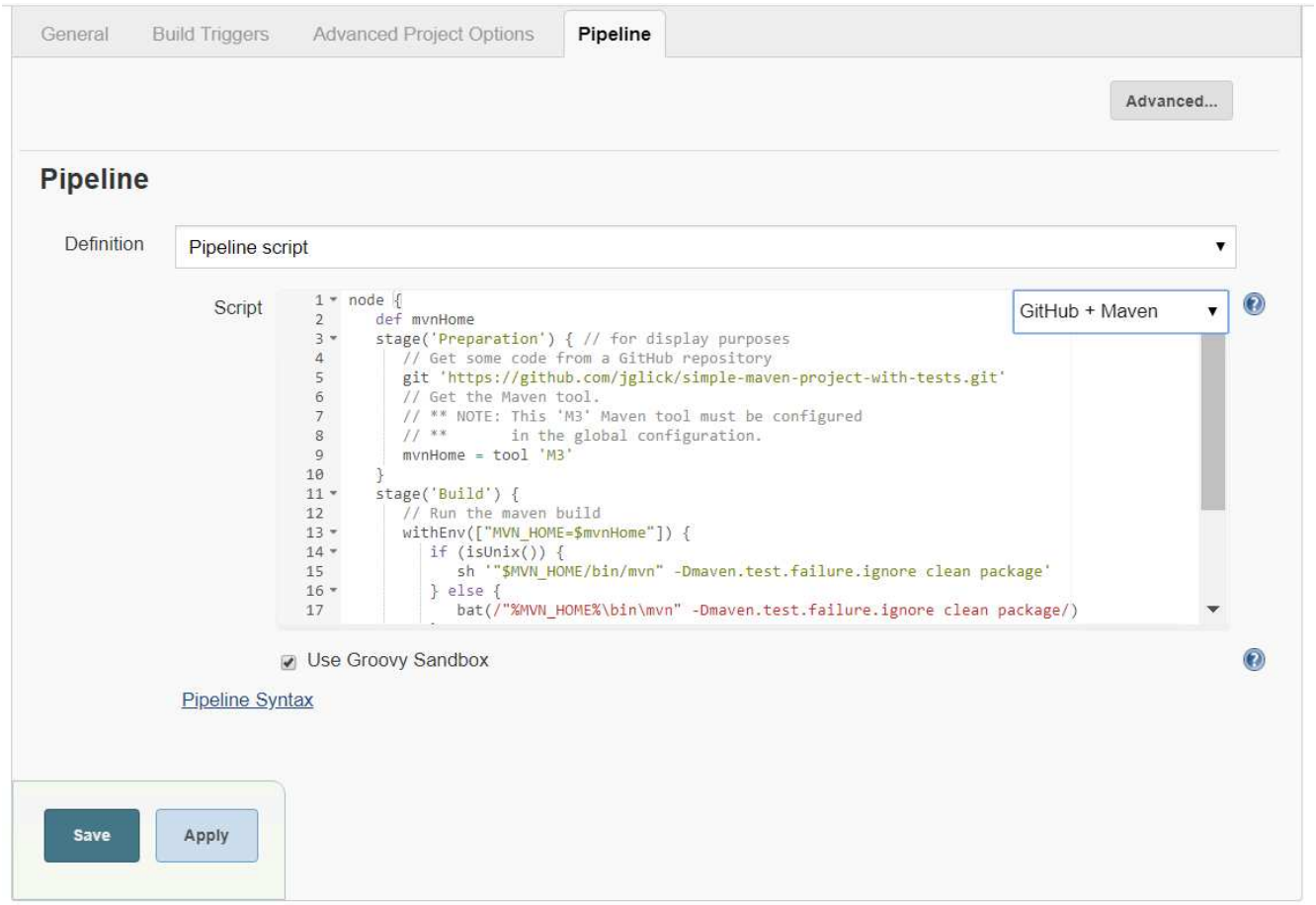
9. Vous pouvez désormais créer un pipeline pour démontrer le workflow ci/CD. Sur la page d'accueil, cliquez sur Créer de nouveaux travaux ou nouvel élément dans le menu de gauche.



10. Sur la page Créer un élément, entrez le nom de votre choix, sélectionnez Pipeline, puis cliquez sur OK.



11. Sélectionnez l'onglet Pipeline. Dans le menu déroulant essayer un pipeline d'échantillon, sélectionnez Github + Maven. Le code est automatiquement renseigné. Cliquez sur Enregistrer.



12. Cliquez sur Créer maintenant pour déclencher le développement tout au long de la phase de préparation, de création et de test. Il peut prendre plusieurs minutes pour terminer l'ensemble du processus de construction et afficher les résultats de la construction.

13. Chaque fois que du code change, le pipeline peut être reconstruit pour corriger la nouvelle version du logiciel permettant l'intégration et la livraison continues. Cliquez sur modifications récentes pour suivre les modifications apportées à la version précédente.

Configurer la colocation sur Red Hat OpenShift avec NetApp ONTAP

Configuration d'une colocation sur Red Hat OpenShift avec NetApp

De nombreuses entreprises qui exécutent plusieurs applications ou charges de travail sur des conteneurs ont tendance à déployer un cluster Red Hat OpenShift par application ou par workload. Ils peuvent ainsi mettre en œuvre une isolation stricte pour l'application ou la charge de travail, optimiser les performances et réduire les vulnérabilités de sécurité. Toutefois, le déploiement d'un cluster Red Hat OpenShift distinct pour chaque application présente ses propres problèmes. Cette solution augmente les frais d'exploitation liés à la surveillance et à la gestion seule de chaque cluster, ce qui augmente les coûts du fait de ressources dédiées pour différentes applications et entrave l'évolutivité efficace.

Pour résoudre ces problèmes, il est possible d'exécuter toutes les applications ou charges de travail dans un seul cluster Red Hat OpenShift. Cependant, dans une telle architecture, l'isolement des ressources et les vulnérabilités liées à la sécurité des applications constituent l'un des défis majeurs. Toute vulnérabilité de sécurité dans une charge de travail pourrait naturellement se répandre sur une autre charge de travail, augmentant ainsi la zone d'impact. En outre, une application peut avoir une incidence soudaine et non contrôlée sur les performances d'une autre application, car il n'existe pas de stratégie d'allocation des ressources par défaut.

Les entreprises recherchent donc des solutions qui offrent les meilleures des deux mondes, par exemple, en leur permettant d'exécuter toutes leurs charges de travail dans un cluster unique, tout en offrant les avantages

d’un cluster dédié pour chaque charge de travail.

L’une de ces solutions est utile : configurer la colocation sur Red Hat OpenShift. La colocation est une architecture qui permet à plusieurs locataires de coexister sur un même cluster avec une isolation appropriée des ressources, de la sécurité, etc. Dans ce contexte, un locataire peut être considéré comme un sous-ensemble des ressources du cluster qui sont configurées pour être utilisées par un groupe d’utilisateurs particulier à des fins exclusives. La configuration d’une colocation sur un cluster Red Hat OpenShift offre les avantages suivants :

- Réduction des dépenses d’investissement et d’exploitation en permettant le partage des ressources du cluster
- Réduisez les frais d’exploitation et de gestion
- Sécurisation des charges de travail contre toute contamination croisée des failles de sécurité
- Protection des charges de travail contre la dégradation inattendue des performances en raison des conflits des ressources

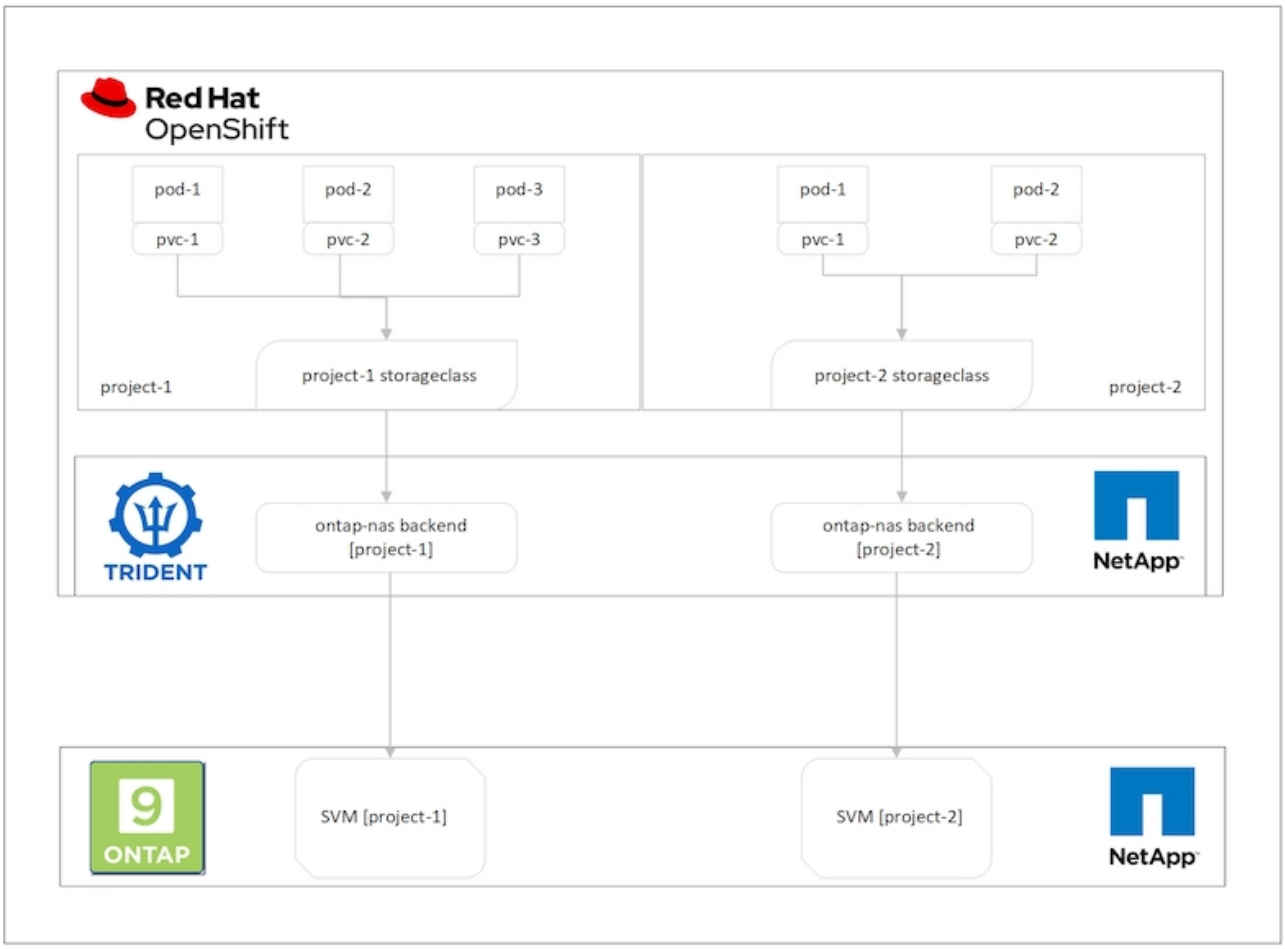
Pour un cluster OpenShift mutualisé entièrement réalisé, les quotas et les restrictions doivent être configurés pour les ressources de cluster appartenant à différents compartiments de ressources : calcul, stockage, réseau, sécurité, etc. Bien que nous aborderons certains aspects de toutes les ressources de cette solution, Nous mettons l’accent sur les bonnes pratiques d’isolation et de sécurisation des données servies ou consommées par plusieurs charges de travail sur le même cluster Red Hat OpenShift en configurant la colocation sur des ressources de stockage allouées de façon dynamique par Astra Trident et sauvegardé par NetApp ONTAP.

Architecture

Bien que Red Hat OpenShift et Astra Trident avec NetApp ONTAP ne assurent pas l’isolation des charges de travail par défaut, ils offrent un large éventail de fonctionnalités qui peuvent être utilisées pour configurer la colocation. Pour mieux comprendre comment concevoir une solution mutualisée sur un cluster Red Hat OpenShift avec Astra Trident basée sur NetApp ONTAP, nous examinons un exemple d’exigences et nous présente la configuration qui l’entoure.

Supposons qu’une entreprise exécute deux de ses charges de travail sur un cluster Red Hat OpenShift dans le cadre de deux projets sur lesquels deux équipes différentes travaillent. Les données de ces workloads résident sur des demandes de volume persistant qui sont provisionnées dynamiquement par Astra Trident sur un back-end NAS NetApp ONTAP. L’entreprise doit concevoir une solution mutualisée pour ces deux charges de travail et isoler les ressources utilisées pour ces projets afin de garantir la sécurité et la performance nécessaires. Elle est axée sur les données qui servent ces applications.

La figure suivante décrit la solution mutualisée sur un cluster Red Hat OpenShift avec Astra Trident et NetApp ONTAP.



Exigences technologiques

1. Cluster de stockage NetApp ONTAP
2. Cluster Red Hat OpenShift
3. Astra Trident

Ressources Red Hat OpenShift – Cluster

Du point de vue du cluster Red Hat OpenShift, la ressource de premier niveau à commencer est le projet. Un projet OpenShift peut être considéré comme une ressource de cluster qui divise l'ensemble du cluster OpenShift en plusieurs clusters virtuels. Ainsi, l'isolation au niveau du projet fournit une base pour la configuration de la colocation.

Ensuite, vous devez configurer RBAC dans le cluster. La meilleure pratique consiste à configurer tous les développeurs sur un seul projet ou charge de travail dans un seul groupe d'utilisateurs du fournisseur d'identités. Red Hat OpenShift permet l'intégration IDP et la synchronisation des groupes d'utilisateurs, ce qui permet d'importer les utilisateurs et les groupes du PDI dans le cluster. Les administrateurs du cluster peuvent ainsi isoler l'accès aux ressources du cluster dédiées à un projet à un ou plusieurs groupes d'utilisateurs travaillant sur ce projet, ce qui limite l'accès non autorisé aux ressources du cluster. Pour en savoir plus sur l'intégration IDP avec Red Hat OpenShift, consultez la documentation "ici".

NetApp ONTAP

Il est important d'isoler le service de stockage partagé en tant que fournisseur de stockage persistant pour un cluster Red Hat OpenShift afin de vérifier que les volumes créés sur le stockage pour chaque projet apparaissent aux hôtes comme s'ils sont créés sur un stockage distinct. Pour ce faire, créez autant de SVM (Storage Virtual machines) sur NetApp ONTAP que des projets ou des charges de travail et dédier chaque SVM à une charge de travail.

Astra Trident

Une fois que vous avez des SVM différents pour les projets créés sur NetApp ONTAP, vous devez mapper chaque SVM sur un back-end Trident différent. La configuration back-end de Trident entraîne l'allocation du stockage persistant aux ressources de cluster OpenShift, et elle requiert le mappage des détails de la SVM sur. Il doit s'agir du pilote de protocole pour le back-end au minimum. Vous pouvez également définir la manière dont les volumes sont provisionnés sur le stockage et définir des limites pour la taille des volumes ou l'utilisation des agrégats, etc. Vous trouverez des informations détaillées sur la définition des systèmes back-end Trident "ici".

Red Hat OpenShift – ressources de stockage

Une fois les systèmes back-end Trident configurés, l'étape suivante consiste à configurer les classes de stockage. Configurez autant de classes de stockage que les systèmes back-end, en donnant à chaque classe de stockage l'accès pour lancer des volumes sur un seul système back-end. Nous pouvons mapper la classe de stockage sur un back-end Trident en utilisant le paramètre storagePools lors de la définition de la classe de stockage. Les détails de la définition d'une classe de stockage sont disponibles "ici". Il existe donc un mappage un-à-un de StorageClass vers le backend Trident qui pointe vers un SVM. Ainsi, toutes les demandes de stockage traitées par la classe de stockage allouée à ce projet sont servies par la SVM dédiée à ce projet uniquement.

Comme les classes de stockage ne namesles ressources qui ne sont pas adaptées, comment pouvons-nous nous assurer que les déclarations de stockage présentées dans la classe d'un projet par des pods dans un autre espace de noms ou dans des projets sont rejetées ? La réponse est d'utiliser ResourceQuotas. ResourceQuotas sont des objets qui contrôlent l'utilisation totale des ressources par projet. Elle peut limiter le nombre ainsi que la quantité totale de ressources pouvant être consommées par des objets dans le projet. Presque toutes les ressources d'un projet peuvent être limitées à l'aide de ResourceQuotas et l'utilisation efficace de cette solution peut aider les entreprises à réduire les coûts et les pannes dus au sur-provisionnement ou à la sur-consommation des ressources. Reportez-vous à la documentation "ici" pour en savoir plus.

Pour ce cas d'utilisation, nous devons limiter les demandes de stockage provenant de classes de stockage qui ne sont pas dédiées à leur projet dans un projet particulier. Il nous faut donc limiter les demandes de volume persistant pour d'autres classes de stockage par paramètre `<storage-class-name>.storageclass.storage.k8s.io/persistentvolumeclaims` à 0. En outre, un administrateur de cluster doit s'assurer que les développeurs d'un projet ne doivent pas avoir accès pour modifier les ResourceQuotas.

Configuration

N'importe quelle solution mutualisée permet à aucun utilisateur d'accéder à davantage de ressources du cluster que nécessaire. Ainsi, l'ensemble des ressources à configurer dans le cadre de la configuration de colocation est divisé entre l'administrateur cluster, l'administrateur stockage et les développeurs travaillant sur chaque projet.

Le tableau suivant présente les différentes tâches à effectuer par différents utilisateurs :

Rôle	Tâches
Cluster-admin	Créez des projets pour différentes applications ou charges de travail
	Créez ClusterRoles et roles pour Storage-admin
	Créez des rôles et des roleliaisons pour les développeurs qui assignaient un accès à des projets spécifiques
	[Facultatif] configurez les projets pour planifier des pods sur des nœuds spécifiques
Storage-admin	Créez des SVM sur NetApp ONTAP
	Création des systèmes back-end Trident
	Créez des classes de stockage
	Créer des devis de ressources de stockage
Développeurs	Valider l'accès pour créer ou corriger des demandes de volume persistant ou des pods dans le projet affecté
	Valider l'accès pour créer ou corriger des demandes de volume persistant ou des pods dans un autre projet
	Validez l'accès pour afficher ou modifier des projets, des ResourceQuotas et des classes de stockage

Configuration

Prérequis

- Cluster NetApp ONTAP
- Cluster Red Hat OpenShift
- Trident est installé sur le cluster
- Station de travail Admin avec les outils tridentctl et oc installés et ajoutés à $PATH
- Accès administrateur à ONTAP
- L'accès cluster-admin au cluster OpenShift
- Le cluster est intégré avec Identity Provider
- Le fournisseur d'identités est configuré pour distinguer efficacement les utilisateurs de différentes équipes

Configuration : tâches d'administration du cluster

Les tâches suivantes sont réalisées par Red Hat OpenShift Cluster-admin :

1. Connectez-vous au cluster Red Hat OpenShift en tant qu'administrateur cluster.
2. Créer deux projets correspondant à différents projets.

```
oc create namespace project-1
oc create namespace project-2
```

3. Créer le rôle de développeur du projet-1.

```
cat << EOF | oc create -f -
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  namespace: project-1
  name: developer-project-1
rules:
  - verbs:
      - '*'
    apiGroups:
      - apps
      - batch
      - autoscaling
      - extensions
      - networking.k8s.io
      - policy
      - apps.openshift.io
      - build.openshift.io
      - image.openshift.io
      - ingress.operator.openshift.io
      - route.openshift.io
      - snapshot.storage.k8s.io
      - template.openshift.io
    resources:
      - '*'
  - verbs:
      - '*'
    apiGroups:
      - ''
    resources:
      - bindings
      - configmaps
      - endpoints
      - events
      - persistentvolumeclaims
      - pods
      - pods/log
      - pods/attach
      - podtemplates
      - replicationcontrollers
```

```
      - services
      - limitranges
      - namespaces
      - componentstatuses
      - nodes
  - verbs:
      - '*'
    apiGroups:
      - trident.netapp.io
    resources:
      - tridentsnapshots
EOF
```

La définition de rôle fournie dans cette section n'est qu'un exemple. Les rôles de développeur doivent être définis en fonction des exigences de l'utilisateur final.

1. De la même façon, créez des rôles de développement pour Project-2.
2. Toutes les ressources de stockage OpenShift et NetApp sont généralement gérées par un administrateur du stockage. L'accès pour les administrateurs du stockage est contrôlé par le rôle de l'opérateur trident créé lors de l'installation de Trident. En outre, l'administrateur du stockage nécessite également l'accès à ResourceQuotas pour contrôler la consommation du stockage.
3. Créez un rôle pour gérer ResourceQuotas dans tous les projets du cluster afin de le relier à l'administrateur de stockage.

```
cat << EOF | oc create -f -
kind: ClusterRole
apiVersion: rbac.authorization.k8s.io/v1
metadata:
  name: resource-quotas-role
rules:
  - verbs:
      - '*'
    apiGroups:
      - ''
    resources:
      - resourcequotas
  - verbs:
      - '*'
    apiGroups:
      - quota.openshift.io
    resources:
      - '*'
EOF
```

4. Assurez-vous que le cluster est intégré au fournisseur d'identité de l'entreprise et que les groupes d'utilisateurs sont synchronisés avec les groupes de clusters. L'exemple suivant montre que le fournisseur d'identités a été intégré au cluster et synchronisé avec les groupes d'utilisateurs.

```
$ oc get groups
NAME                          USERS
ocp-netapp-storage-admins     ocp-netapp-storage-admin
ocp-project-1                 ocp-project-1-user
ocp-project-2                 ocp-project-2-user
```

1. Configurer les liaisons ClusterRoleBindages pour les administrateurs de stockage.

```
cat << EOF | oc create -f -
kind: ClusterRoleBinding
apiVersion: rbac.authorization.k8s.io/v1
metadata:
  name: netapp-storage-admin-trident-operator
subjects:
  - kind: Group
    apiGroup: rbac.authorization.k8s.io
    name: ocp-netapp-storage-admins
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: ClusterRole
  name: trident-operator
---
kind: ClusterRoleBinding
apiVersion: rbac.authorization.k8s.io/v1
metadata:
  name: netapp-storage-admin-resource-quotas-cr
subjects:
  - kind: Group
    apiGroup: rbac.authorization.k8s.io
    name: ocp-netapp-storage-admins
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: ClusterRole
  name: resource-quotas-role
EOF
```

Pour les administrateurs du stockage, deux rôles doivent être liés : trident-Operator et Resource-quotas.

1. Créer des liaisons de type rôle pour les développeurs liant le rôle développeur-projet-1 au groupe correspondant (ocp-project-1) dans Project-1.

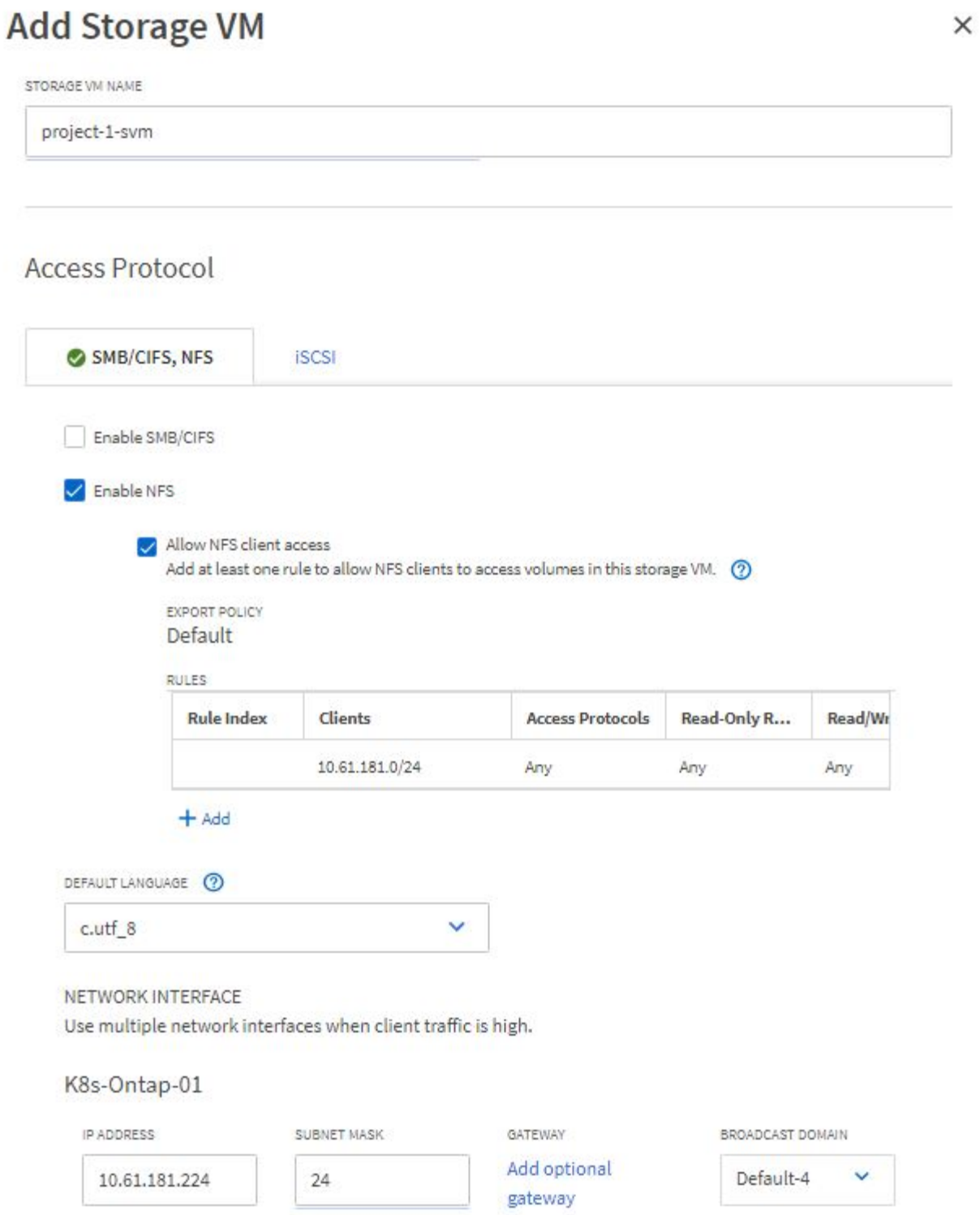
```
cat << EOF | oc create -f -
kind: RoleBinding
apiVersion: rbac.authorization.k8s.io/v1
metadata:
  name: project-1-developer
  namespace: project-1
subjects:
  - kind: Group
    apiGroup: rbac.authorization.k8s.io
    name: ocp-project-1
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: Role
  name: developer-project-1
EOF
```

2. De même, créez des liaisons de type rôle pour les développeurs qui lient les rôles de développeur au groupe d'utilisateurs correspondant dans Project-2.

Configuration : tâches d'administration du stockage

Les ressources suivantes doivent être configurées par un administrateur de stockage :

1. Connectez-vous au cluster NetApp ONTAP en tant qu'administrateur.
2. Accédez à Storage > Storage VM et cliquez sur Add. Créer deux SVM, un pour le projet-1 et l'autre pour le projet-2, en fournissant les détails requis. Créer également un compte vsadmin pour gérer le SVM et ses ressources



1. Connectez-vous au cluster Red Hat OpenShift en tant qu'administrateur du stockage.
2. Créer le backend pour projet-1 et le mapper au SVM dédié au projet NetApp recommande d'utiliser le compte vsadmin du SVM afin de connecter le backend au SVM au lieu d'utiliser l'administrateur du cluster ONTAP

```
cat << EOF | tridentctl -n trident create backend -f
{
    "version": 1,
    "storageDriverName": "ontap-nas",
    "backendName": "nfs_project_1",
    "managementLIF": "172.21.224.210",
    "dataLIF": "10.61.181.224",
    "svm": "project-1-svm",
    "username": "vsadmin",
    "password": "NetApp123"
}
EOF
```

Nous utilisons le pilote ontap-nas dans cet exemple. Utilisez le pilote approprié lors de la création du back-end en fonction du cas d'utilisation.

Nous partons du principe que Trident est installé dans le projet trident.

1. Créer de la même manière le back-end Trident pour le projet-2 et le mapper sur le SVM dédié au projet-2.
2. Créez ensuite les classes de stockage. Créez la classe de stockage pour Project-1 et configurez-la pour utiliser les pools de stockage du back-end dédié au projet-1 en définissant le paramètre storagePools.

```
cat << EOF | oc create -f -
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: project-1-sc
provisioner: csi.trident.netapp.io
parameters:
  backendType: ontap-nas
  storagePools: "nfs_project_1:.*"
EOF
```

3. De même, créez une classe de stockage pour Project-2 et configurez-la pour utiliser les pools de stockage du système back-end dédié au projet-2.
4. Créer un Resourcequota pour limiter les ressources dans le projet-1 demandant le stockage de storageclasses dédiés à d'autres projets.

```
cat << EOF | oc create -f -
kind: ResourceQuota
apiVersion: v1
metadata:
  name: project-1-sc-rq
  namespace: project-1
spec:
  hard:
    project-2-sc.storageclass.storage.k8s.io/persistentvolumeclaims: 0
EOF
```

5. De même, créez un Resourcequota pour limiter les ressources du projet 2 demandant du stockage de storageclasses dédiés à d'autres projets.

Validation

Pour valider l'architecture mutualisée configurée lors des étapes précédentes, procédez comme suit :

Valider l'accès pour créer des demandes de volume persistant ou des pods dans le projet attribué

1. Connectez-vous en tant qu'utilisateur ocp-project-1, développeur dans Project-1.
2. Vérifiez l'accès pour créer un nouveau projet.

```
oc create ns sub-project-1
```

3. Créez un PVC dans Project-1 en utilisant le storageclass affecté au projet-1.

```
cat << EOF | oc create -f -
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: test-pvc-project-1
  namespace: project-1
  annotations:
    trident.netapp.io/reclaimPolicy: Retain
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 1Gi
  storageClassName: project-1-sc
EOF
```

4. Vérifiez le volume persistant associé à la demande de volume persistant.

```
oc get pv
```

5. Vérifiez que le volume persistant et son volume sont créés dans un SVM dédié à Project-1 sur NetApp ONTAP.

```
volume show -vserver project-1-svm
```

6. Créez un pod dans Project-1 et montez le PVC créé à l'étape précédente.

```
cat << EOF | oc create -f -
kind: Pod
apiVersion: v1
metadata:
  name: test-pvc-pod
  namespace: project-1
spec:
  volumes:
    - name: test-pvc-project-1
      persistentVolumeClaim:
       claimName: test-pvc-project-1
  containers:
    - name: test-container
      image: nginx
      ports:
        - containerPort: 80
          name: "http-server"
      volumeMounts:
        - mountPath: "/usr/share/nginx/html"
          name: test-pvc-project-1
EOF
```

7. Vérifiez si le pod est en cours d'exécution et si il a monté le volume.

```
oc describe pods test-pvc-pod -n project-1
```

Valider l'accès pour créer des demandes de volume persistant ou des pods dans un autre projet ou utiliser des ressources dédiées à un autre projet

1. Connectez-vous en tant qu'utilisateur ocp-project-1, développeur dans Project-1.
2. Créez un PVC dans Project-1 en utilisant le storageclass affecté au projet-2.

```
cat << EOF | oc create -f -
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: test-pvc-project-1-sc-2
  namespace: project-1
  annotations:
    trident.netapp.io/reclaimPolicy: Retain
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 1Gi
  storageClassName: project-2-sc
EOF
```

3. Création d'une demande de volume persistant dans Project-2.

```
cat << EOF | oc create -f -
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: test-pvc-project-2-sc-1
  namespace: project-2
  annotations:
    trident.netapp.io/reclaimPolicy: Retain
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 1Gi
  storageClassName: project-1-sc
EOF
```

4. Assurez-vous que les ESV `test-pvc-project-1-sc-2` et `test-pvc-project-2-sc-1` n'ont pas été créés.

```
oc get pvc -n project-1
oc get pvc -n project-2
```

5. Créez un pod dans Project-2.

```
cat << EOF | oc create -f -
kind: Pod
apiVersion: v1
metadata:
  name: test-pvc-pod
  namespace: project-1
spec:
  containers:
    - name: test-container
      image: nginx
      ports:
        - containerPort: 80
          name: "http-server"
EOF
```

Validez l'accès pour afficher et modifier les projets, ResourceQuotas et Storageclasses

1. Connectez-vous en tant qu'utilisateur ocp-project-1, développeur dans Project-1.
2. Vérifiez l'accès pour créer de nouveaux projets.

```
oc create ns sub-project-1
```

3. Valider l'accès pour afficher les projets.

```
oc get ns
```

4. Vérifiez si l'utilisateur peut afficher ou modifier ResourceQuotas dans Project-1.

```
oc get resourcequotas -n project-1
oc edit resourcequotas project-1-sc-rq -n project-1
```

5. Vérifiez que l'utilisateur a accès à l'affichage des données de stockage.

```
oc get sc
```

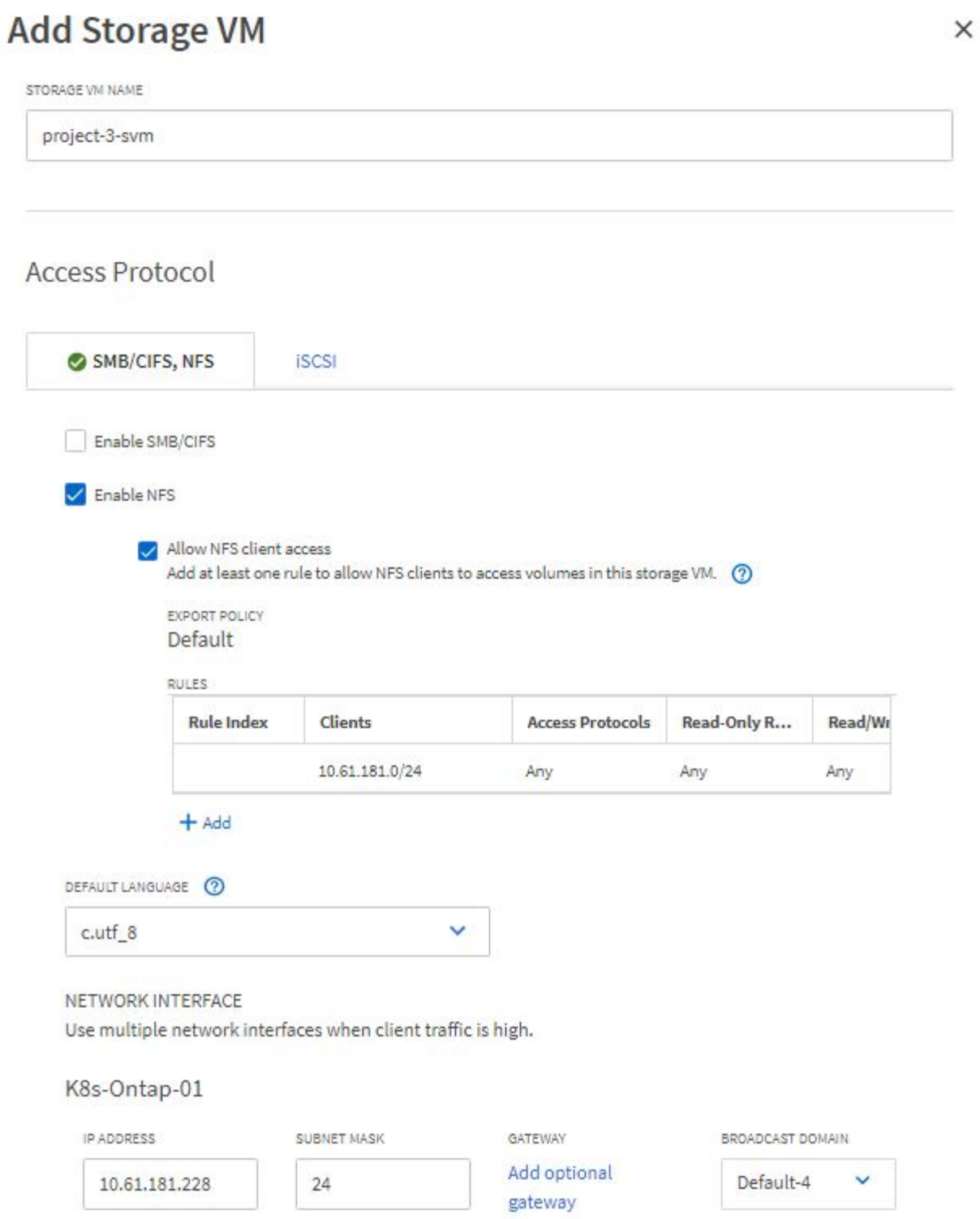
6. Vérifiez l'accès pour décrire les storageclasses.
7. Validez l'accès de l'utilisateur pour modifier les storageclasses.

```
oc edit sc project-1-sc
```

Évolutivité : ajout de projets

Dans une configuration mutualisée, l'ajout de nouveaux projets avec des ressources de stockage nécessite une configuration supplémentaire pour garantir que la colocation n'est pas respectée. Pour ajouter d'autres projets dans un cluster mutualisé, effectuez les opérations suivantes :

1. Connectez-vous au cluster NetApp ONTAP en tant qu'administrateur du stockage.
2. Accédez à `Storage → Storage VMs` et cliquez sur `Add`. Créez un nouveau SVM dédié au projet-3. Créer également un compte vsadmin pour gérer le SVM et ses ressources



1. Connectez-vous au cluster Red Hat OpenShift en tant qu'administrateur de cluster.
2. Créer un nouveau projet.

```
oc create ns project-3
```

3. Assurez-vous que le groupe d'utilisateurs du projet Project-3 est créé sur IDP et synchronisé avec le

cluster OpenShift.

```
oc get groups
```

4. Créer le rôle de développeur du projet-3.

```
cat << EOF | oc create -f -
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  namespace: project-3
  name: developer-project-3
rules:
  - verbs:
      - '*'
    apiGroups:
      - apps
      - batch
      - autoscaling
      - extensions
      - networking.k8s.io
      - policy
      - apps.openshift.io
      - build.openshift.io
      - image.openshift.io
      - ingress.operator.openshift.io
      - route.openshift.io
      - snapshot.storage.k8s.io
      - template.openshift.io
    resources:
      - '*'
  - verbs:
      - '*'
    apiGroups:
      - ''
    resources:
      - bindings
      - configmaps
      - endpoints
      - events
      - persistentvolumeclaims
      - pods
      - pods/log
      - pods/attach
      - podtemplates
```

```
      - replicationcontrollers
      - services
      - limitranges
      - namespaces
      - componentstatuses
      - nodes
  - verbs:
      - '*'
    apiGroups:
      - trident.netapp.io
    resources:
      - tridentsnapshots
EOF
```

> La définition de rôle fournie dans cette section n'est qu'un exemple. Le rôle de développeur doit être défini en fonction des exigences de l'utilisateur final.

1. Créer RoleBinding pour les développeurs dans projet-3 liant le rôle développeur-projet-3 au groupe correspondant (ocp-project-3) dans projet-3.

```
cat << EOF | oc create -f -
kind: RoleBinding
apiVersion: rbac.authorization.k8s.io/v1
metadata:
  name: project-3-developer
  namespace: project-3
subjects:
  - kind: Group
    apiGroup: rbac.authorization.k8s.io
    name: ocp-project-3
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: Role
  name: developer-project-3
EOF
```

2. Connectez-vous au cluster Red Hat OpenShift en tant qu'administrateur du stockage
3. Créer un système back-end Trident et le mapper sur le SVM dédié au projet-3. NetApp recommande d'utiliser le compte vsadmin du SVM afin de connecter le backend au SVM au lieu d'utiliser l'administrateur du cluster ONTAP

```
cat << EOF | tridentctl -n trident create backend -f
{
    "version": 1,
    "storageDriverName": "ontap-nas",
    "backendName": "nfs_project_3",
    "managementLIF": "172.21.224.210",
    "dataLIF": "10.61.181.228",
    "svm": "project-3-svm",
    "username": "vsadmin",
    "password": "NetApp!23"
}
EOF
```

Nous utilisons le pilote ontap-nas dans cet exemple. Utilisez le pilote approprié pour créer le back-end en fonction du cas d'utilisation.

Nous partons du principe que Trident est installé dans le projet trident.

1. Créez la classe de stockage pour Project-3 et configurez-la pour qu'elle utilise les pools de stockage du système back-end dédié au projet-3.

```
cat << EOF | oc create -f -
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: project-3-sc
provisioner: csi.trident.netapp.io
parameters:
  backendType: ontap-nas
  storagePools: "nfs_project_3:.*"
EOF
```

2. Créer un Resourcequota pour limiter les ressources dans le projet-3 demandant du stockage de storageclasses dédié à d'autres projets.

```
cat << EOF | oc create -f -
kind: ResourceQuota
apiVersion: v1
metadata:
  name: project-3-sc-rq
  namespace: project-3
spec:
  hard:
    project-1-sc.storageclass.storage.k8s.io/persistentvolumeclaims: 0
    project-2-sc.storageclass.storage.k8s.io/persistentvolumeclaims: 0
EOF
```

3. Patch des ResourceQuotas dans d'autres projets pour limiter les ressources de ces projets à l'accès au stockage depuis le storageclass dédié au projet-3.

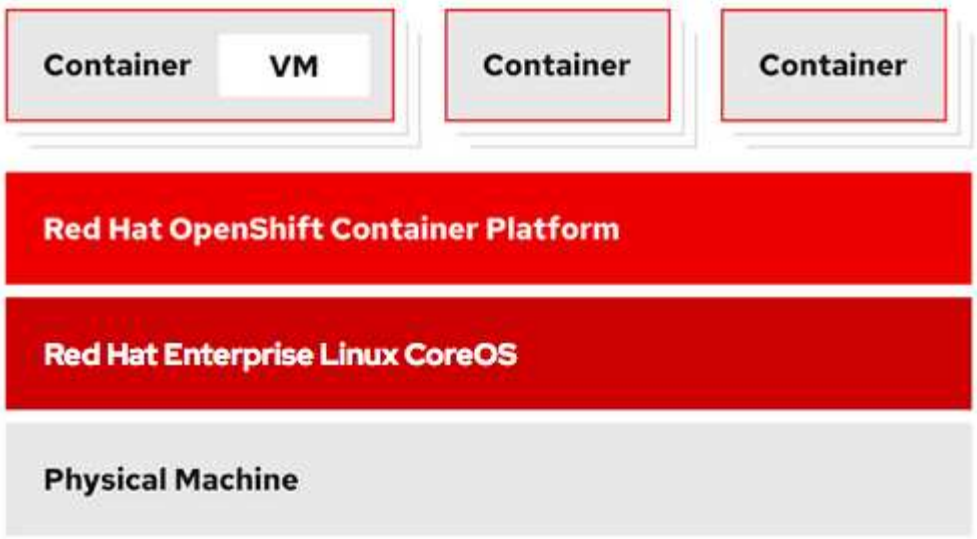
```
oc patch resourcequotas project-1-sc-rq -n project-1 --patch
'{"spec":{"hard":{ "project-3-
sc.storageclass.storage.k8s.io/persistentvolumeclaims": 0}}}'
oc patch resourcequotas project-2-sc-rq -n project-2 --patch
'{"spec":{"hard":{ "project-3-
sc.storageclass.storage.k8s.io/persistentvolumeclaims": 0}}}'
```

Red Hat OpenShift Virtualization avec NetApp ONTAP

Red Hat OpenShift Virtualization avec NetApp ONTAP

Selon l'utilisation, les conteneurs et les machines virtuelles peuvent servir de plateformes optimales pour différents types d'applications. Par conséquent, de nombreuses entreprises exécutent certaines de leurs workloads sur des conteneurs et certaines sur des VM. Les entreprises doivent souvent relever des challenges supplémentaires : la gestion de plateformes distinctes : un hyperviseur pour les machines virtuelles et un orchestrateur de conteneur pour les applications.

Pour relever ce défi, Red Hat a lancé OpenShift Virtualization (anciennement appelé Container Native Virtualization) à partir de la version 4.6 d'OpenShift. La fonction de virtualisation OpenShift vous permet d'exécuter et de gérer les machines virtuelles avec des conteneurs sur la même installation OpenShift Container Platform. Elle offre une fonctionnalité de gestion hybride permettant d'automatiser le déploiement et la gestion des machines virtuelles par l'intermédiaire des opérateurs. Outre la création de VM dans OpenShift, Red Hat prend également en charge l'importation de VM à partir de VMware vSphere, Red Hat Virtualization et Red Hat OpenStack Platform.



Certaines fonctionnalités comme la migration de VM en direct, le clonage de disques de VM, les snapshots de VM, etc. Sont également prises en charge par OpenShift Virtualization avec l'aide d'Astra Trident, avec le soutien de NetApp ONTAP. Des exemples de chacun de ces flux de travail sont présentés plus loin dans ce document dans leurs sections respectives.

Pour en savoir plus sur Red Hat OpenShift Virtualization, consultez la documentation "ici".

Déploiement pour OpenShift Virtualization

Déploiement de Red Hat OpenShift Virtualization avec NetApp ONTAP

Prérequis

- Un cluster Red Hat OpenShift (version ultérieure à la version 4.6) installé sur une infrastructure bare-Metal avec des nœuds worker RHCOS
- Le cluster OpenShift doit être installé via l'infrastructure provisionnée du programme d'installation (IPI).
- Déploiement de vérifications de l'état des machines pour garantir la haute disponibilité des machines virtuelles
- Un cluster NetApp ONTAP
- Astra Trident installé sur le cluster OpenShift
- Un système back-end Trident configuré avec un SVM sur le cluster ONTAP
- Classe de stockage configurée sur le cluster OpenShift avec Astra Trident en tant que mécanisme de provisionnement
- L'accès cluster-admin au cluster Red Hat OpenShift
- Accès au cluster NetApp ONTAP par administrateur
- Une station de travail d'administration avec des outils tridentctl et oc installés et ajoutés à $PATH

OpenShift Virtualization est gérée par un opérateur installé sur le cluster OpenShift et impose une surcharge supplémentaire pour la mémoire, le processeur et le stockage, ce qui doit être pris en compte lors de la planification des exigences matérielles du cluster. Voir la documentation "ici" pour en savoir plus.

Vous pouvez également spécifier un sous-ensemble des nœuds du cluster OpenShift pour héberger les opérateurs, contrôleurs et VM OpenShift Virtualization en configurant des règles de placement des nœuds.

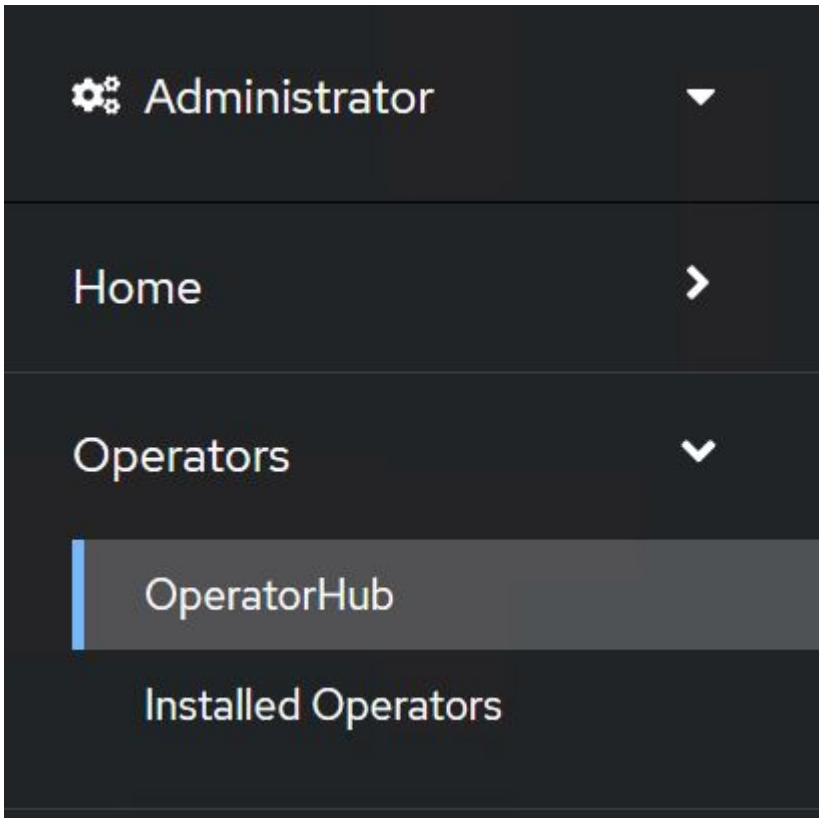
Pour configurer les règles de placement des nœuds pour OpenShift Virtualization, suivez la documentation "ici".

Pour la prise en charge du stockage d'OpenShift Virtualization, NetApp recommande d'utiliser une classe de stockage dédiée qui demande le stockage auprès d'un back-end Trident spécifique, qui est ensuite soutenue par un SVM dédié. Cela permet à un niveau d'architecture en colocation s'agissant des données servies aux charges de travail basées sur des VM du cluster OpenShift.

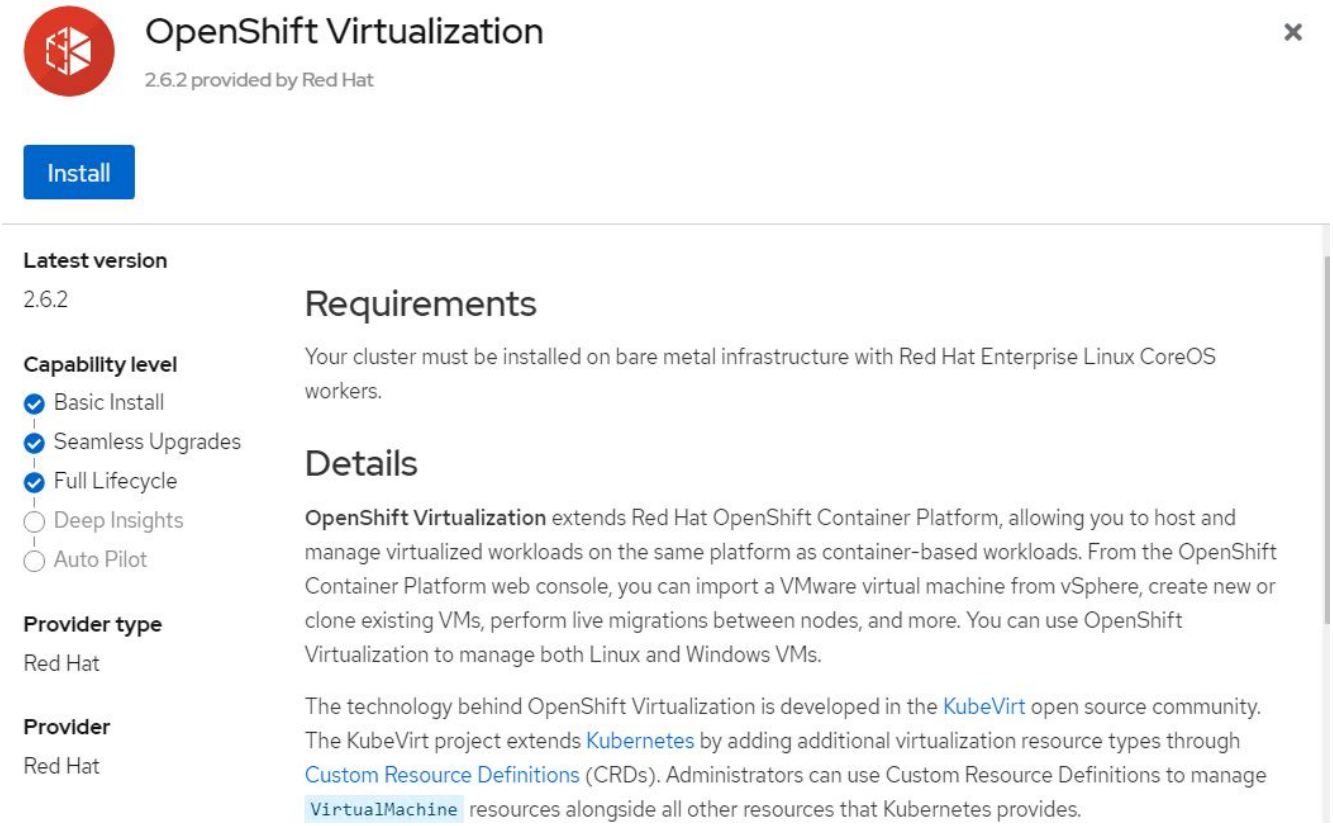
Déploiement de Red Hat OpenShift Virtualization avec NetApp ONTAP

Pour installer OpenShift Virtualization, procédez comme suit :

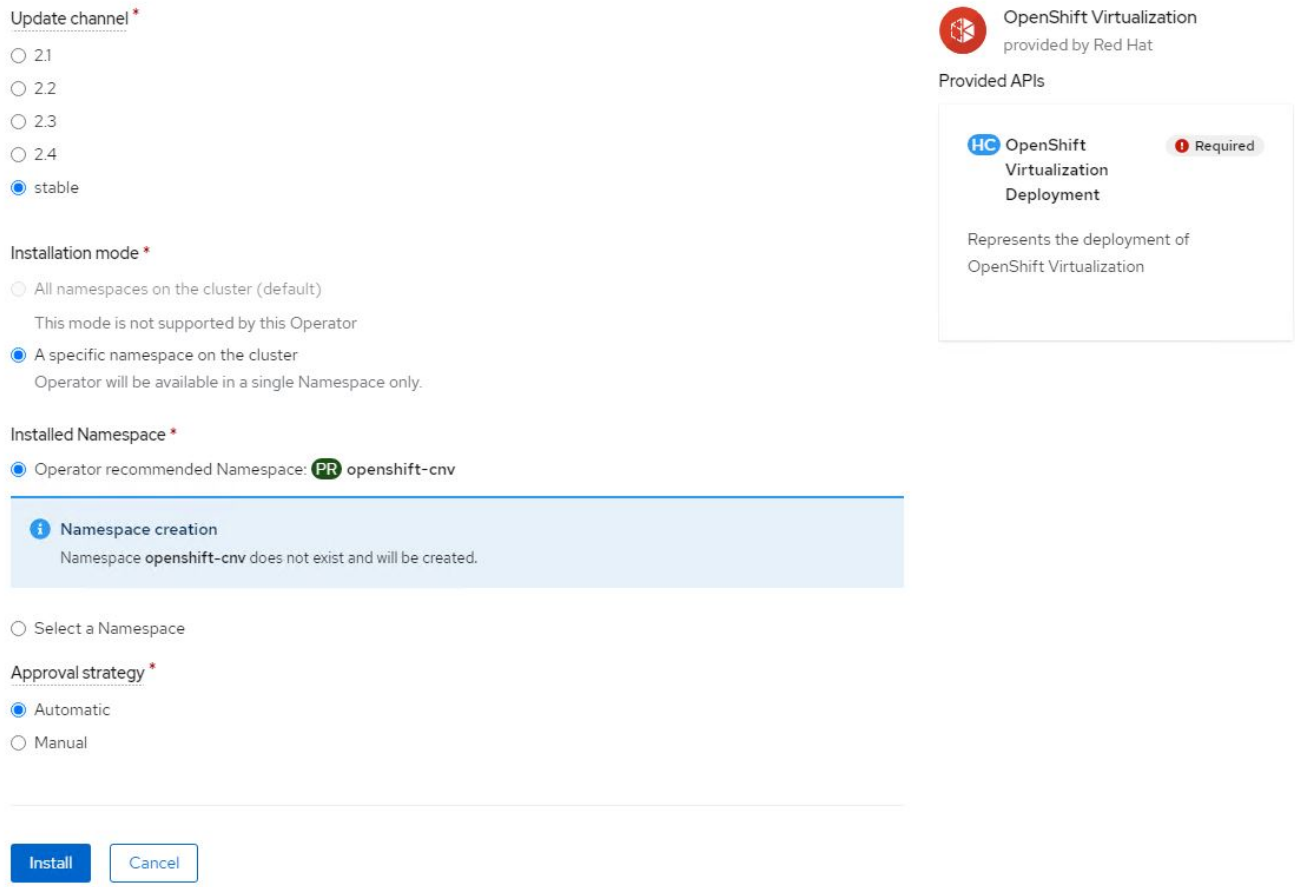
1. Connectez-vous au cluster sans système d'exploitation Red Hat OpenShift avec l'accès cluster-admin.
2. Sélectionnez Administrateur dans la liste déroulante perspective.
3. Accédez à Operators > OperatorHub et recherchez OpenShift Virtualization.



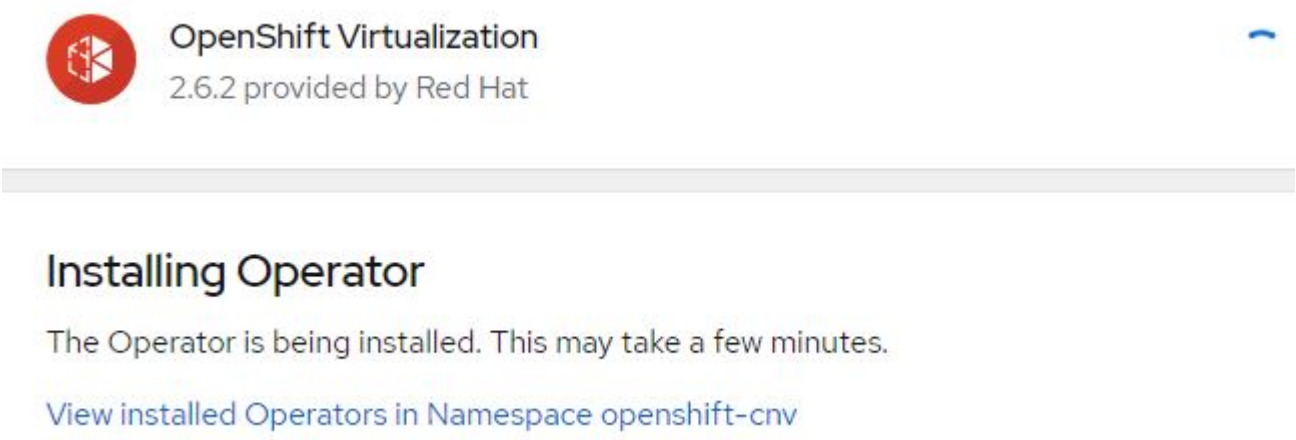
4. Sélectionnez la mosaïque OpenShift Virtualization et cliquez sur Install.



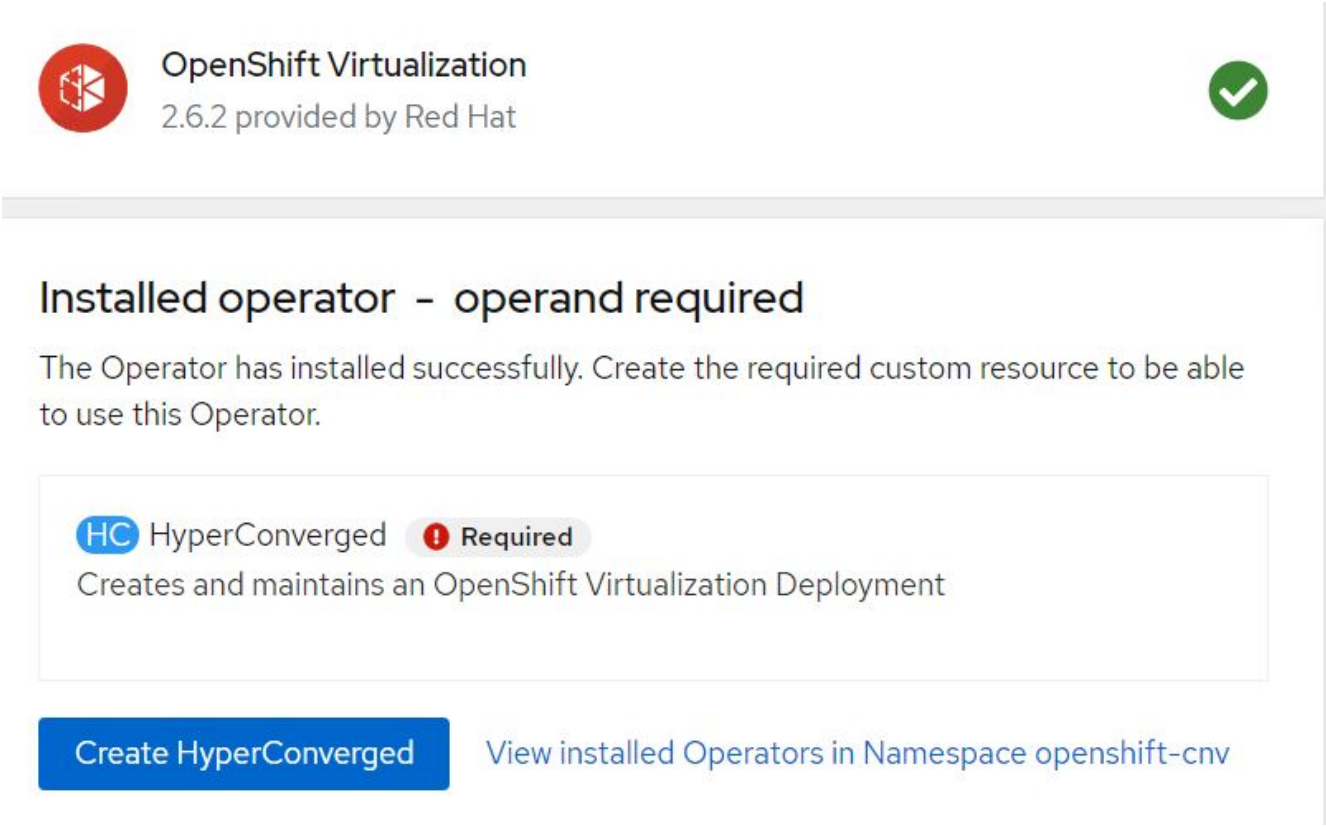
5. Sur l’écran installer l’opérateur, laissez tous les paramètres par défaut et cliquez sur installer.



6. Attendre la fin de l'installation par l'opérateur.



7. Une fois l'opérateur installé, cliquez sur Créer une Hyperconvergé.



8. Sur l'écran Créer une Hyperconvergeance, cliquez sur Créer, accepter tous les paramètres par défaut. Cette étape démarre l'installation d'OpenShift Virtualization.

Name *

kubevirt-hyperconverged

Labels

app=frontend

Infra

infra HyperConvergedConfig influences the pod configuration (currently only placement) for all the infra components needed on the virtualization enabled cluster but not necessarely directly on each node running VMs/VMIs.

Workloads

workloads HyperConvergedConfig influences the pod configuration (currently only placement) of components which need to be running on a node where virtualization workloads should be able to run. Changes to Workloads HyperConvergedConfig can be applied only without existing workload.

Bare Metal Platform

true

BareMetalPlatform indicates whether the infrastructure is baremetal.

Feature Gates

featureGates is a map of feature gate flags. Setting a flag to `true` will enable the feature. Setting `false` or removing the feature gate, disables the feature.

Local Storage Class Name

LocalStorageClassName the name of the local storage class.

Create Cancel

9. Une fois que tous les pods passent à l'état d'exécution dans l'espace de noms openshift-cnv et que l'opérateur OpenShift Virtualization est dans l'état « réussi », l'opérateur est prêt à l'emploi. Les VM peuvent désormais être créés sur le cluster OpenShift.

Project: openshift-cnv

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	Managed Namespaces	Status	Last updated	Provided APIs
OpenShift Virtualization 2.6.2 provided by Red Hat	NS openshift-cnv	Succeeded Up to date	May 18, 8:02 pm	OpenShift Virtualization Deployment HostPathProvisioner deployment

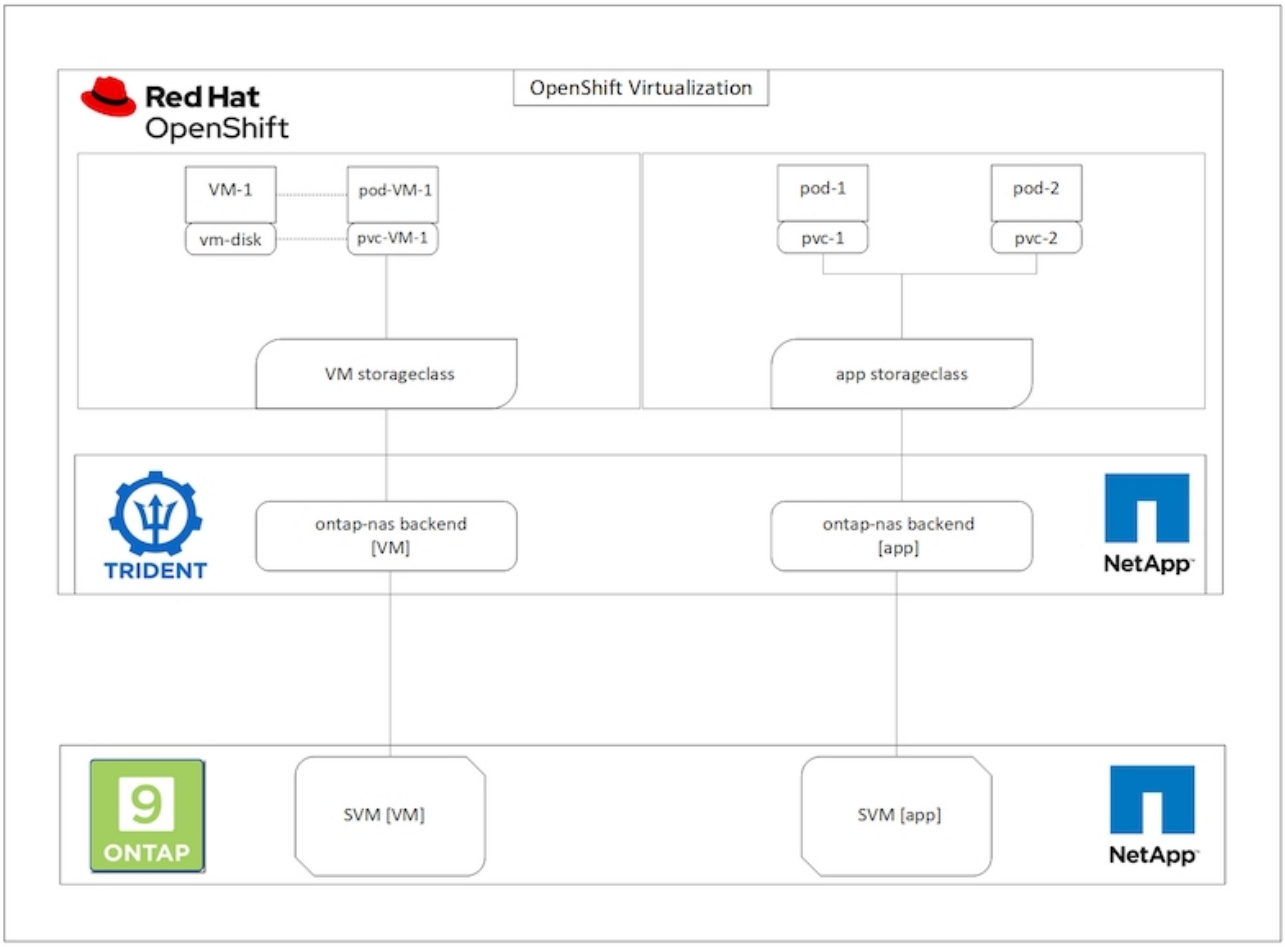
Flux de travail

Flux de travail : Red Hat OpenShift Virtualization avec NetApp ONTAP

Créer une machine virtuelle

Les machines virtuelles sont des déploiements avec état qui requièrent des volumes pour héberger le système d'exploitation et les données. Avec CNV, les machines virtuelles étant exécutées comme des pods, ces

dernières sont sauvegardées par des volumes persistants hébergés sur NetApp ONTAP via Trident. Ces volumes sont connectés en tant que disques et stockent l'intégralité du système de fichiers, y compris la source de démarrage de la machine virtuelle.



Pour créer un serveur virtuel sur le cluster OpenShift, effectuez les opérations suivantes :

1. Accédez à charges de travail > virtualisation > ordinateurs virtuels, puis cliquez sur Créer > avec l'assistant.
2. Sélectionnez le système d'exploitation souhaité et cliquez sur Suivant.
3. Si aucune source d'amorçage n'est configurée sur le système d'exploitation sélectionné, vous devez la configurer. Dans Source d'amorçage, indiquez si vous souhaitez importer l'image OS à partir d'une URL ou d'un registre et fournissez les détails correspondants. Développez Advanced et sélectionnez la classe de stockage sauvegardée par Trident. Cliquez ensuite sur Suivant.

Boot source

This template does not have a boot source. Provide a custom boot source for this **CentOS 8.0+ VM** virtual machine.

Boot source type *

Import via URL (creates PVC)

Import URL *

https://access.cdn.redhat.com/content/origin/files/sha256/58/588167f828001e57688ec4b9b31c11a59d532489f527488ebc89ac5e952...

Example: For RHEL, visit the **RHEL download page** (requires login) and copy the download link URL of the KVM guest image

☑ Mount this as a CD-ROM boot source

Persistent Volume Claim size *

5 GiB

Ensure your PVC size covers the requirements of the uncompressed image and any other space requirements. More storage can be added later.

Advanced

Storage class *

basic (default)

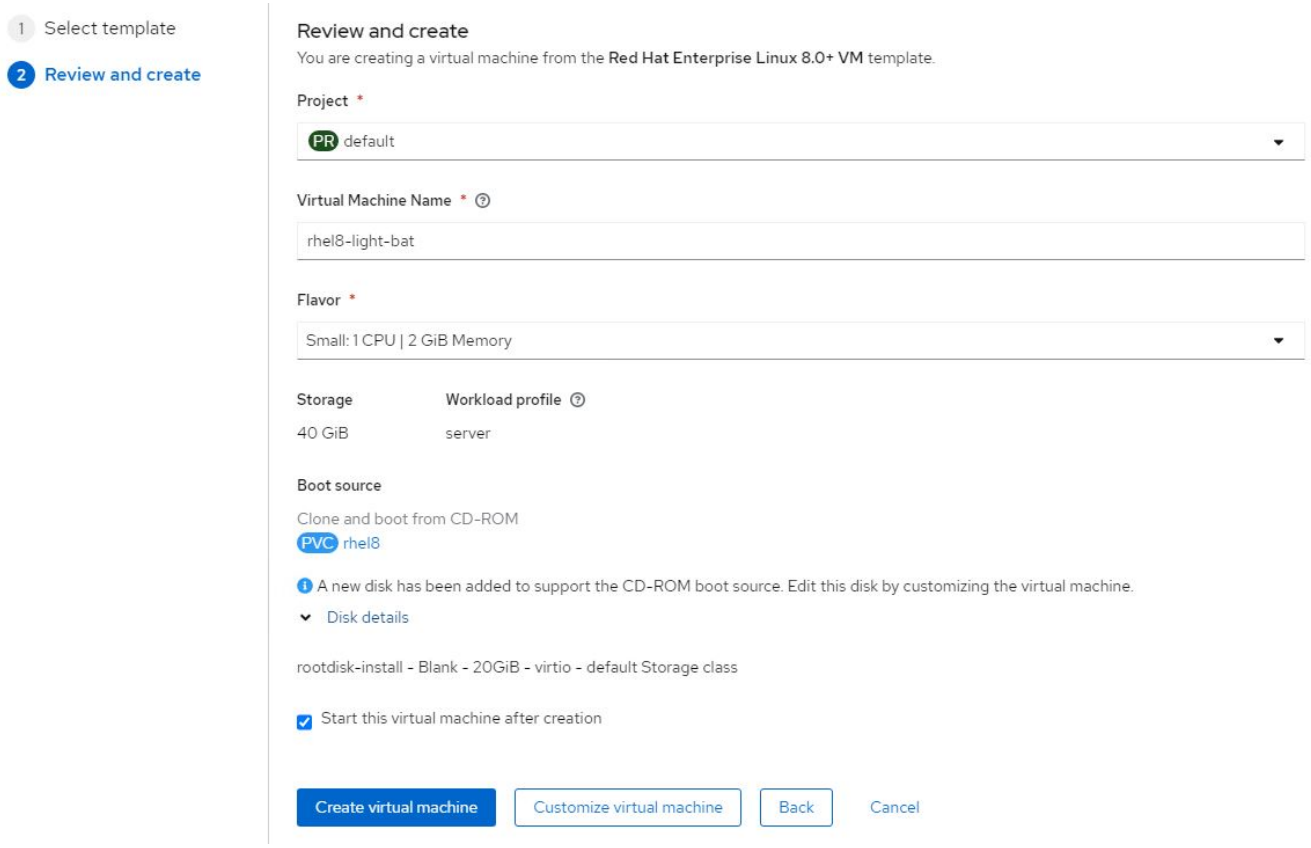
Access mode *

Single User (RWO)

Volume mode *

Filesystem

4. Si une source d'amorçage est déjà configurée sur le système d'exploitation sélectionné, l'étape précédente peut être ignorée.
5. Dans le volet révision et création, sélectionnez le projet dans lequel vous souhaitez créer la machine virtuelle et indiquez les détails de la machine virtuelle. Assurez-vous que la source de démarrage est sélectionnée pour être Clone et boot à partir du CD-ROM avec le PVC approprié affecté au système d'exploitation sélectionné.



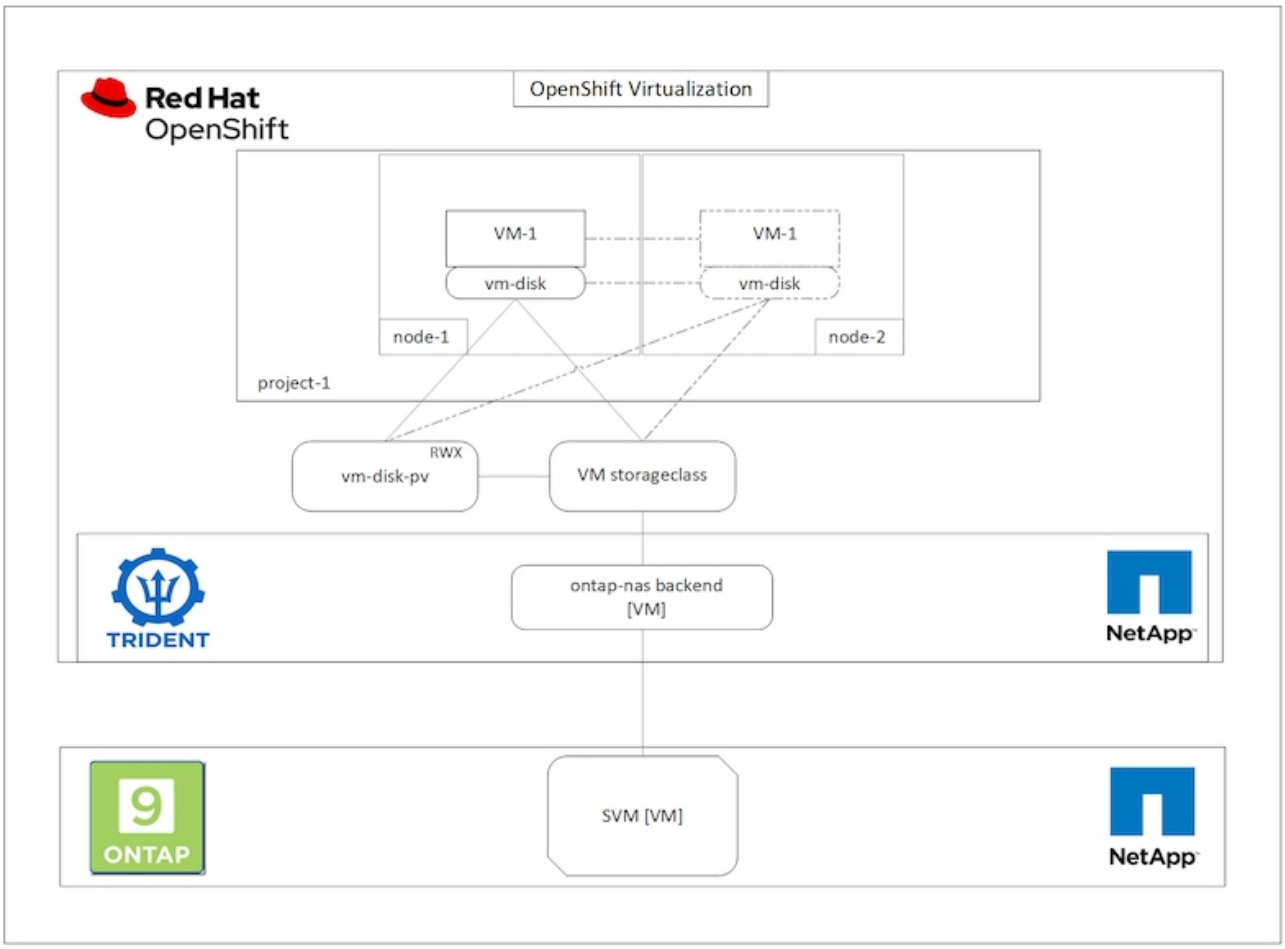
6. Si vous souhaitez personnaliser la machine virtuelle, cliquez sur Personnaliser la machine virtuelle et modifiez les paramètres requis.
7. Cliquez sur Créer une machine virtuelle pour créer la machine virtuelle ; le pod correspondant est alors pivotez en arrière-plan.

Lorsqu'une source d'amorçage est configurée pour un modèle ou un système d'exploitation à partir d'une URL ou d'un registre, elle crée une demande de volume persistant dans le `openshift-virtualization-os-images` Projetez et téléchargez l'image hôte KVM sur la demande de volume persistant. Vous devez vous assurer que les demandes de volume persistant du modèle disposent d'un espace provisionné suffisant pour prendre en charge l'image hôte KVM pour le système d'exploitation correspondant. Ces demandes de volume virtuel sont ensuite clonées et reliées en tant que rootdisks aux machines virtuelles lors de leur création à l'aide des modèles respectifs de n'importe quel projet.

Flux de travail : Red Hat OpenShift Virtualization avec NetApp ONTAP

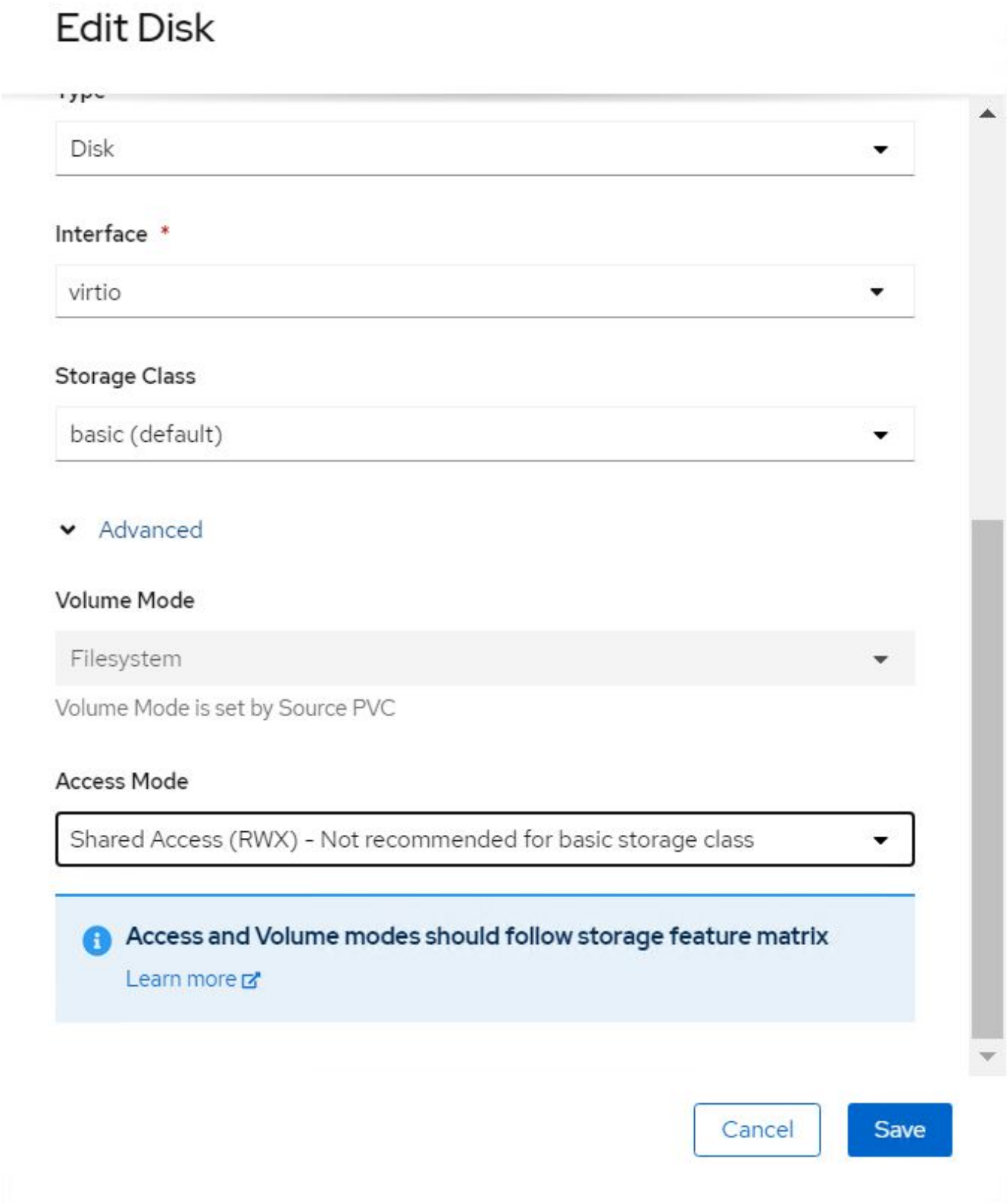
Migration en direct des machines virtuelles

Live migration est un processus de migration d'une instance de VM d'un nœud vers un autre dans un cluster OpenShift sans aucun temps d'indisponibilité. Pour que la migration en direct puisse fonctionner dans un cluster OpenShift, les VM doivent être liés aux demandes de volume virtuel avec le mode d'accès ReadWriteMany partagé. Le système back-end Astra Trident configuré avec un SVM sur un cluster NetApp ONTAP activé pour le protocole NFS prend en charge l'accès partagé ReadWriteMany pour les demandes de volume persistant. Par conséquent, les machines virtuelles avec des demandes de volume persistant demandées par les classes de stockage provisionnées par Trident à partir d'un SVM compatible NFS peuvent être migrées sans temps d'indisponibilité.



Pour créer une VM liée à des demandes de volume virtuel avec un accès ReadWriteMany partagé :

1. Accédez à charges de travail > virtualisation > ordinateurs virtuels, puis cliquez sur Créer > avec l'assistant.
2. Sélectionnez le système d'exploitation souhaité et cliquez sur Suivant. Supposons que l'OS sélectionné dispose déjà d'une source d'amorçage configurée avec celle-ci.
3. Dans le volet révision et création, sélectionnez le projet dans lequel vous souhaitez créer la machine virtuelle et indiquez les détails de la machine virtuelle. Assurez-vous que la source de démarrage est sélectionnée pour être Clone et boot à partir du CD-ROM avec le PVC approprié affecté au système d'exploitation sélectionné.
4. Cliquez sur Personnaliser l'ordinateur virtuel, puis sur stockage.
5. Cliquez sur les points de suspension en regard de rootdisk et assurez-vous que le storageclass provisionné à l'aide de Trident est sélectionné. Développez Avancé et sélectionnez accès partagé (RWX) pour le mode d'accès. Cliquez ensuite sur Enregistrer.



6. Cliquez sur vérifier et confirmer, puis sur Créer une machine virtuelle.

Pour migrer manuellement un VM vers un autre nœud du cluster OpenShift, procédez comme suit.

1. Accédez aux charges de travail > virtualisation > machines virtuelles.

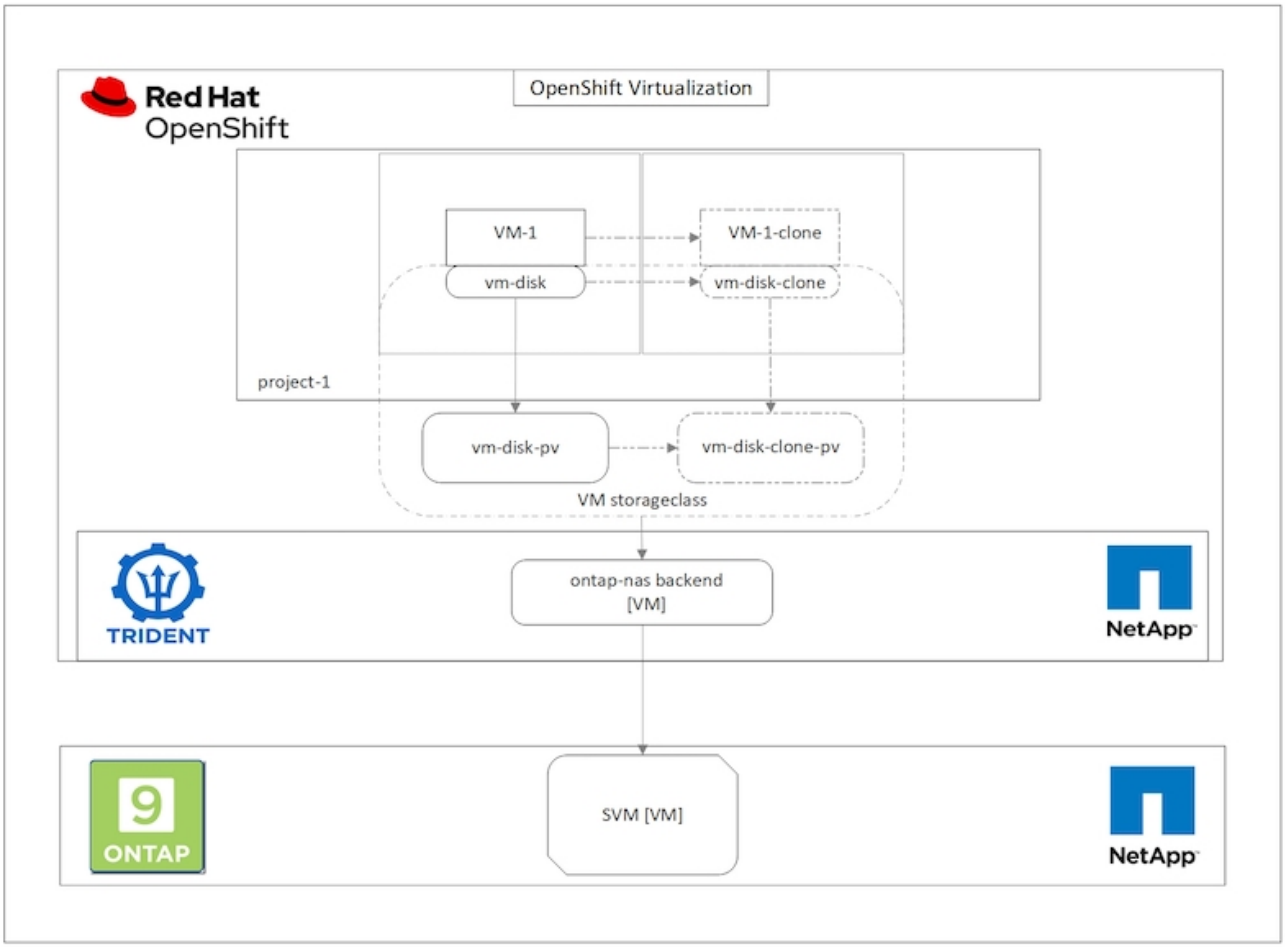
2. Pour la VM à migrer, cliquez sur les points de suspension, puis sur migrer la machine virtuelle.
3. Cliquez sur migrer lorsque le message s'affiche pour confirmer.

> Une instance de machine virtuelle d'un cluster OpenShift migre automatiquement vers un autre nœud lorsque le nœud d'origine est placé en mode maintenance si la stratégie d'éviction est définie sur LiveMigrate.

Flux de travail : Red Hat OpenShift Virtualization avec NetApp ONTAP

Clonage de VM

Le clonage d'une machine virtuelle existante dans OpenShift est réalisé avec la prise en charge de la fonctionnalité de clonage de volumes CSI d'Astra Trident. Le clonage de volumes CSI permet de créer une nouvelle demande de volume persistant en utilisant une demande de volume en tant que source de données en dupliquant son volume persistant. Une fois le nouveau PVC créé, il fonctionne comme une entité distincte et sans lien ou dépendance sur le PVC source.



Le clonage de volumes CSI peut prendre en compte certaines restrictions :

1. Le PVC source et le PVC de destination doivent être dans le même projet.
2. Le clonage est pris en charge au sein de la même classe de stockage.
3. Le clonage n'est possible que lorsque les volumes source et de destination utilisent le même paramètre

Volumemode. Par exemple, un volume de bloc ne peut être cloné que vers un autre volume de bloc.

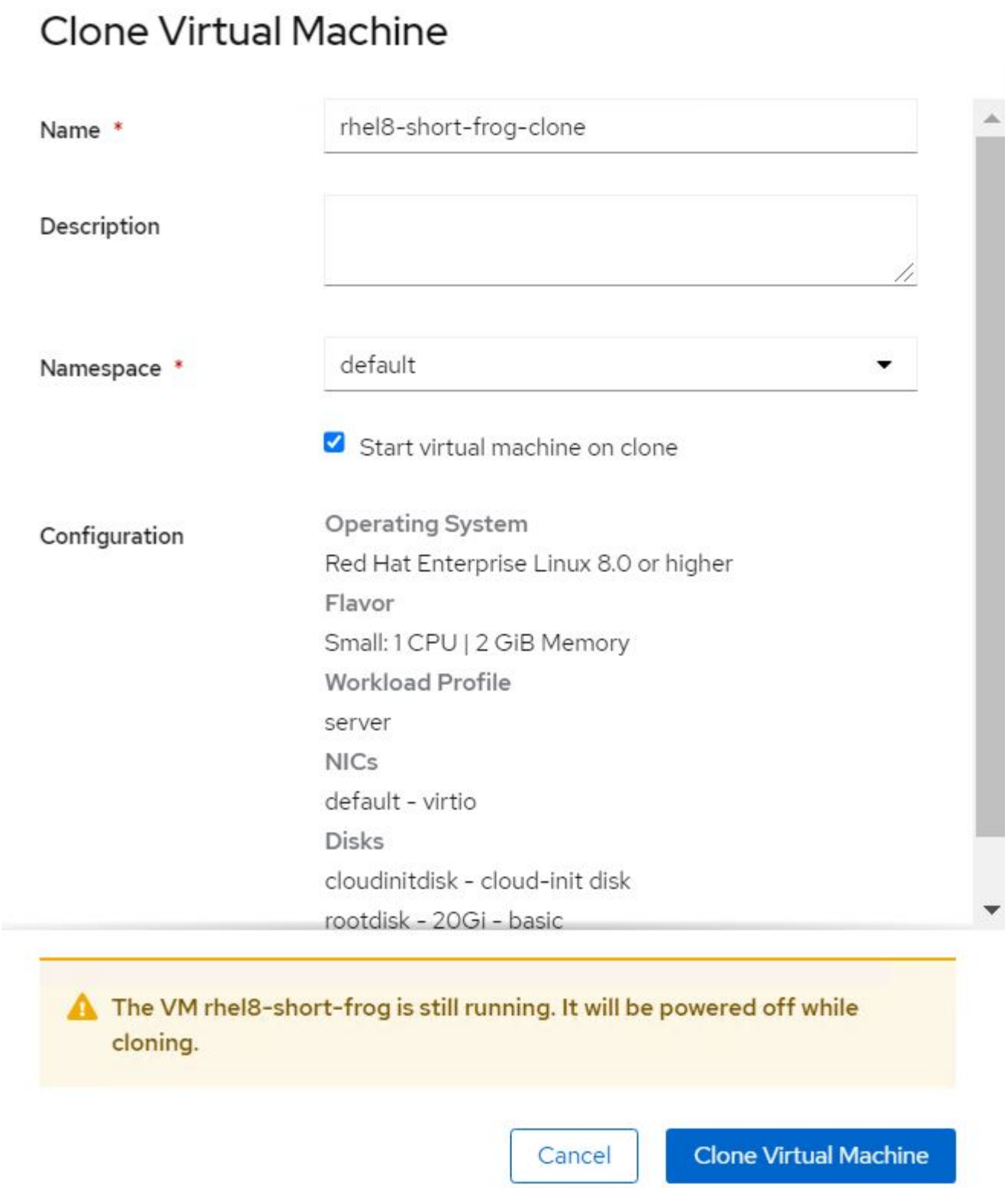
Les VM d'un cluster OpenShift peuvent être clonés de deux manières :

1. En cours d'arrêt de la machine virtuelle source
2. En conservant la machine virtuelle source en service

En cours d'arrêt de la machine virtuelle source

Le clonage d'une machine virtuelle existante en fermant cette machine virtuelle est une fonctionnalité OpenShift native prise en charge d'Astra Trident. Procédez comme suit pour cloner une machine virtuelle.

1. Accédez à charges de travail > virtualisation > machines virtuelles, puis cliquez sur les points de suspension situés à côté de la machine virtuelle que vous souhaitez cloner.
2. Cliquez sur Cloner l'ordinateur virtuel et fournissez les détails concernant la nouvelle machine virtuelle.

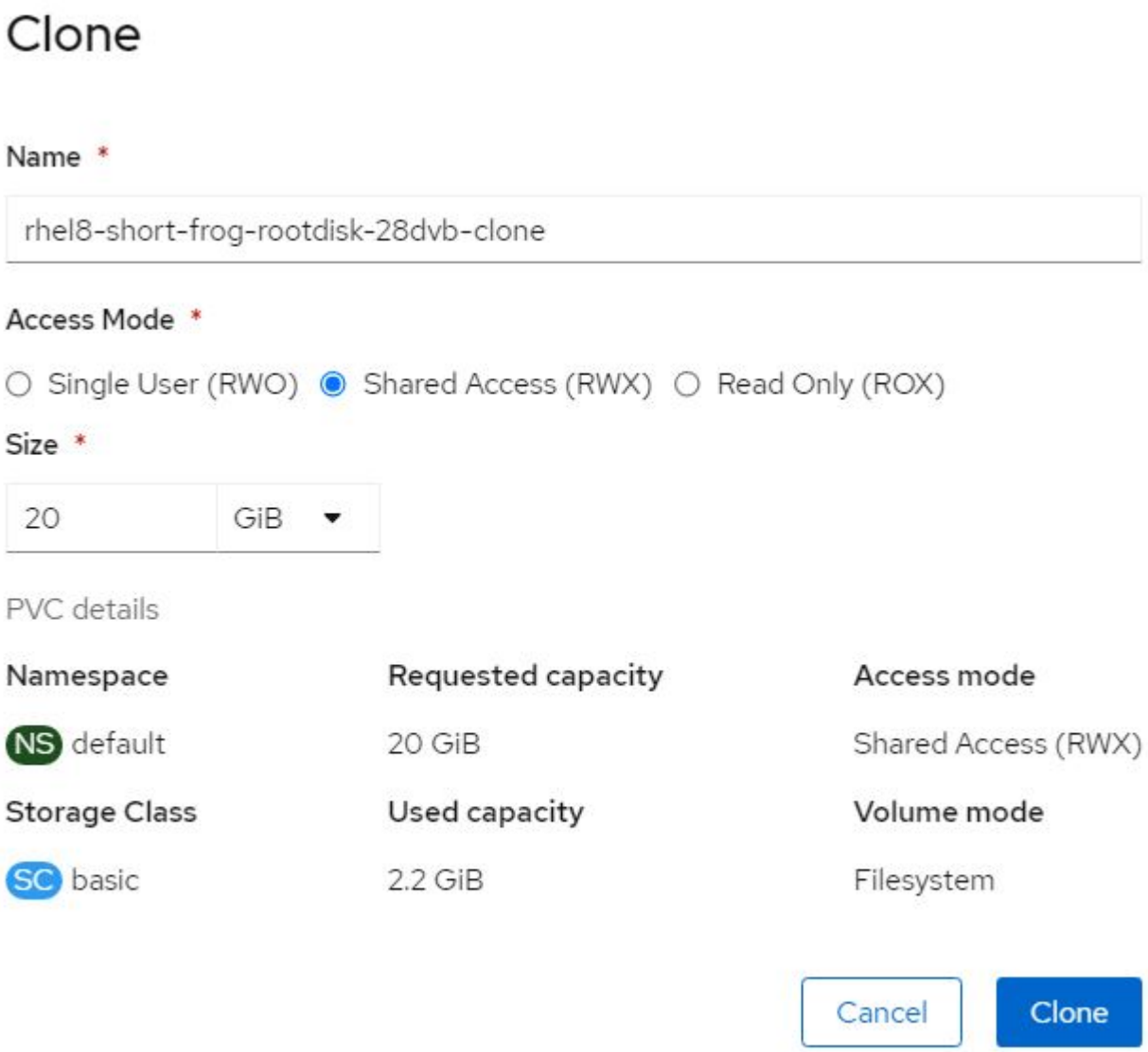


3. Cliquez sur Cloner l'ordinateur virtuel. La machine virtuelle source est arrêtée et commence la création de la machine virtuelle clone.
4. Une fois cette étape terminée, vous pouvez accéder au contenu de la machine virtuelle clonée et le vérifier.

En conservant la machine virtuelle source en service

Une machine virtuelle existante peut également être clonée en clonant le volume persistant existant de la machine virtuelle source, puis en créant une nouvelle machine virtuelle à l'aide du volume persistant cloné. Cette méthode n'exige pas l'arrêt de la machine virtuelle source. Procédez comme suit pour cloner une machine virtuelle sans la désactiver.

1. Accédez à Storage > PersistentVolumeClaims et cliquez sur les points de suspension en regard du volume persistant associé à la machine virtuelle source.
2. Cliquez sur Cloner le PVC et fournir les détails du nouveau PVC.



3. Cliquez ensuite sur Cloner. Cela crée une demande de volume persistant pour la nouvelle machine virtuelle.
4. Accédez à charges de travail > virtualisation > machines virtuelles, puis cliquez sur Créer > avec YAML.
5. Dans la section spécifications > modèle > spécifications > volumes, fixez le PVC cloné à la place du disque conteneur. Fournir tous les autres détails relatifs à la nouvelle machine virtuelle selon vos besoins.

```
- name: rootdisk
  persistentVolumeClaim:
    claimName: rhel8-short-frog-rootdisk-28dvb-clone
```

6. Cliquez sur Créer pour créer la nouvelle machine virtuelle.
7. Une fois la machine virtuelle créée, accédez-y et vérifiez que la nouvelle machine virtuelle est un clone de la machine virtuelle source.

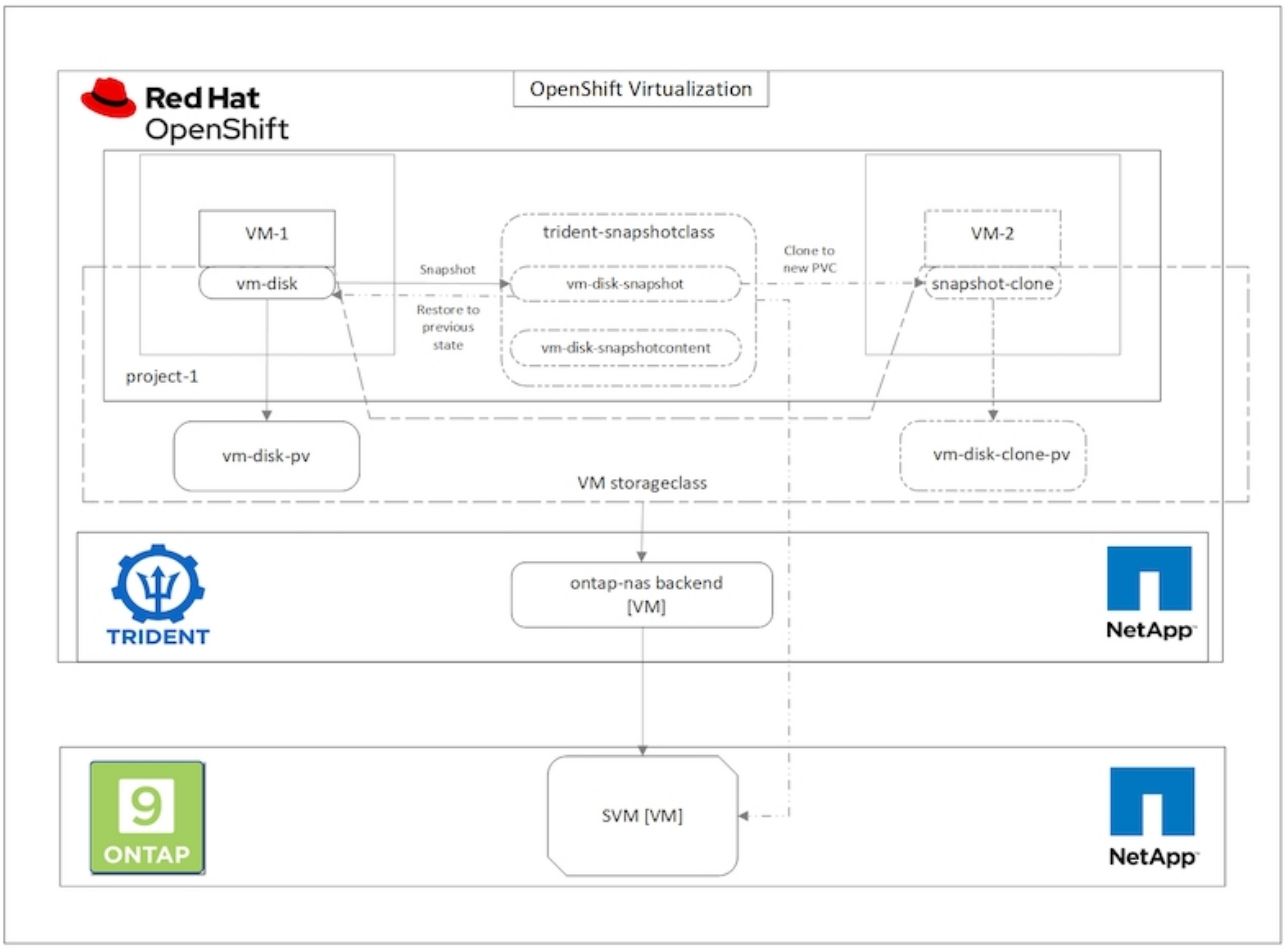
Flux de travail : Red Hat OpenShift Virtualization avec NetApp ONTAP

Créer un serveur virtuel à partir d'un Snapshot

Avec Astra Trident et Red Hat OpenShift, les utilisateurs peuvent créer un snapshot du volume persistant avec les classes de stockage provisionnées par celui-ci. Avec cette fonctionnalité, les utilisateurs peuvent effectuer une copie instantanée d'un volume et l'utiliser pour créer un nouveau volume ou restaurer le même volume à un état précédent. Cela permet d'activer ou de prendre en charge de nombreux cas d'utilisation, de la restauration aux clones en passant par la restauration des données.

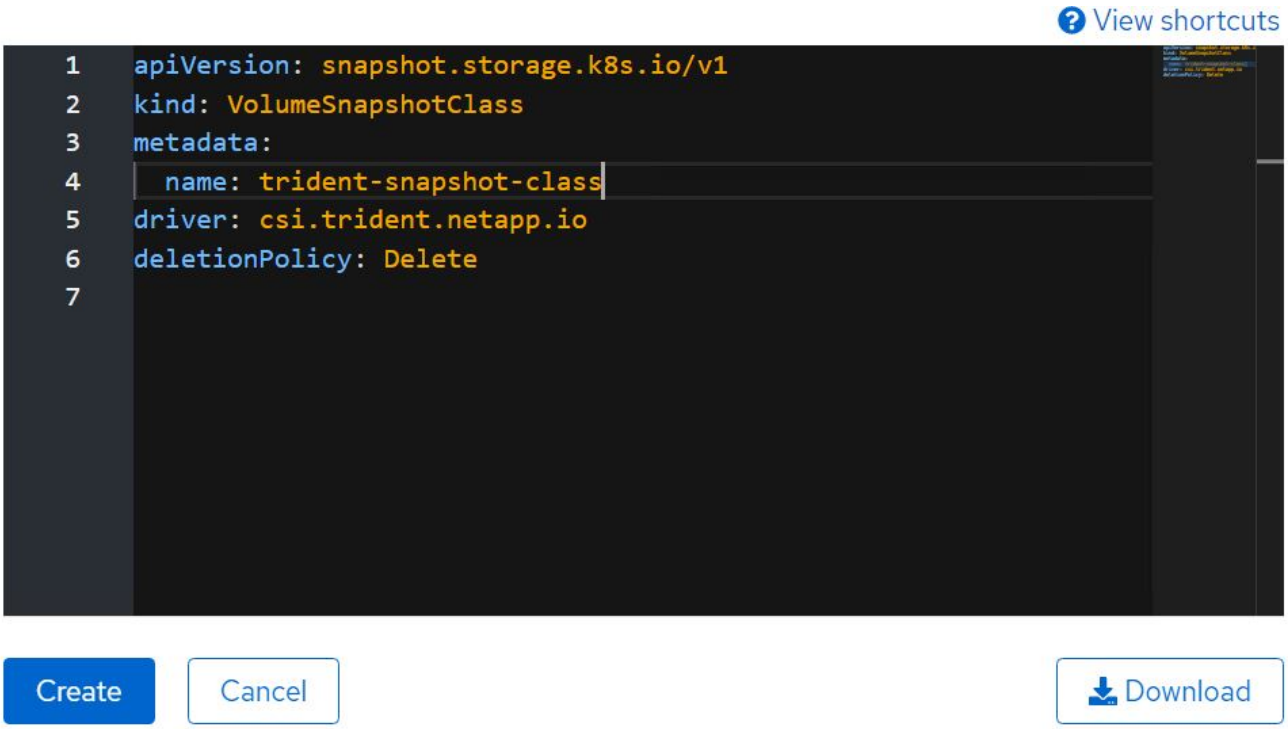
Pour les opérations Snapshot dans OpenShift, les ressources VolumeSnapshotClass, VolumeSnapshot et VolumeContent doivent être définies.

- Un VolumeSnapshotContent est le snapshot réellement pris à partir d'un volume du cluster. Il s'agit d'une ressource à l'échelle du cluster, semblable au volume persistant pour le stockage.
- Un VolumeSnapshot est une demande de création du snapshot d'un volume. Il est similaire à une demande de volume persistant.
- VolumeSnapshotClass permet à l'administrateur de spécifier différents attributs d'un VolumeSnapshot. Il vous permet d'avoir différents attributs pour les différents snapshots pris à partir du même volume.

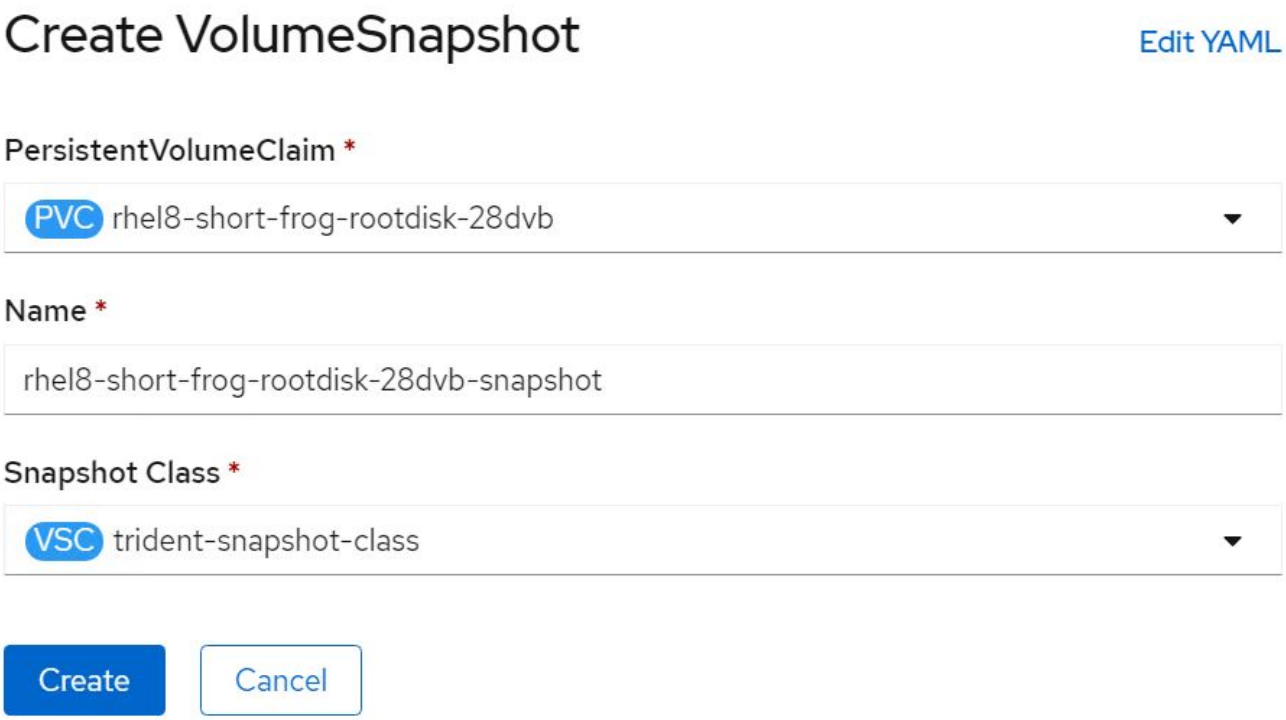


Pour créer le snapshot d’une machine virtuelle, effectuez la procédure suivante :

1. Créez une classe VolumeSnapshotClass qui peut ensuite être utilisée pour créer un Snapshot VolumeCas. Accédez à Storage > VolumeSnapshotclasses et cliquez sur Create VolumeSnapshotClass.
2. Entrez le nom de la classe d’instantanés, entrez csi.trident.netapp.io pour le pilote, puis cliquez sur Créer.



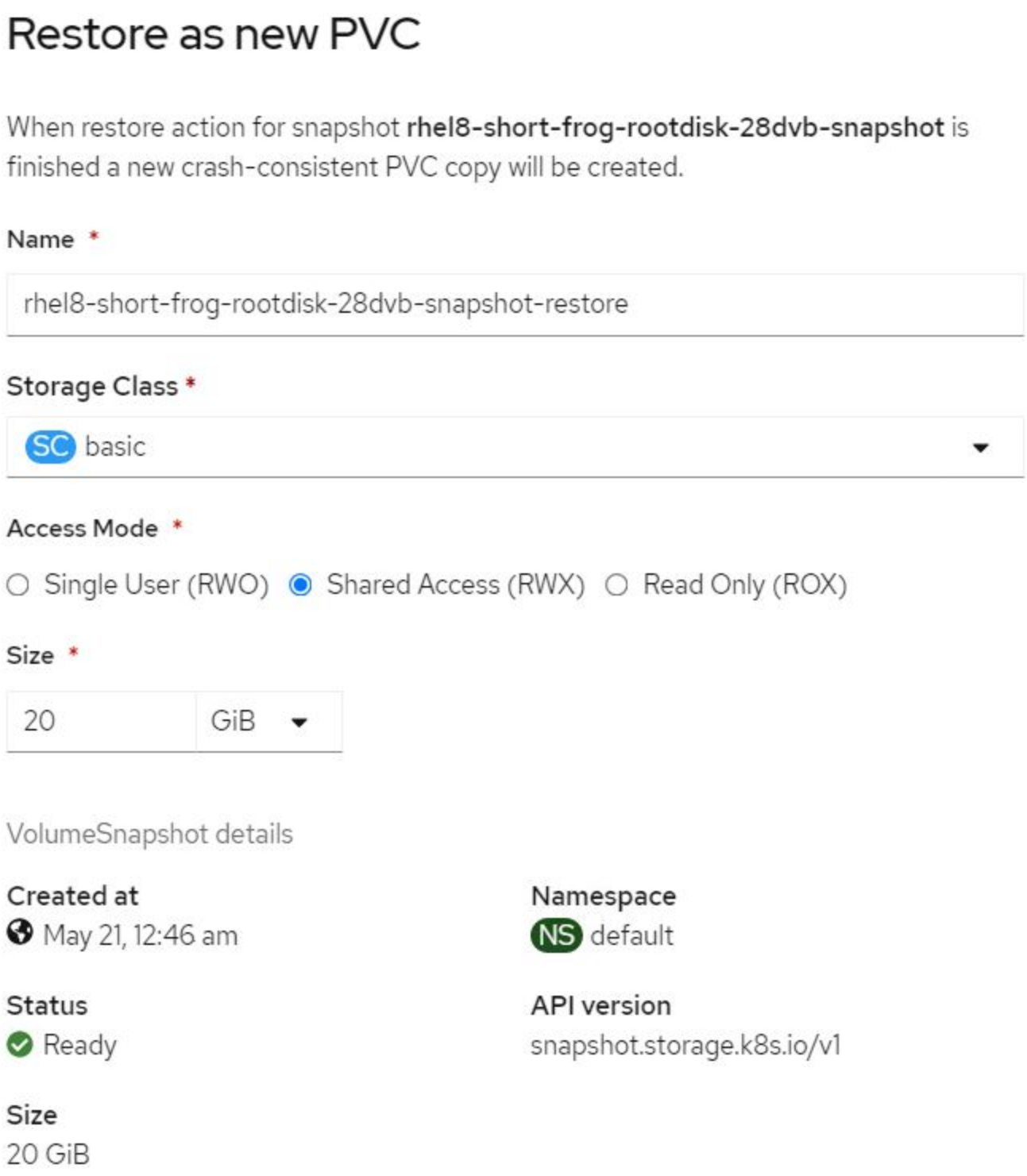
3. Identifiez le volume de volume persistant connecté à la machine virtuelle source, puis créez un Snapshot de cette demande de volume persistant. Accédez à `Storage > VolumeSnapshots` Puis cliquez sur Créer des copies Snapshot VolumeCas.
4. Sélectionnez la demande de volume persistant pour laquelle vous souhaitez créer l'instantané, entrez le nom de l'instantané ou acceptez la valeur par défaut, puis sélectionnez la classe VolumeSnapshotClass appropriée. Cliquez ensuite sur Créer.



5. La création du snapshot de la demande de volume persistant est alors possible.

Créer une nouvelle machine virtuelle à partir du snapshot

1. Tout d'abord, restaurez la copie Snapshot dans un nouveau volume persistant. Accédez à stockage > Volumesnapshots, cliquez sur les points de suspension situés à côté du Snapshot que vous souhaitez restaurer, puis cliquez sur Restaurer en tant que nouveau volume de volume persistant.
2. Entrez les détails du nouveau PVC et cliquez sur Restaurer. Cela crée un nouveau PVC.



3. Ensuite, créez une nouvelle machine virtuelle à partir de ce volume persistant. Accédez à charges de travail > virtualisation > machines virtuelles, puis cliquez sur Créer > avec YAML.

4. Dans la section spec > template > spec > volumes, spécifiez le nouveau PVC créé à partir de Snapshot au lieu du disque conteneur. Fournir tous les autres détails relatifs à la nouvelle machine virtuelle selon vos besoins.

```
- name: rootdisk
  persistentVolumeClaim:
    claimName: rhel8-short-frog-rootdisk-28dvb-snapshot-restore
```

5. Cliquez sur Créer pour créer la nouvelle machine virtuelle.
6. Une fois la machine virtuelle créée, accédez-y et vérifiez que la nouvelle machine virtuelle possède le même état que celle de la machine virtuelle dont le volume de demande de volume persistant a été utilisé pour créer le Snapshot au moment de la création du Snapshot.

Flux de travail : Red Hat OpenShift Virtualization avec NetApp ONTAP

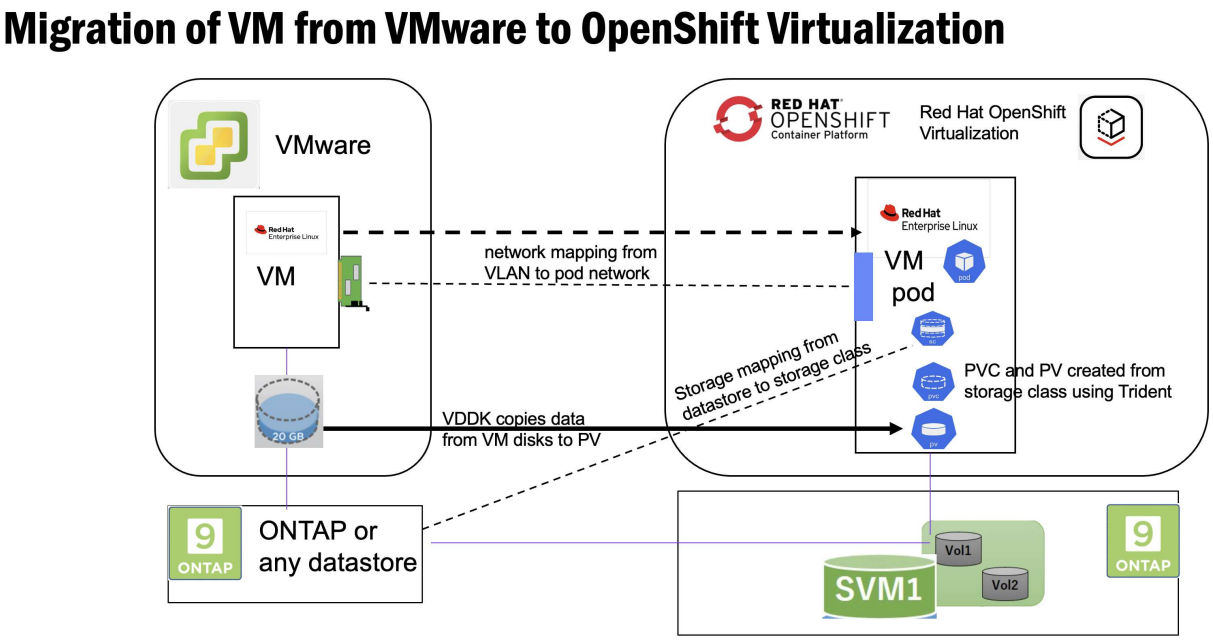
Migration de VM de VMware vers OpenShift Virtualization à l'aide de migration Toolkit pour la virtualisation

Dans cette section, nous allons voir comment utiliser le kit d'outils de migration pour la virtualisation (MTV) pour migrer des machines virtuelles de VMware vers OpenShift Virtualization s'exécutant sur OpenShift Container Platform et intégré avec le stockage NetApp ONTAP à l'aide d'Astra Trident.

La vidéo suivante montre une démonstration de la migration d'une machine virtuelle RHEL de VMware vers OpenShift Virtualization à l'aide d'ontap-san pour le stockage persistant.

Utilisation de Red Hat MTV pour migrer des machines virtuelles vers OpenShift Virtualization avec le stockage NetApp ONTAP

Le schéma suivant présente une vue d'ensemble de la migration d'une machine virtuelle de VMware vers Red Hat OpenShift Virtualization.



Conditions préalables pour l'exemple de migration

Sur VMware

- Une machine virtuelle RHEL 9 utilisant rhel 9.3 avec les configurations suivantes a été installée :
  - CPU: 2, mémoire: 20 Go, disque dur: 20 Go
  - informations d'identification de l'utilisateur : informations d'identification de l'utilisateur root et d'un utilisateur admin
- Une fois la machine virtuelle prête, le serveur postgresql a été installé.
  - le serveur postgresql a été démarré et activé pour démarrer au démarrage

```
systemctl start postgresql.service`
systemctl enable postgresql.service
The above command ensures that the server can start in the VM in
OpenShift Virtualization after migration
```

  - Ajout de 2 bases de données, 1 table et 1 ligne dans la table. Reportez-vous à "ici" Pour obtenir des instructions sur l'installation du serveur postgresql sur RHEL et la création d'entrées de base de données et de table.

> Assurez-vous que vous démarrez le serveur postgresql et que le service démarre au démarrage.

Sur OpenShift Cluster

Les installations suivantes ont été effectuées avant l'installation de MTV :

- OpenShift Cluster 4.13.34
- "ASTRA Trident 23.10"
- Chemins d'accès multiples sur les nœuds de cluster activés pour iSCSI (pour la classe de stockage ontap-san). Consultez le yaml fourni pour créer un jeu de démons qui active iSCSI sur chaque nœud du cluster.
- Système back-end Trident et classe de stockage pour SAN ONTAP utilisant iSCSI. Consultez les fichiers yaml fournis pour le back-end trident et la classe de stockage.
- "Virtualisation OpenShift"

Pour installer iscsi et le multipath sur les nœuds OpenShift Cluster, utilisez le fichier yaml indiqué ci-dessous
Préparation des nœuds de cluster pour iSCSI

```yaml
apiVersion: apps/v1
kind: DaemonSet
metadata:
  namespace: trident
  name: trident-iscsi-init
  labels:
    name: trident-iscsi-init
spec:
```

```
  selector:
    matchLabels:
      name: trident-iscsi-init
  template:
    metadata:
      labels:
        name: trident-iscsi-init
    spec:
      hostNetwork: true
      serviceAccount: trident-node-linux
      initContainers:
      - name: init-node
        command:
          - nsenter
          - --mount=/proc/1/ns/mnt
          - --
          - sh
          - -c
        args: ["$(STARTUP_SCRIPT)"]
        image: alpine:3.7
        env:
        - name: STARTUP_SCRIPT
          value: |
            #! /bin/bash
            sudo yum install -y lsscsi iscsi-initiator-utils sg3_utils
device-mapper-multipath
            rpm -q iscsi-initiator-utils
            sudo sed -i 's/^\(node.session.scan\).*/\1 = manual/'
/etc/iscsi/iscsid.conf
            cat /etc/iscsi/initiatorname.iscsi
            sudo mpathconf --enable --with_multipathd y --find_multipaths
n
            sudo systemctl enable --now iscsid multipathd
            sudo systemctl enable --now iscsi
        securityContext:
          privileged: true
      hostPID: true
      containers:
      - name: wait
        image: k8s.gcr.io/pause:3.1
      hostPID: true
      hostNetwork: true
      tolerations:
      - effect: NoSchedule
        key: node-role.kubernetes.io/master
  updateStrategy:
```

```
    type: RollingUpdate
```

Utilisez le fichier yaml suivant pour créer la configuration back-end trident pour l'utilisation du stockage san ONTAP

Back-end Trident pour iSCSI

```
apiVersion: v1
kind: Secret
metadata:
  name: backend-tbc-ontap-san-secret
type: Opaque
stringData:
  username: <username>
  password: <password>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: ontap-san
spec:
  version: 1
  storageDriverName: ontap-san
  managementLIF: <management LIF>
  backendName: ontap-san
  svm: <SVM name>
  credentials:
    name: backend-tbc-ontap-san-secret
```

Utilisez le fichier yaml suivant pour créer la configuration de classe de stockage trident pour l'utilisation du stockage san ONTAP

Classe de stockage Trident pour iSCSI

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: ontap-san
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-san"
  media: "ssd"
  provisioningType: "thin"
  snapshots: "true"
allowVolumeExpansion: true
```

Installer MTV

Vous pouvez maintenant installer le kit d'outils de migration pour la virtualisation (MTV). Reportez-vous aux instructions fournies "ici" pour obtenir de l'aide lors de l'installation.

L'interface utilisateur MTV (migration Toolkit for Virtualization) est intégrée à la console Web OpenShift. Vous pouvez vous référer "ici" pour commencer à utiliser l'interface utilisateur pour différentes tâches.

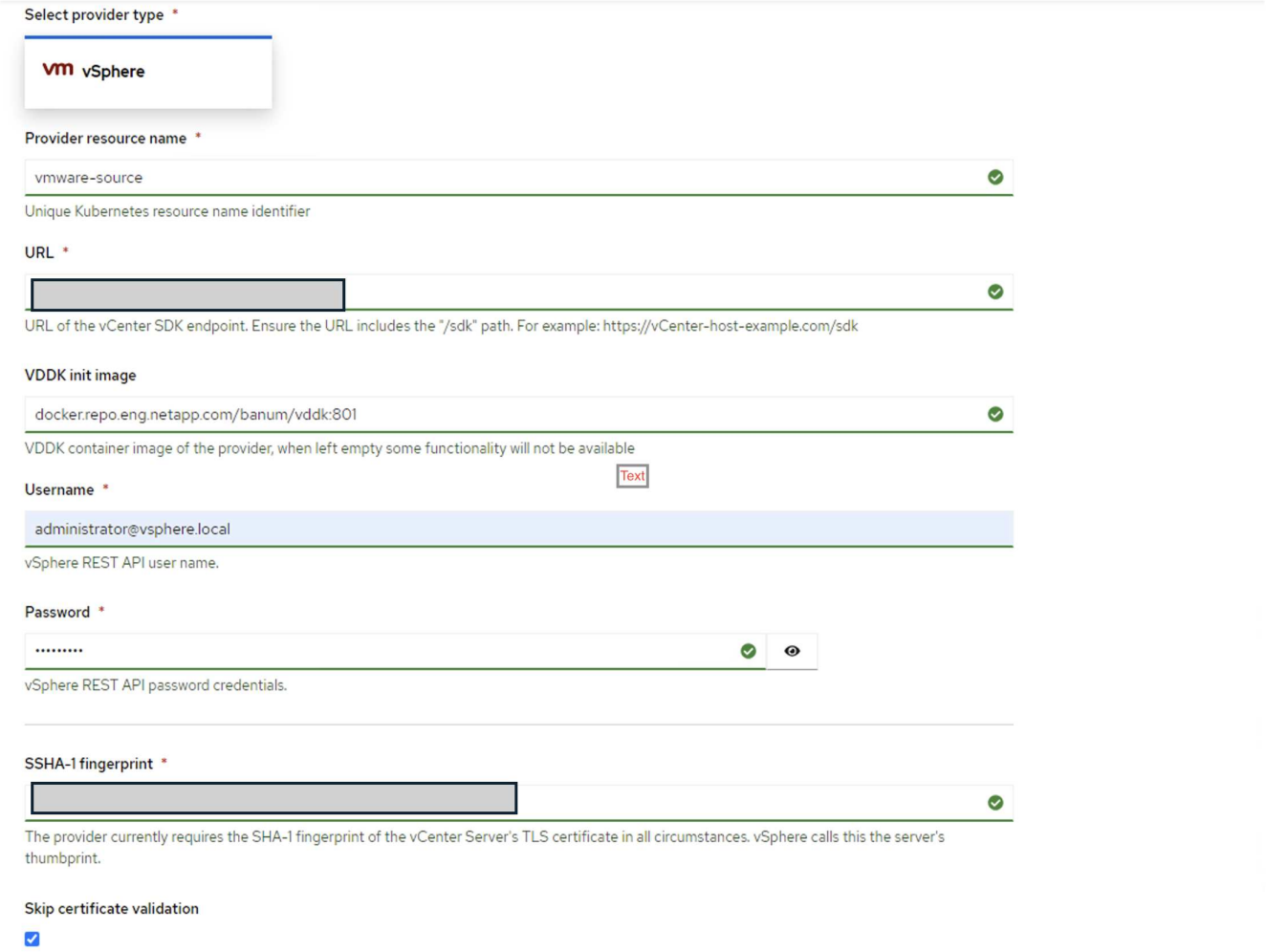
Créer un fournisseur source

Pour migrer la machine virtuelle RHEL de VMware vers OpenShift Virtualization, vous devez d'abord créer le fournisseur source pour VMware. Reportez-vous aux instructions "ici" pour créer le fournisseur source.

Vous avez besoin des éléments suivants pour créer votre fournisseur source VMware :

- url vCenter
- Informations d'identification vCenter
- Empreinte du serveur vCenter
- Image VDDK dans un référentiel

Exemple de création de fournisseur source :



> Le kit MTV (migration Toolkit for Virtualization) utilise le kit de développement de disques virtuels VMware (VDDK) pour accélérer le transfert des disques virtuels à partir de VMware vSphere. Par conséquent, la création d'une image VDDK, bien que facultative, est fortement recommandée.
> Pour utiliser cette fonction, vous téléchargez le kit de développement de disques virtuels VMware (VDDK), créez une image VDDK et envoyez l'image VDDK dans votre registre d'images.

Suivez les instructions fournies "ici" Pour créer et envoyer l'image VDDK vers un registre accessible à partir d'OpenShift Cluster.

Créer un fournisseur de destination

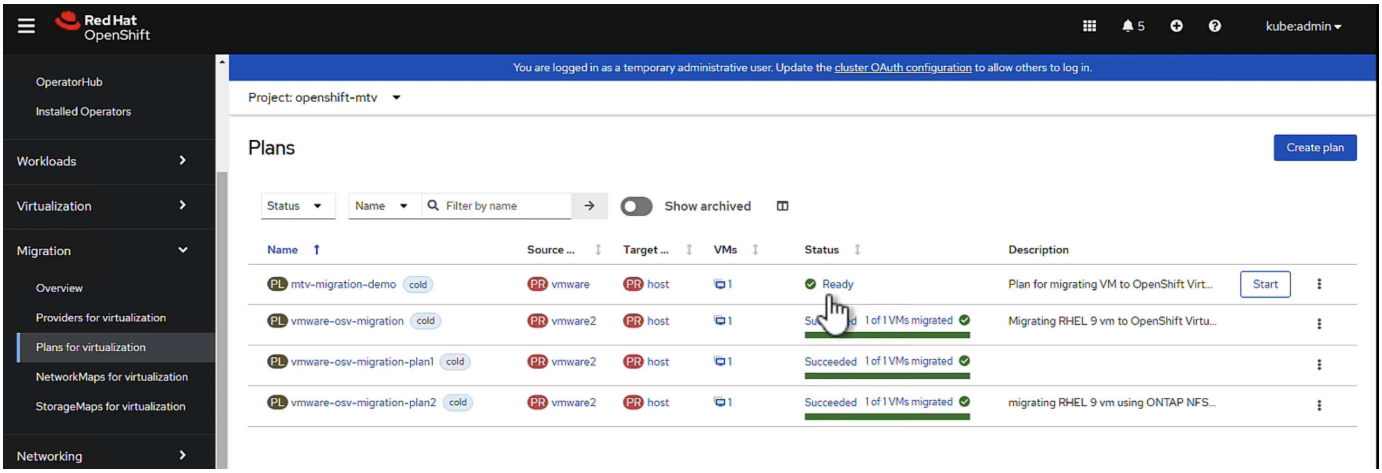
Le cluster hôte est automatiquement ajouté car le fournisseur de virtualisation OpenShift est le fournisseur source.

Créer un plan de migration

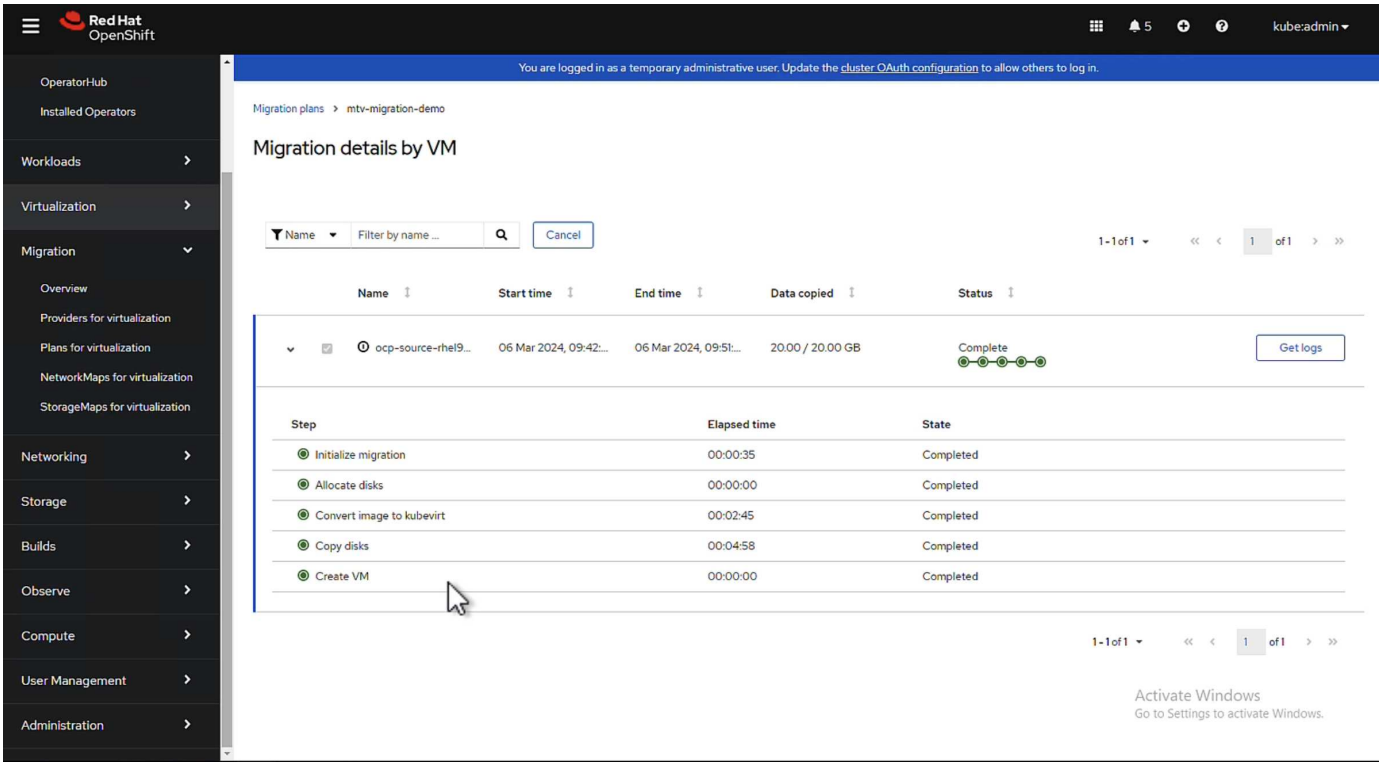
Suivez les instructions fournies "ici" pour créer un plan de migration.

Lors de la création d'un plan, vous devez créer les éléments suivants s'ils ne sont pas déjà créés :

- Mappage réseau pour mapper le réseau source au réseau cible.
- Un mappage de stockage pour mapper le datastore source à la classe de stockage cible. Pour cela, vous pouvez choisir la classe de stockage ontap-san.
  Une fois le plan de migration créé, le statut du plan doit indiquer **prêt** et vous devriez maintenant être en mesure de **démarrer** le plan.



Cliquez sur **Start** pour exécuter une séquence d'étapes pour terminer la migration de la machine virtuelle.



Lorsque toutes les étapes sont terminées, vous pouvez voir les VM migrés en cliquant sur les **machines virtuelles** sous **virtualisation** dans le menu de navigation de gauche.
Des instructions pour accéder aux machines virtuelles sont fournies "ici".

Vous pouvez vous connecter à la machine virtuelle et vérifier le contenu des bases de données postpostgresql. Les bases de données, les tables et les entrées de la table doivent être identiques à celles

créées sur la machine virtuelle source.

Protection des données pour OpenShift Virtualization

Protection des données pour les VM dans OpenShift Virtualization à l'aide d'OpenShift API for Data protection (OADP)

Auteur: Banu Sundhar, NetApp

Cette section du document de référence fournit des informations détaillées sur la création de sauvegardes de machines virtuelles à l'aide d'OpenShift API for Data protection (OADP) avec Velero sur NetApp ONTAP S3 ou NetApp StorageGRID S3. Les sauvegardes des volumes persistants (persistent volumes) des disques de la machine virtuelle sont créées à l'aide des snapshots CSI Astra Trident.

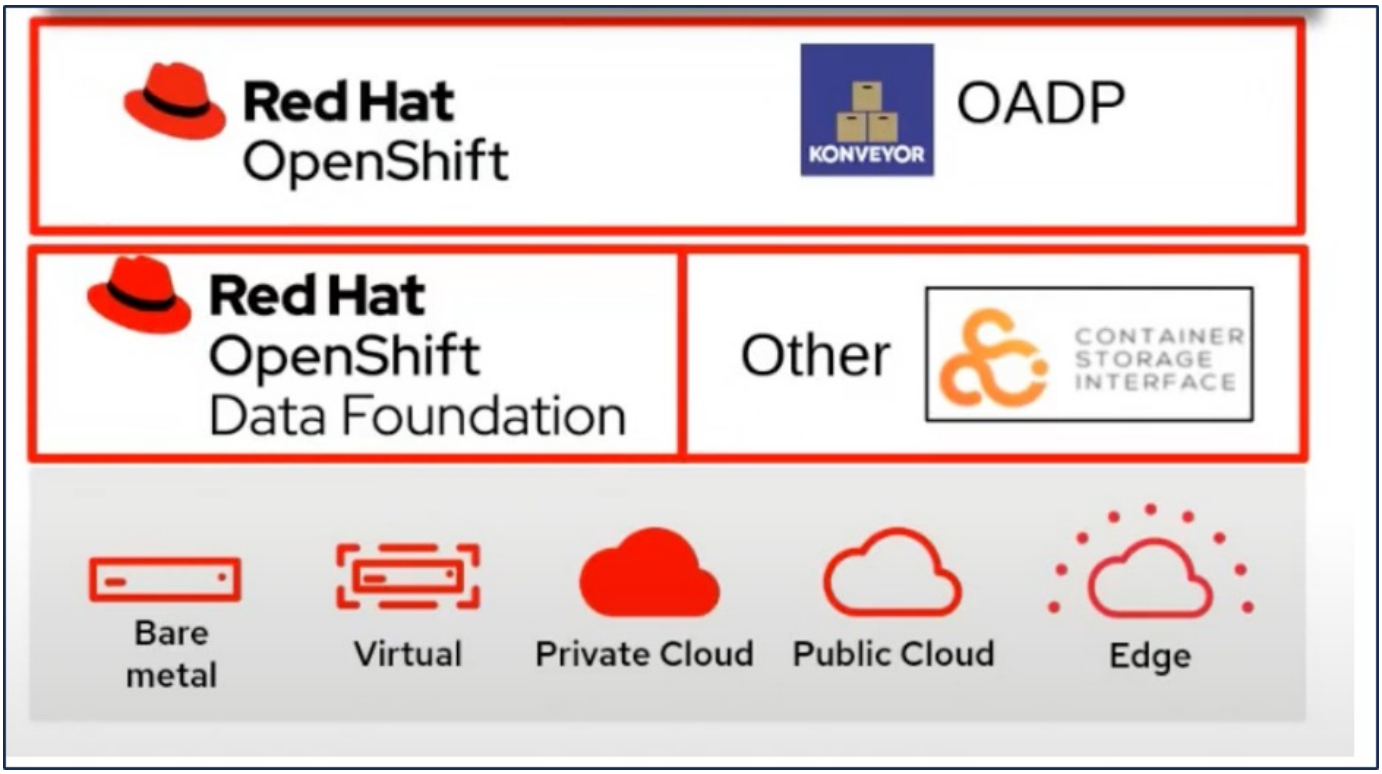
Les machines virtuelles de l'environnement OpenShift Virtualization sont des applications conteneurisées qui s'exécutent dans les nœuds workers de votre plateforme OpenShift Container. Il est important de protéger les métadonnées des machines virtuelles ainsi que les disques persistants des machines virtuelles, afin que vous puissiez les restaurer en cas de perte ou de corruption.

Les disques persistants des VM de virtualisation OpenShift peuvent être pris en charge par le stockage ONTAP intégré au cluster OpenShift à l'aide de "ASTRA Trident CSI". Dans cette section, nous utilisons "OpenShift API pour la protection des données (OADP)" Pour effectuer la sauvegarde des machines virtuelles, y compris leurs volumes de données vers

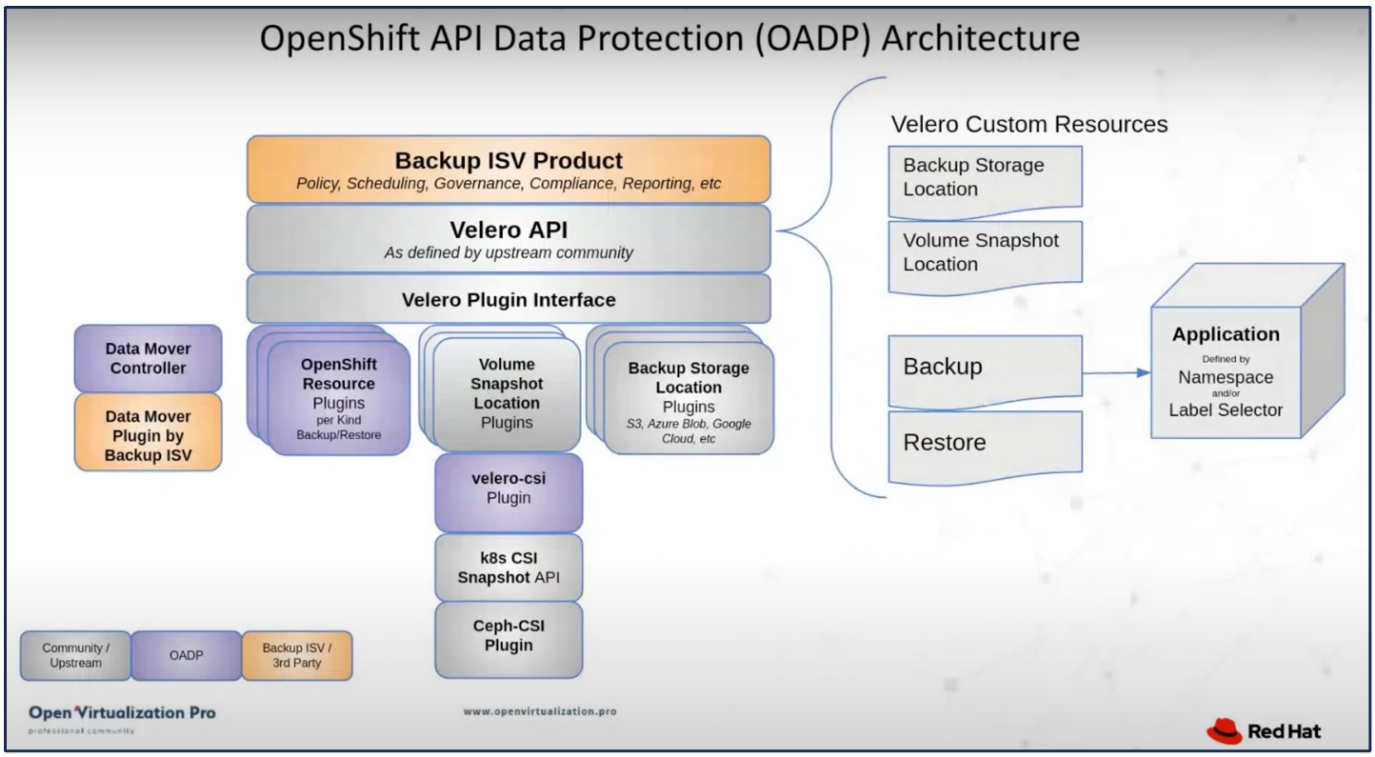
- Stockage objet ONTAP
- StorageGRID

Nous restaurons ensuite les données à partir de la sauvegarde si nécessaire.

OADP permet la sauvegarde, la restauration et la reprise après incident des applications sur un cluster OpenShift. Les données protégées avec OADP incluent les objets de ressource Kubernetes, les volumes persistants et les images internes.



Red Hat OpenShift a mis à profit les solutions développées par les communautés OpenSource pour la protection des données. "Velero" Est un outil open source qui permet de sauvegarder et de restaurer en toute sécurité, d'effectuer une reprise d'activité et de migrer les ressources de cluster Kubernetes et les volumes persistants. Pour utiliser Velero facilement, OpenShift a développé l'opérateur OADP et le plug-in Velero pour s'intégrer aux pilotes de stockage CSI. Les principales API OADP exposées sont basées sur les API Velero. Après l'installation de l'opérateur OADP et sa configuration, les opérations de sauvegarde/restauration qui peuvent être effectuées sont basées sur les opérations exposées par l'API Velero.



OADP 1.3 est disponible sur le hub opérateur d'OpenShift cluster 4.12 et versions ultérieures. Il est doté d'un

Data Mover intégré qui peut déplacer les instantanés de volume CSI vers un magasin d'objets distant. Ces fonctionnalités assurent la portabilité et la durabilité en déplaçant les snapshots vers un emplacement de stockage objet pendant la sauvegarde. Les snapshots sont ensuite disponibles pour la restauration après un incident.

Les versions suivantes des différents composants utilisés dans les exemples de cette section

- OpenShift Cluster 4.14
- OpenShift Virtualization installé via OperatorOpenShift Virtualization Operator fourni par Red Hat
- Opérateur OADP 1.13 fourni par Red Hat
- Velero CLI 1.13 pour Linux
- ASTRA Trident 24.02
- ONTAP 9.12

"ASTRA Trident CSI"
"OpenShift API pour la protection des données (OADP)"
"Velero"

Installation de l'opérateur OpenShift API for Data protection (OADP)

Prérequis

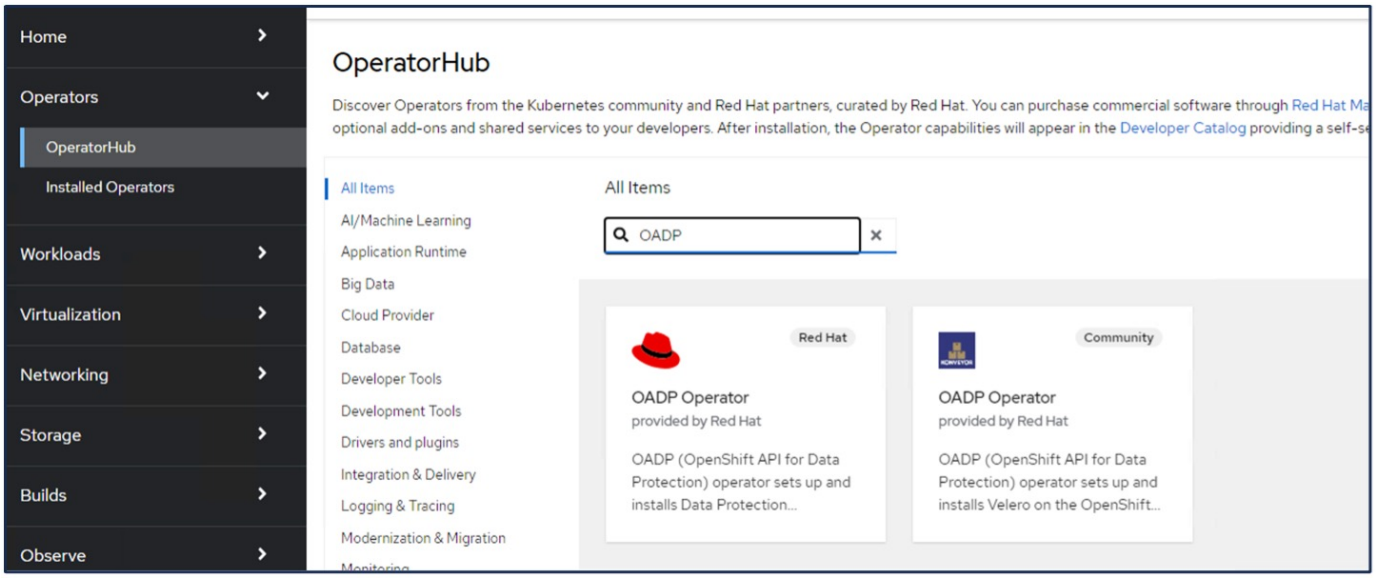
- Cluster Red Hat OpenShift (version ultérieure à la version 4.12) installé sur une infrastructure sans système d'exploitation avec des nœuds worker RHCOS
- Un cluster NetApp ONTAP intégré au cluster via Astra Trident
- Un système back-end Trident configuré avec un SVM sur le cluster ONTAP
- Classe de stockage configurée sur le cluster OpenShift avec Astra Trident en tant que mécanisme de provisionnement
- Classe Snapshot Trident créée sur le cluster
- L'accès cluster-admin au cluster Red Hat OpenShift
- Accès au cluster NetApp ONTAP par administrateur
- L'opérateur OpenShift Virtualization est installé et configuré
- VM déployées dans un espace de noms sur OpenShift Virtualization
- Une station de travail d'administration avec des outils tridentctl et oc installés et ajoutés à $PATH

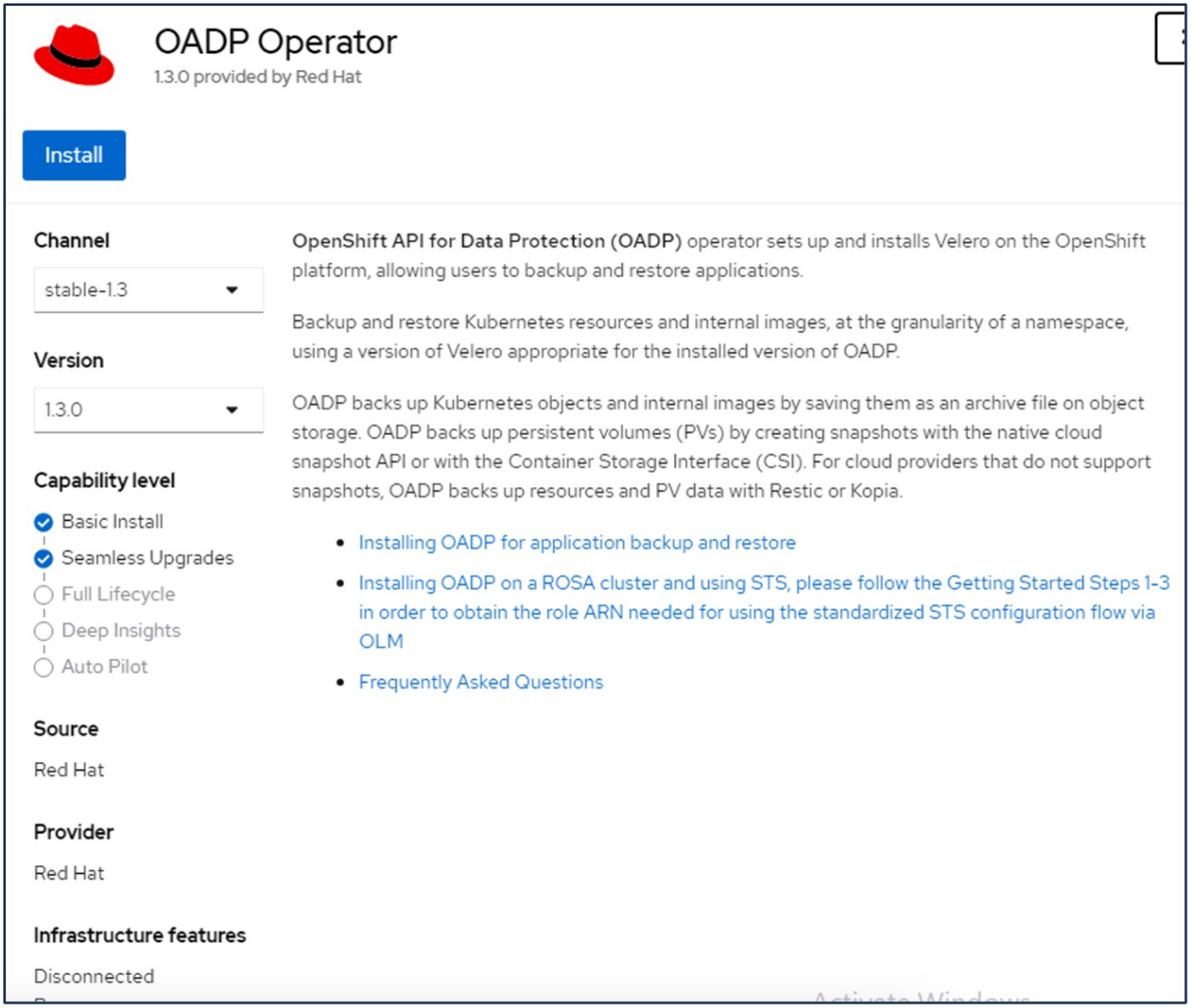
> Si vous souhaitez effectuer une sauvegarde d'une machine virtuelle alors qu'elle est en cours d'exécution, vous devez installer l'agent invité QEMU sur cette machine virtuelle. Si vous installez la machine virtuelle à l'aide d'un modèle existant, l'agent QEMU est automatiquement installé. QEMU permet à l'agent invité de suspendre les données en vol dans le système d'exploitation invité pendant le processus de snapshot et d'éviter toute corruption potentielle des données. Si QEMU n'est pas installé, vous pouvez arrêter la machine virtuelle avant d'effectuer une sauvegarde.

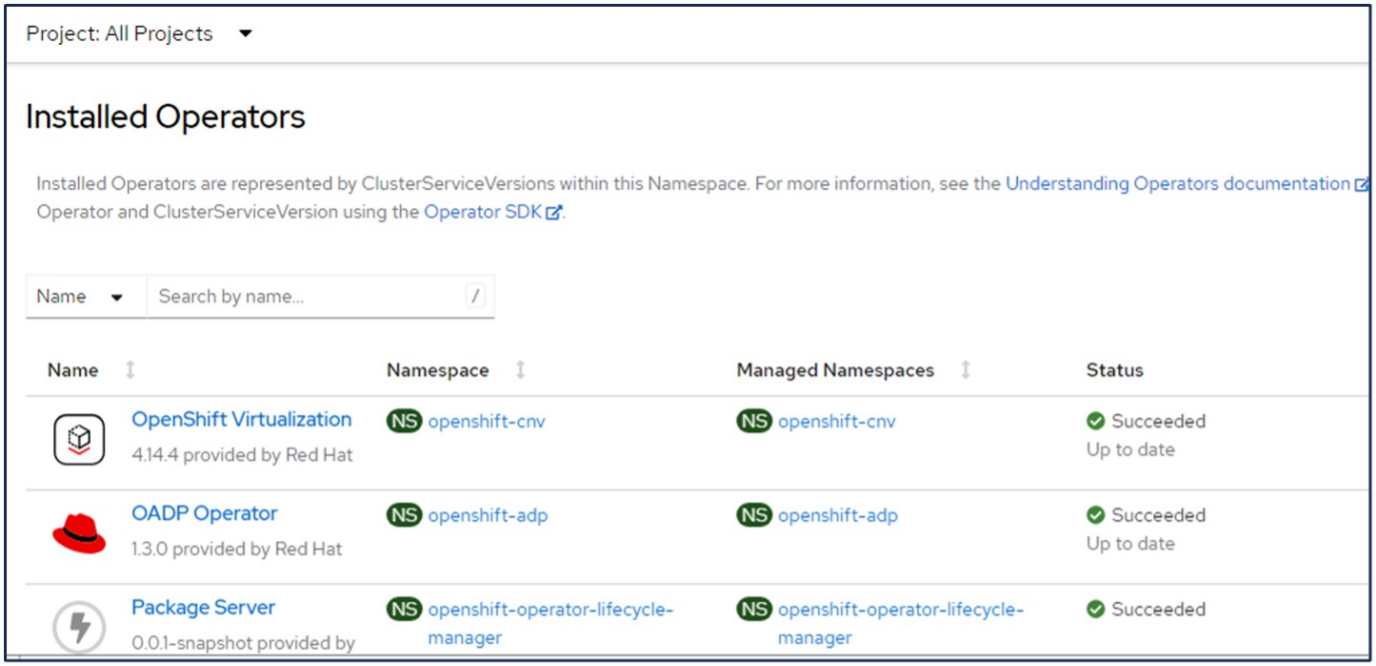
Procédure d'installation de l'opérateur OADP

1. Accédez au hub opérateur du cluster et sélectionnez opérateur OADP Red Hat. Dans la page installer, utilisez toutes les sélections par défaut et cliquez sur installer. Sur la page suivante, utilisez à nouveau toutes les valeurs par défaut et cliquez sur installer. L'opérateur OADP sera installé dans l'espace de noms

openshift-adp.







Conditions préalables pour la configuration Velero avec les détails de ONTAP S3

Une fois l'installation de l'opérateur réussie, configurez l'instance de Velero.
Velero peut être configuré pour utiliser le stockage objet compatible S3. Configurez ONTAP S3 à l'aide des procédures indiquées dans le "Section gestion du stockage objet de la documentation ONTAP". Pour l'intégration à Velero, vous aurez besoin des informations suivantes de votre configuration ONTAP S3.

- Une interface logique (LIF) qui peut être utilisée pour accéder à S3
- Informations d'identification de l'utilisateur pour accéder à S3, y compris la clé d'accès et la clé d'accès secrète
- Nom de compartiment dans S3 pour les sauvegardes avec des autorisations d'accès pour l'utilisateur
- Pour un accès sécurisé au stockage objet, le certificat TLS doit être installé sur le serveur de stockage objet.

Conditions préalables pour la configuration Velero avec les détails de StorageGRID S3

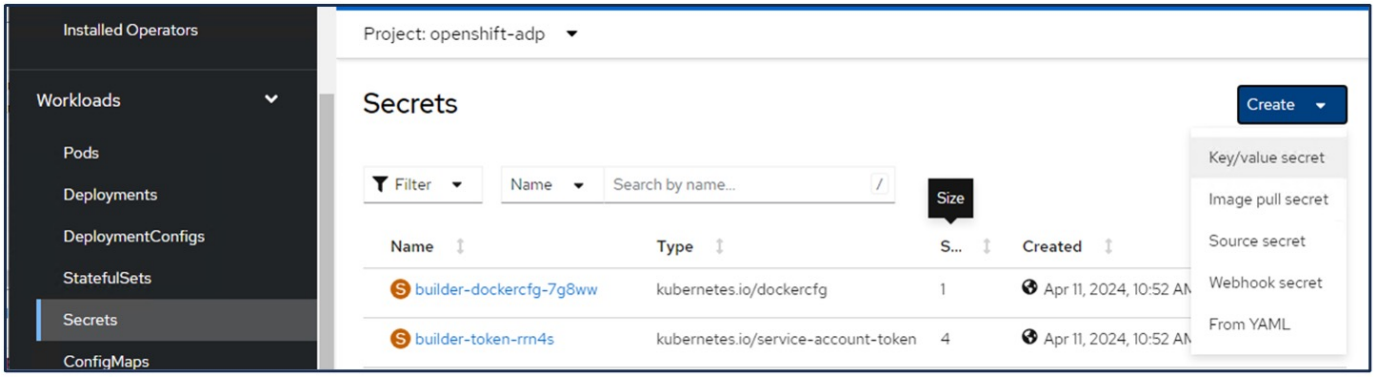
Velero peut être configuré pour utiliser le stockage objet compatible S3. Vous pouvez configurer StorageGRID S3 à l'aide des procédures indiquées dans le "Documentation StorageGRID". Pour l'intégration à Velero, vous aurez besoin des informations suivantes de votre configuration StorageGRID S3.

- Terminal pouvant être utilisé pour accéder à S3
- Informations d'identification de l'utilisateur pour accéder à S3, y compris la clé d'accès et la clé d'accès secrète
- Nom de compartiment dans S3 pour les sauvegardes avec des autorisations d'accès pour l'utilisateur
- Pour un accès sécurisé au stockage objet, le certificat TLS doit être installé sur le serveur de stockage objet.

Procédure de configuration de Velero

- Commencez par créer un secret pour les informations d'identification d'un utilisateur ONTAP S3 ou pour les informations d'identification d'un utilisateur de locataire StorageGRID. Ceci sera utilisé pour configurer

Velero ultérieurement. Vous pouvez créer un secret à partir de l'interface de ligne de commande ou de la console Web.
Pour créer un secret à partir de la console Web, sélectionnez secrets, puis cliquez sur clé/valeur Secret. Indiquez les valeurs pour le nom, la clé et la valeur des informations d'identification, comme indiqué. Assurez-vous d'utiliser l'ID de clé d'accès et la clé d'accès secrète de votre utilisateur S3. Nommez le secret de manière appropriée. Dans l'exemple ci-dessous, un code secret associé aux informations d'identification d'utilisateur ONTAP S3 nommées identifiants ontap-s3 est créé.



Project: openshift-adp

Create key/value secret

Key/value secrets let you inject sensitive data into your application as files or environment variables.

Secret name *

cloud-credentials

Unique name of the new secret.

Key *

cloud

Value

Browse...

Drag and drop file with your value here or browse to upload it.

```
[default]
aws_access_key_id=<Access Key Id of S3 user>
aws_secret_access_key=<Secret Access Key of S3 user>
```

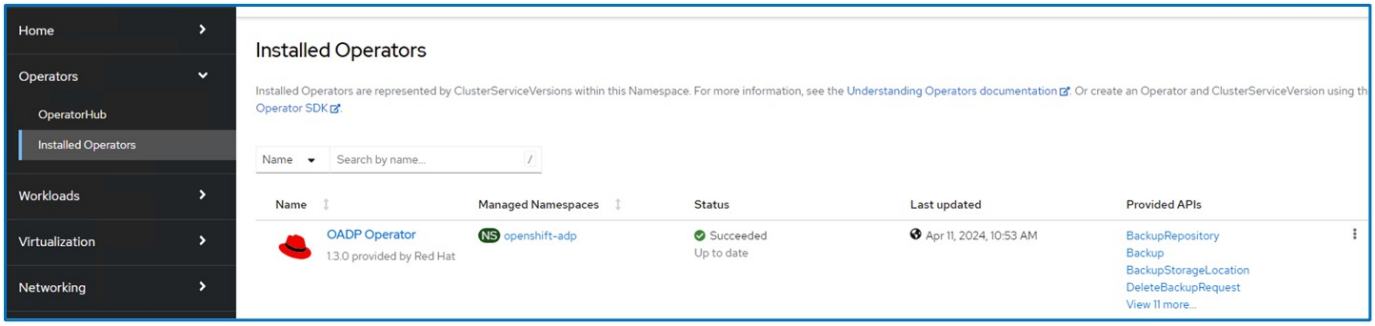
Add key/value

Create Cancel

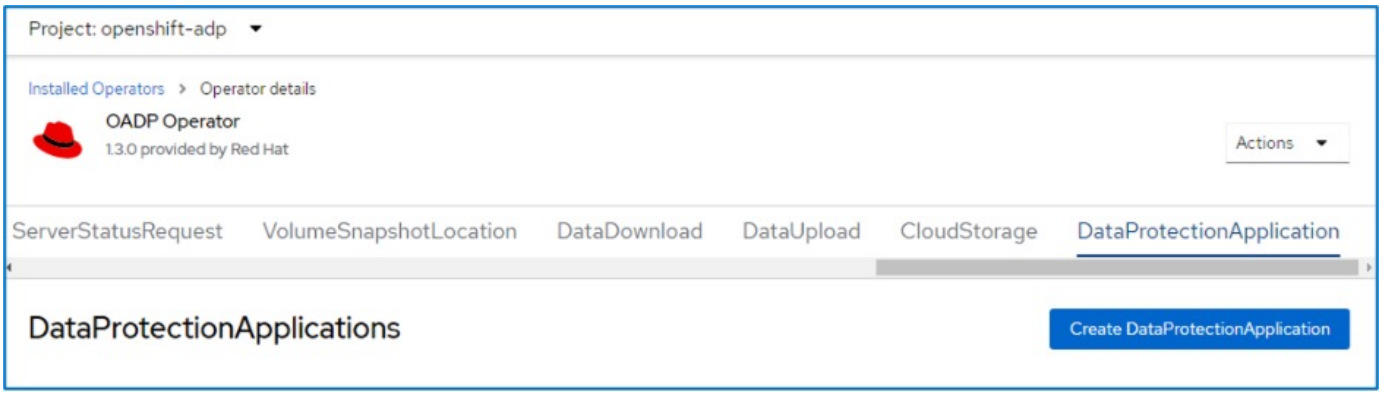
Pour créer un secret nommé sg-s3-credentials à partir de l'interface de ligne de commande, vous pouvez utiliser la commande suivante.

```
# oc create secret generic cloud-credentials --namespace openshift-adp --
from-file cloud=cloud-credentials.txt

credentials.txt file contains the Access Key Id and the Secret Access Key of
the S3 user in the following format:

[default]
aws_access_key_id=<Access Key Id of S3 user>
aws_secret_access_key=<Secret Access Key of S3 user>
```

- Ensuite, pour configurer Velero, sélectionnez opérateurs installés dans l'élément de menu sous opérateurs, cliquez sur opérateur OADP, puis sélectionnez l'onglet DataProtectionapplication.



Cliquez sur Create DataProtectionApplication. Dans la vue formulaire, indiquez un nom pour l'application Dataprotection ou utilisez le nom par défaut.



Allez maintenant à la vue YAML et remplacez les informations de spécification comme indiqué dans les exemples de fichier yaml ci-dessous.

Exemple de fichier yaml pour la configuration de Velero avec ONTAP S3 comme emplacement de sauvegarde

```
spec:
  backupLocations:
    - velero:
        config:
          insecureSkipTLSVerify: 'true' ->use this for https communication
with ONTAP S3
          profile: default
          region: us-east
          s3ForcePathStyle: 'True' ->This allows use of IP in s3URL
          s3Url: 'https://10.xx.xx.xx' ->Ensure TLS certificate for S3 is
configured
        credential:
          key: cloud
          name: ontap-s3-credentials ->previously created secret
        default: true
        objectStorage:
          bucket: velero ->Your bucket name previously created in S3 for
backups
          prefix: demobackup ->The folder that will be created in the
bucket
        provider: aws
  configuration:
    nodeAgent:
      enable: true
      uploaderType: kopia
      #default Data Mover uses Kopia to move snapshots to Object Storage
    velero:
      defaultPlugins:
        - csi ->Add this plugin
        - openshift
        - aws
        - kubevirt ->Add this plugin
```

Exemple de fichier yaml pour la configuration de Velero avec StorageGRID S3 comme backupLocation et snapshotLocation

```
spec:
  backupLocations:
    - velero:
        config:
          insecureSkipTLSVerify: 'true'
          profile: default
          region: us-east-1 ->region of your StorageGrid system
          s3ForcePathStyle: 'True'
          s3Url: 'https://172.21.254.25:10443' ->the IP used to access S3
        credential:
          key: cloud
          name: sg-s3-credentials ->secret created earlier
        default: true
        objectStorage:
          bucket: velero
          prefix: demobackup
        provider: aws
  configuration:
    nodeAgent:
      enable: true
      uploaderType: kopia
    velero:
      defaultPlugins:
        - csi
        - openshift
        - aws
        - kubevirt
```

La section des spécifications du fichier yaml doit être configurée de manière appropriée pour les paramètres suivants, comme dans l'exemple ci-dessus

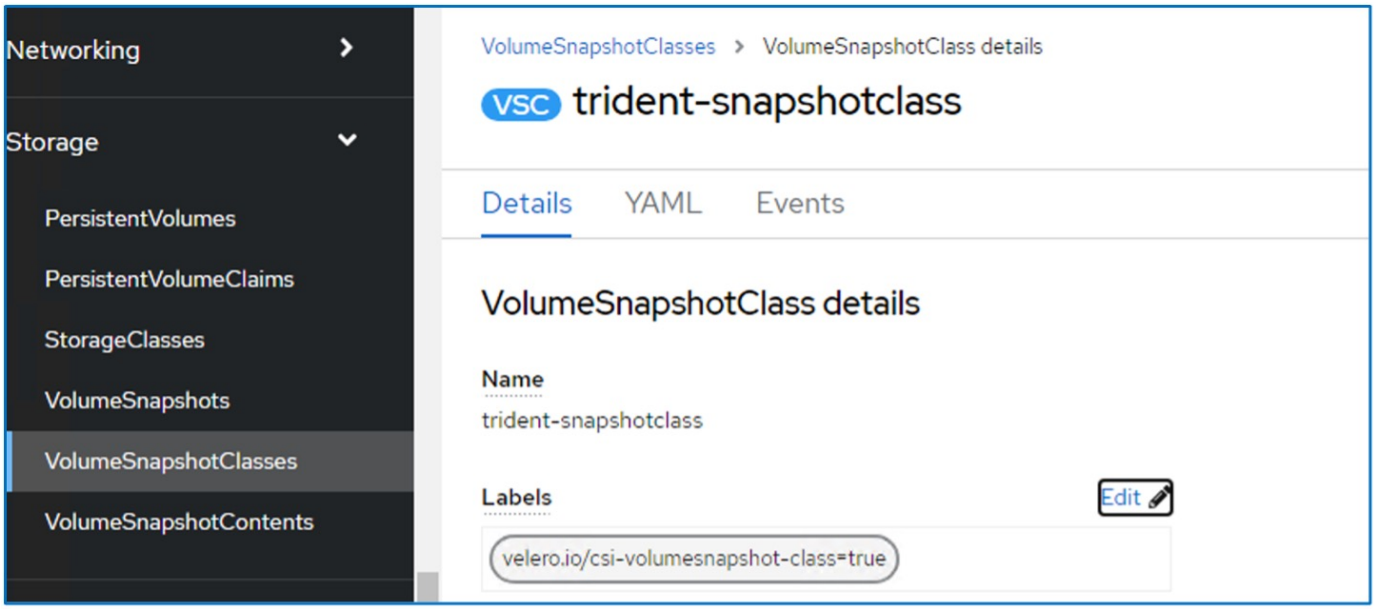
BackupLocation
ONTAP S3 ou StorageGRID S3 (avec ses informations d'identification et d'autres informations comme indiqué dans le yaml) est configuré comme emplacement de sauvegarde par défaut pour velero.

SnapshotLocation
Si vous utilisez des instantanés Container Storage interface (CSI), vous n'avez pas besoin de spécifier un emplacement de snapshot car vous allez créer un VolumeSnapshotClass CR pour enregistrer le pilote CSI. Dans cet exemple, vous utilisez Astra Trident CSI et vous avez déjà créé VolumeSnapShotClass CR à l'aide du pilote Trident CSI.

Activer le plug-in CSI

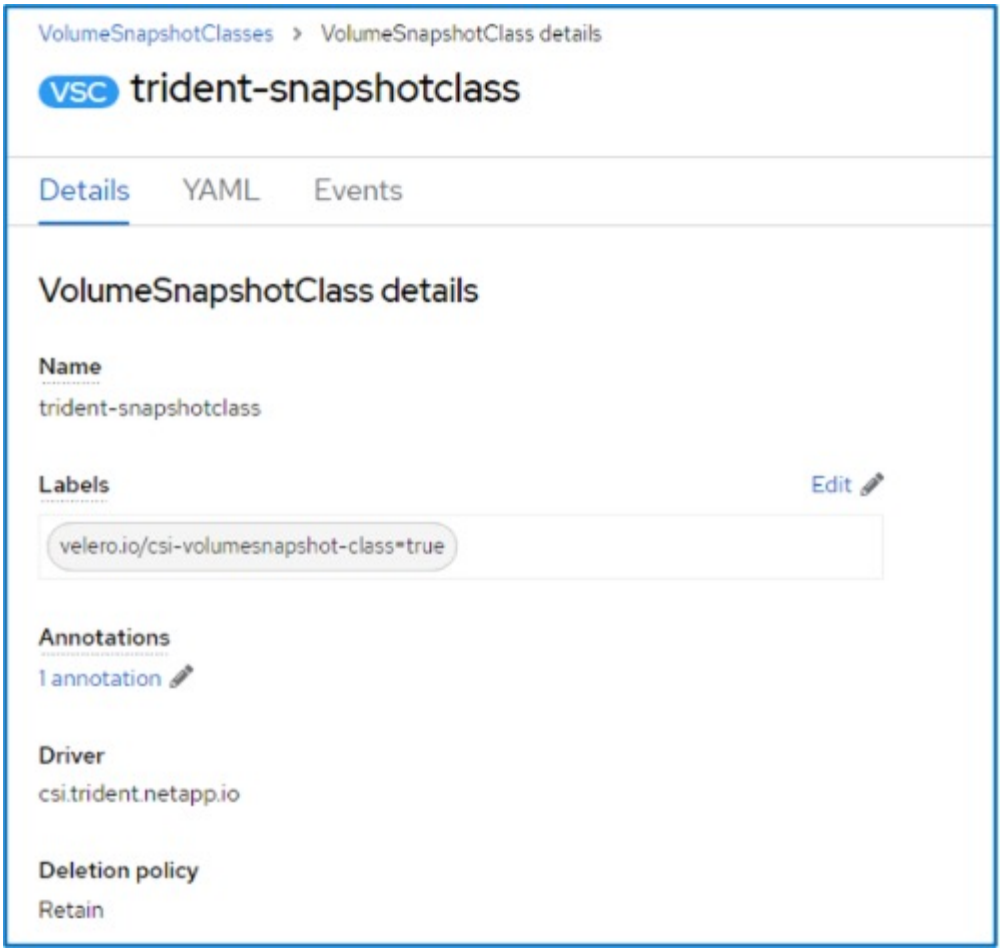
Ajoutez csi aux plug-ins par défaut de Velero pour sauvegarder les volumes persistants avec des snapshots CSI.
Les plug-ins Velero CSI, pour sauvegarder les PVC CSI, choisiront le VolumeSnapshotClass dans le cluster qui a le label **velero.io/csi-volumesnapshot-class** sur celui-ci. Pour cela

- Vous devez avoir créé la classe VolumeSnapshotClass.
- Modifiez le libellé de la classe trident-snapshotclass et définissez-le sur **velero.io/csi-volumesnapshot-class=true** comme indiqué ci-dessous.

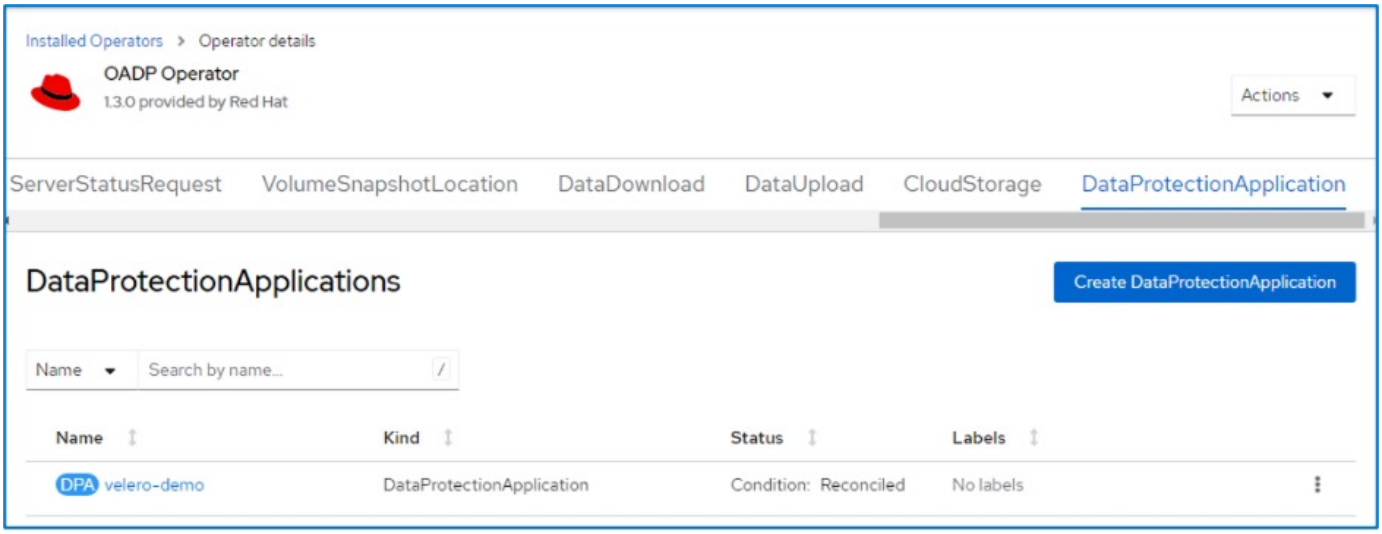


Assurez-vous que les snapshots peuvent persister même si les objets VolumeSnapshot sont supprimés. Pour ce faire, définissez la **deletionPolicy** à conserver. Si ce n'est pas le cas, la suppression d'un namespace perd complètement toutes les demandes de volume virtuels sauvegardées.

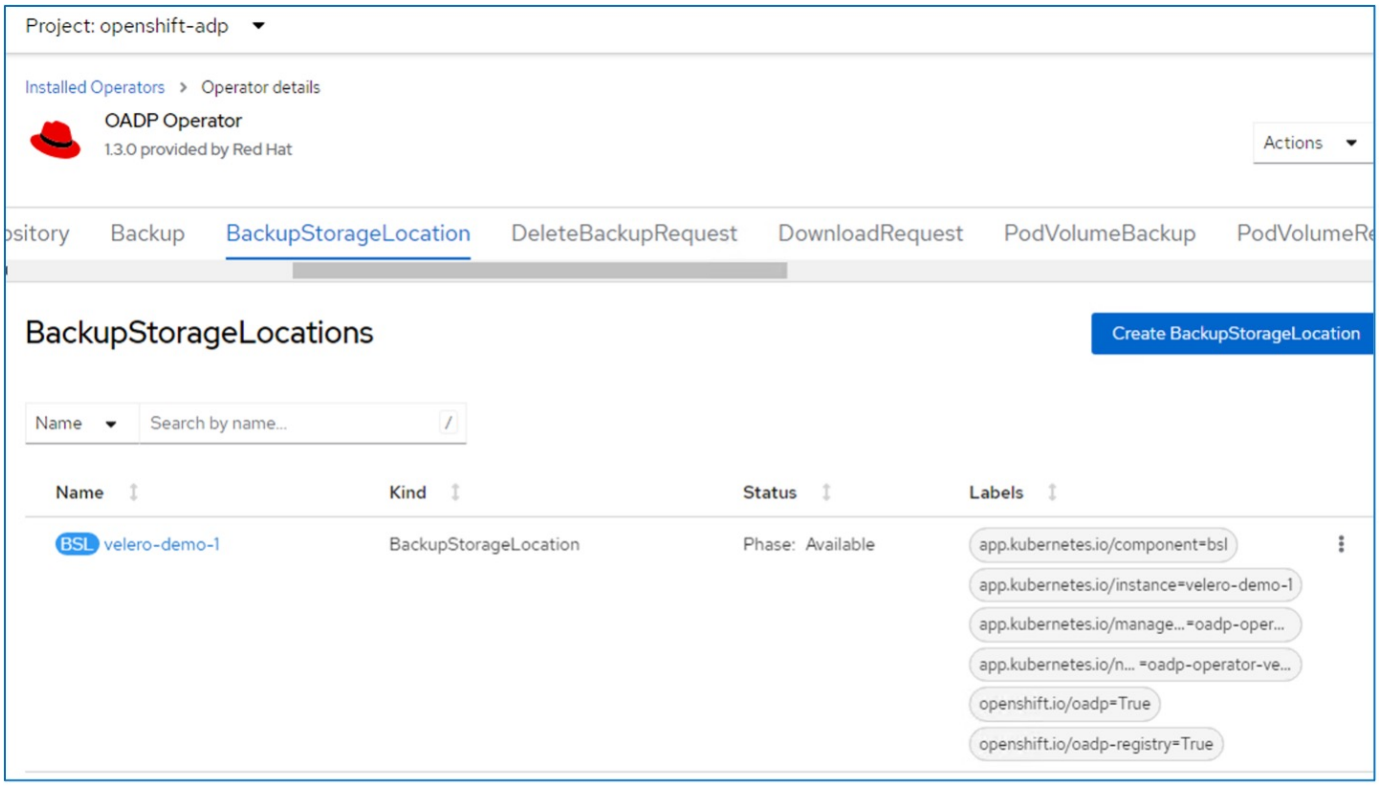
```
apiVersion: snapshot.storage.k8s.io/v1
kind: VolumeSnapshotClass
metadata:
  name: trident-snapshotclass
driver: csi.trident.netapp.io
deletionPolicy: Retain
```



Assurez-vous que l'application DataProtectionApplication est créée et qu'elle est en condition:réconciliée.



L'opérateur OADP va créer un BackupStorageLocation correspondant. Il sera utilisé lors de la création d'une sauvegarde.



Création d'une sauvegarde à la demande pour les machines virtuelles dans OpenShift Virtualization

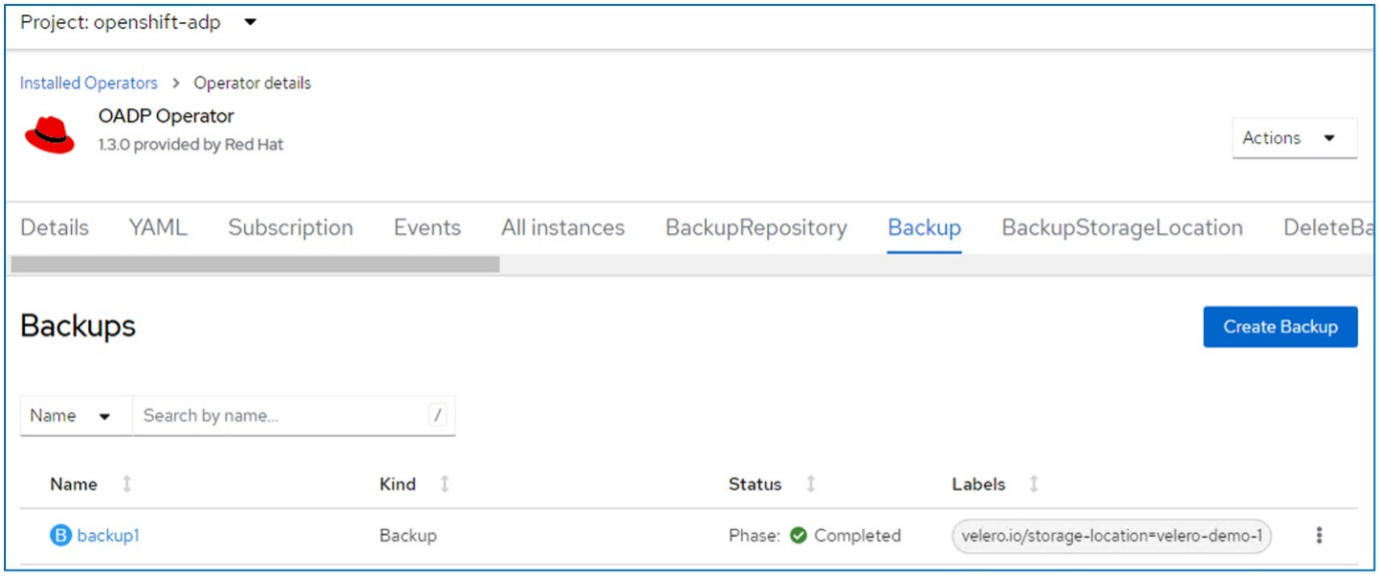
Étapes de création d'une sauvegarde d'une machine virtuelle

Pour créer une sauvegarde à la demande de l'ensemble de la machine virtuelle (métadonnées de la machine virtuelle et disques de la machine virtuelle), cliquez sur l'onglet **sauvegarde**. Cela crée une ressource personnalisée de sauvegarde (CR). Un exemple de yaml est fourni pour créer la CR de sauvegarde. En utilisant ce yaml, la machine virtuelle et ses disques dans l'espace de noms spécifié seront sauvegardés. Des paramètres supplémentaires peuvent être définis comme indiqué dans le "documentation".

Un instantané des volumes persistants qui soutiennent les disques sera créé par le CSI. Une sauvegarde de la machine virtuelle ainsi que l'instantané de ses disques sont créés et stockés dans l'emplacement de sauvegarde spécifié dans le yaml. La sauvegarde restera dans le système pendant 30 jours, comme spécifié dans le ttl.

```
apiVersion: velero.io/v1
kind: Backup
metadata:
  name: backup1
  namespace: openshift-adp
spec:
  includedNamespaces:
  - virtual-machines-demo
  snapshotVolumes: true
  storageLocation: velero-demo-1 -->this is the backupStorageLocation
previously created
                                    when Velero is configured.
  ttl: 720h0m0s
```

Une fois la sauvegarde terminée, sa phase s'affiche comme terminée.



Vous pouvez inspecter la sauvegarde dans le stockage objet à l'aide d'une application de navigateur S3. Le chemin de la sauvegarde s'affiche dans le compartiment configuré avec le préfixe nom (velero/démobilackup). Vous pouvez voir le contenu de la sauvegarde inclut les snapshots de volume, les journaux et d'autres métadonnées de la machine virtuelle.

> Dans StorageGRID, vous pouvez également utiliser la console S3 disponible dans le gestionnaire de locataires pour afficher les objets de sauvegarde.

Path: / demobackup/ backups/ backup1/

Name	Size	Type	Last Modified	Storage Class
..				
backup1.tar.gz	230.36 KB	GZ File	4/15/2024 10:26:29 PM	STANDARD
velero-backup.json	3.35 KB	JSON File	4/15/2024 10:26:29 PM	STANDARD
backup1-resource-list.json.gz	1.12 KB	GZ File	4/15/2024 10:26:29 PM	STANDARD
backup1-itemoperations.json.gz	600 bytes	GZ File	4/15/2024 10:26:28 PM	STANDARD
backup1-volumesnapshots.json.gz	29 bytes	GZ File	4/15/2024 10:26:28 PM	STANDARD
backup1-podvolumebackups.json.gz	29 bytes	GZ File	4/15/2024 10:26:28 PM	STANDARD
backup1-results.gz	49 bytes	GZ File	4/15/2024 10:26:28 PM	STANDARD
backup1-csi-volumesnapshotclasses.json.gz	426 bytes	GZ File	4/15/2024 10:26:28 PM	STANDARD
backup1-csi-volumesnapshotcontents.json.gz	1.43 KB	GZ File	4/15/2024 10:26:28 PM	STANDARD
backup1-csi-volumesnapshots.json.gz	1.34 KB	GZ File	4/15/2024 10:26:28 PM	STANDARD
backup1-logs.gz	13.49 KB	GZ File	4/15/2024 10:26:28 PM	STANDARD

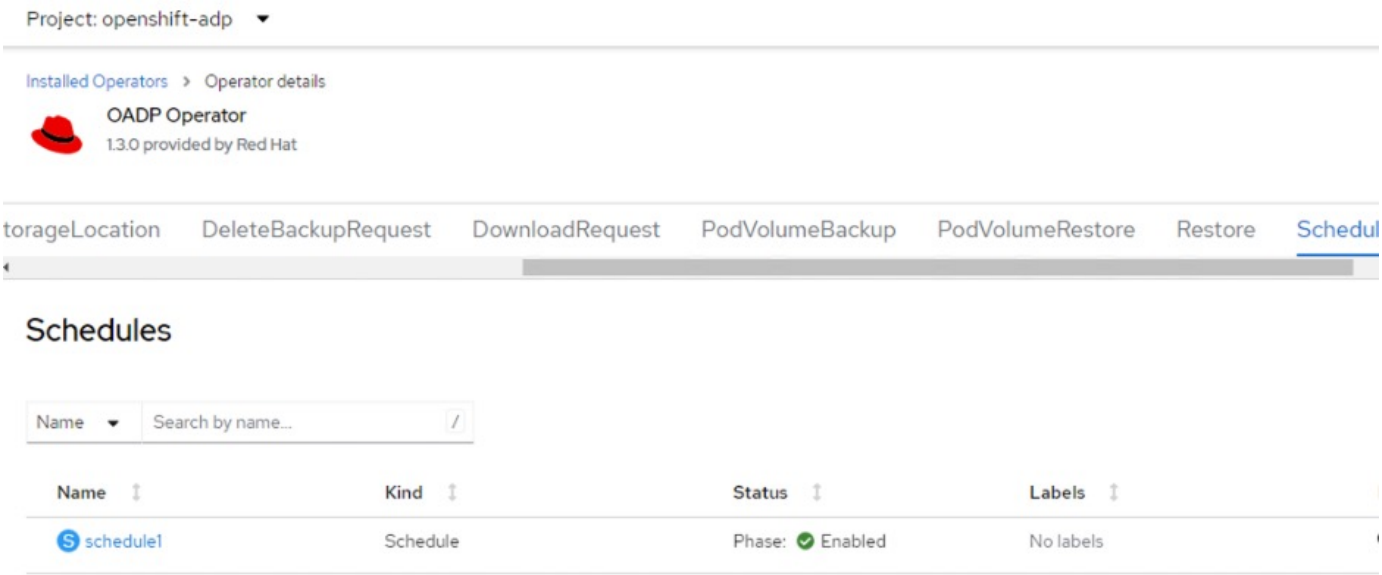
Création de sauvegardes planifiées pour les machines virtuelles dans OpenShift Virtualization

Pour créer des sauvegardes sur un planning, vous devez créer une demande de modification d'horaires. Le planning est simplement une expression cron qui vous permet de spécifier l'heure à laquelle vous souhaitez créer la sauvegarde. Un exemple de yaml pour créer une demande de modification d'horaire.

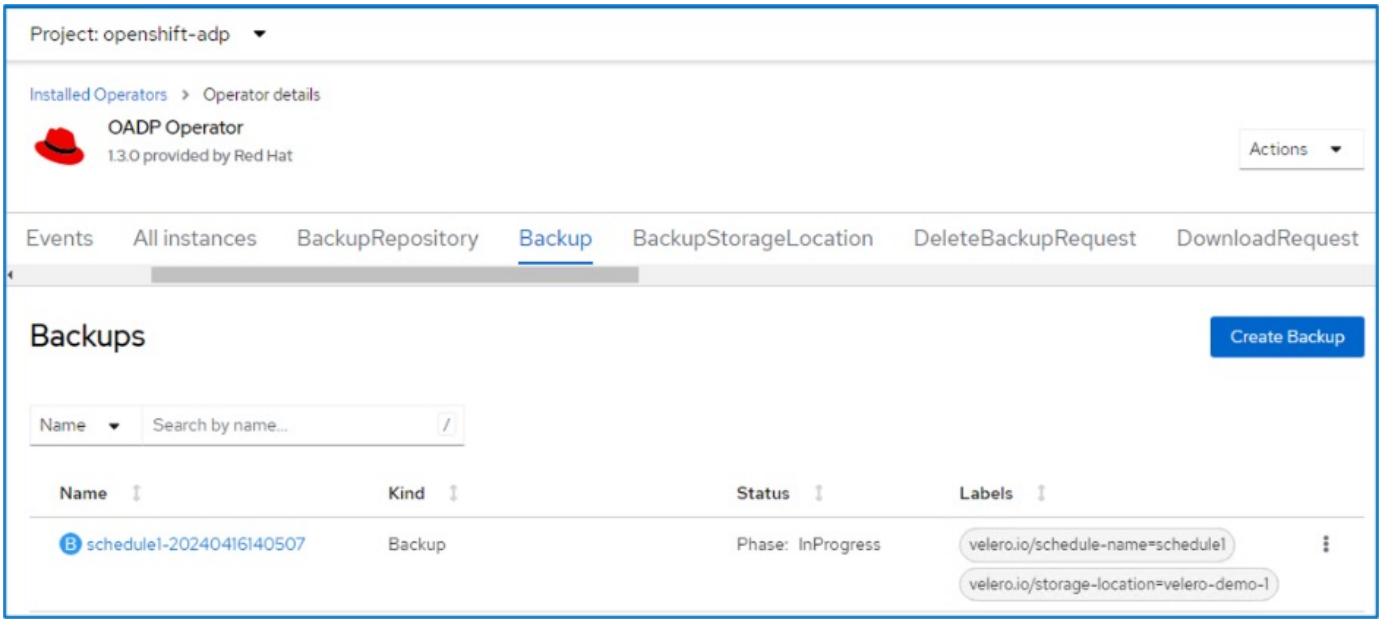
```
apiVersion: velero.io/v1
kind: Schedule
metadata:
  name: <schedule>
  namespace: openshift-adp
spec:
  schedule: 0 7 * * *
  template:
    hooks: {}
    includedNamespaces:
    - <namespace>
    storageLocation: velero-demo-1
    defaultVolumesToFsBackup: true
    ttl: 720h0m0s
```

L'expression Cron 0 7 * * * * signifie qu'une sauvegarde sera créée à 7:00 chaque jour. Les espaces de noms à inclure dans la sauvegarde et l'emplacement de stockage de la sauvegarde sont également spécifiés. Par conséquent, au lieu d'une CR de sauvegarde, la CR de planification est utilisée pour créer une sauvegarde à l'heure et à la fréquence spécifiées.

Une fois le planning créé, il est activé.



Les sauvegardes seront créées en fonction de ce planning et peuvent être affichées à partir de l'onglet sauvegarde.



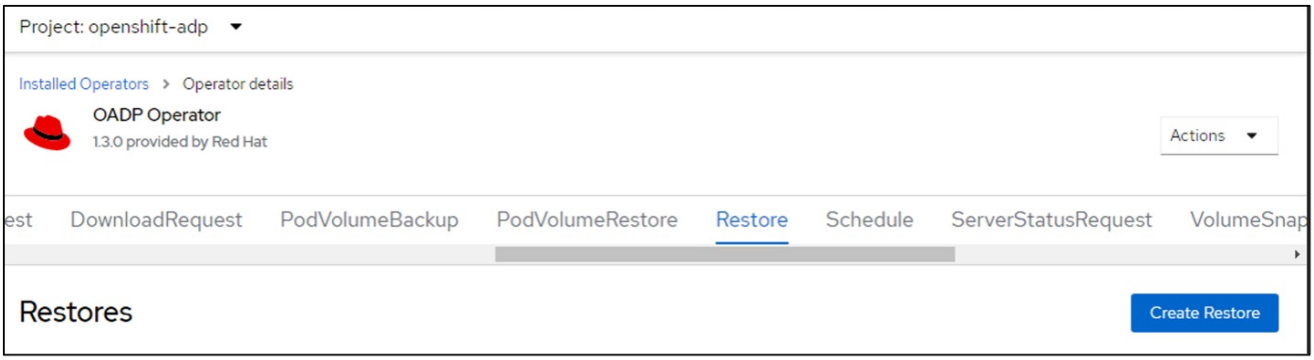
Restaurer une machine virtuelle à partir d'une sauvegarde

Prérequis

Pour effectuer une restauration à partir d'une sauvegarde, supposons que l'espace de noms dans lequel existait la machine virtuelle a été accidentellement supprimé.

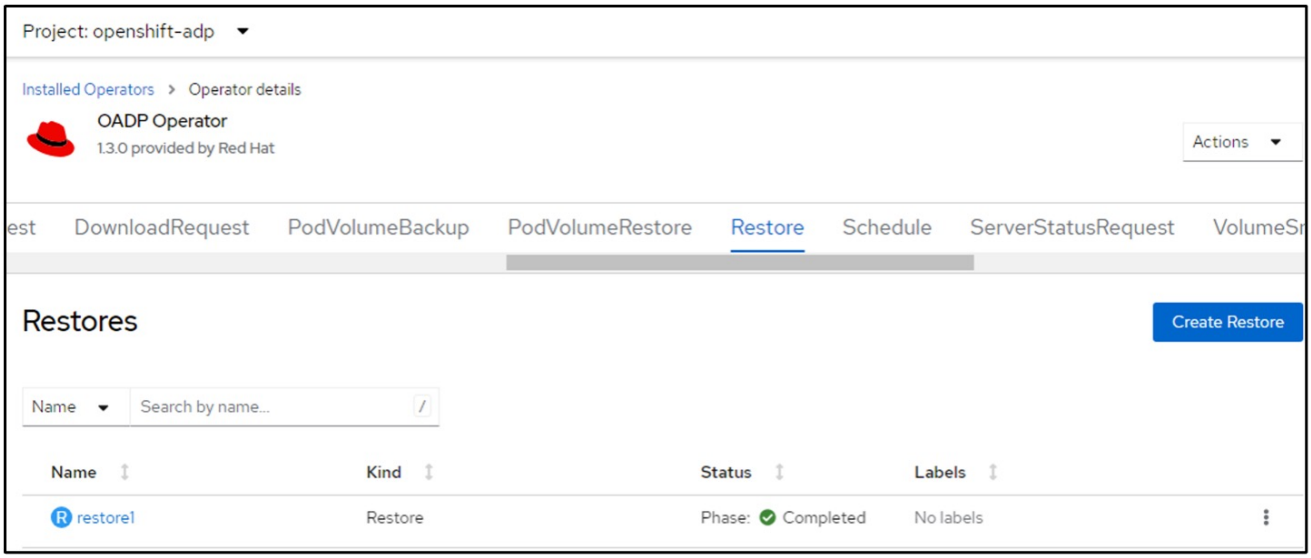
Restaurer dans le même espace de noms

Pour restaurer à partir de la sauvegarde que nous venons de créer, nous devons créer une ressource personnalisée de restauration (CR). Nous devons lui fournir un nom, fournir le nom de la sauvegarde à partir de laquelle nous voulons restaurer et définir les PV de restauration sur true. Des paramètres supplémentaires peuvent être définis comme indiqué dans le "documentation". Cliquez sur le bouton Créer.




```
apiVersion: velero.io/v1
kind: Restore
metadata:
  name: restore1
  namespace: openshift-adp
spec:
  backupName: backup1
  restorePVs: true
```

Lorsque la phase affiche terminé, vous pouvez voir que les machines virtuelles ont été restaurées à l'état au moment où l'instantané a été pris. (Si la sauvegarde a été créée lors de l'exécution de la machine virtuelle, la restauration de la machine virtuelle à partir de la sauvegarde démarre la machine virtuelle restaurée et la met en état d'exécution). La machine virtuelle est restaurée dans le même espace de noms.



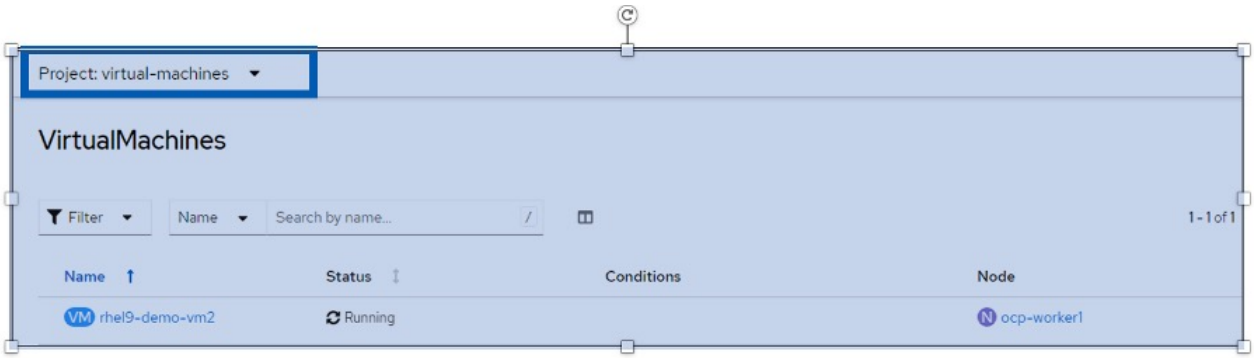
Restauration dans un autre espace de noms

Pour restaurer la machine virtuelle dans un espace de noms différent, vous pouvez fournir un espace de noms dans la définition yaml de la CR de restauration.

L'exemple de fichier yaml suivant crée une CR de restauration pour restaurer une machine virtuelle et ses disques dans l'espace de nom de démonstration des machines virtuelles lorsque la sauvegarde a été effectuée dans l'espace de noms des machines virtuelles.

```
apiVersion: velero.io/v1
kind: Restore
metadata:
  name: restore-to-different-ns
  namespace: openshift-adp
spec:
  backupName: backup
  restorePVs: true
  includedNamespaces:
  - virtual-machines-demo
  namespaceMapping:
    virtual-machines-demo: virtual-machines
```

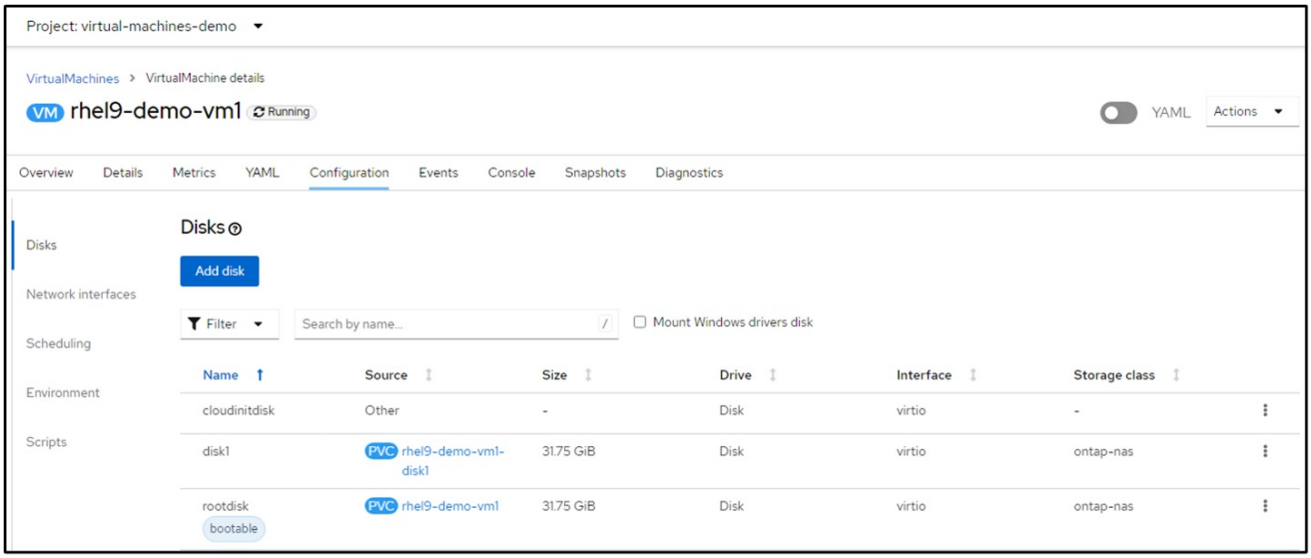
Lorsque la phase affiche terminé, vous pouvez voir que les machines virtuelles ont été restaurées à l'état au moment où l'instantané a été pris. (Si la sauvegarde a été créée lors de l'exécution de la machine virtuelle, la restauration de la machine virtuelle à partir de la sauvegarde démarre la machine virtuelle restaurée et la met en état d'exécution). La machine virtuelle est restaurée dans un espace de noms différent, comme spécifié dans le yaml.

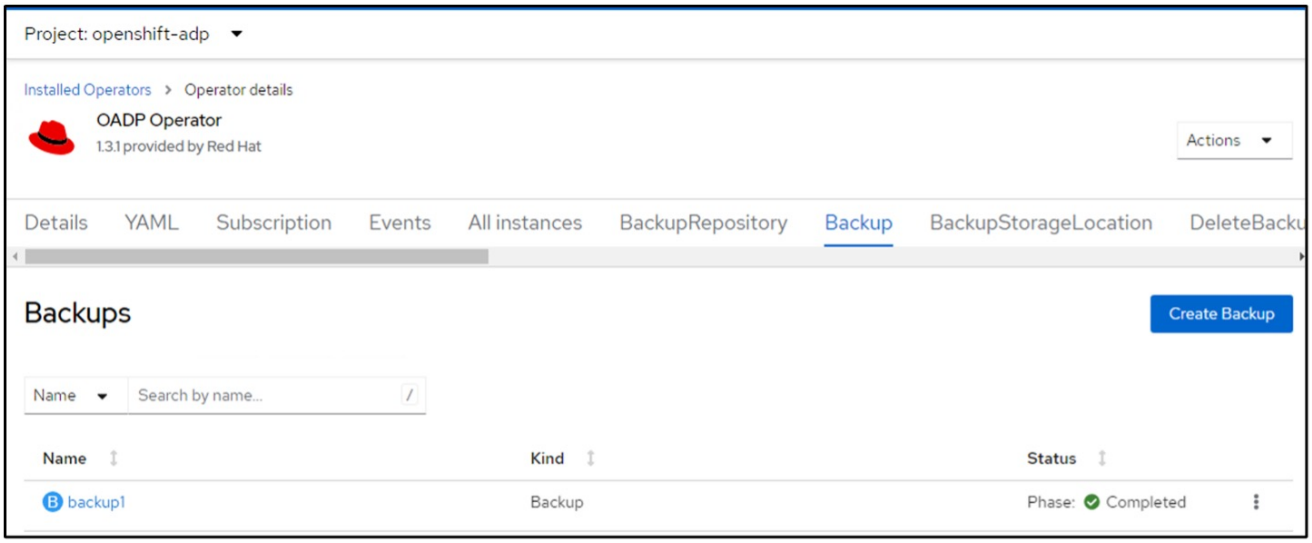


Restauration vers une autre classe de stockage

Velero fournit une capacité générique de modifier les ressources pendant la restauration en spécifiant des correctifs json. Les correctifs json sont appliqués aux ressources avant leur restauration. Les patches json sont spécifiés dans un configmap et le configmap est référencé dans la commande restore. Cette fonctionnalité vous permet de restaurer à l'aide d'une classe de stockage différente.

Dans l'exemple ci-dessous, la machine virtuelle, lors de la création, utilise ontap-nas comme classe de stockage pour ses disques. Une sauvegarde de la machine virtuelle nommée backup1 est créée.





Simulez une perte de la machine virtuelle en supprimant cette dernière.

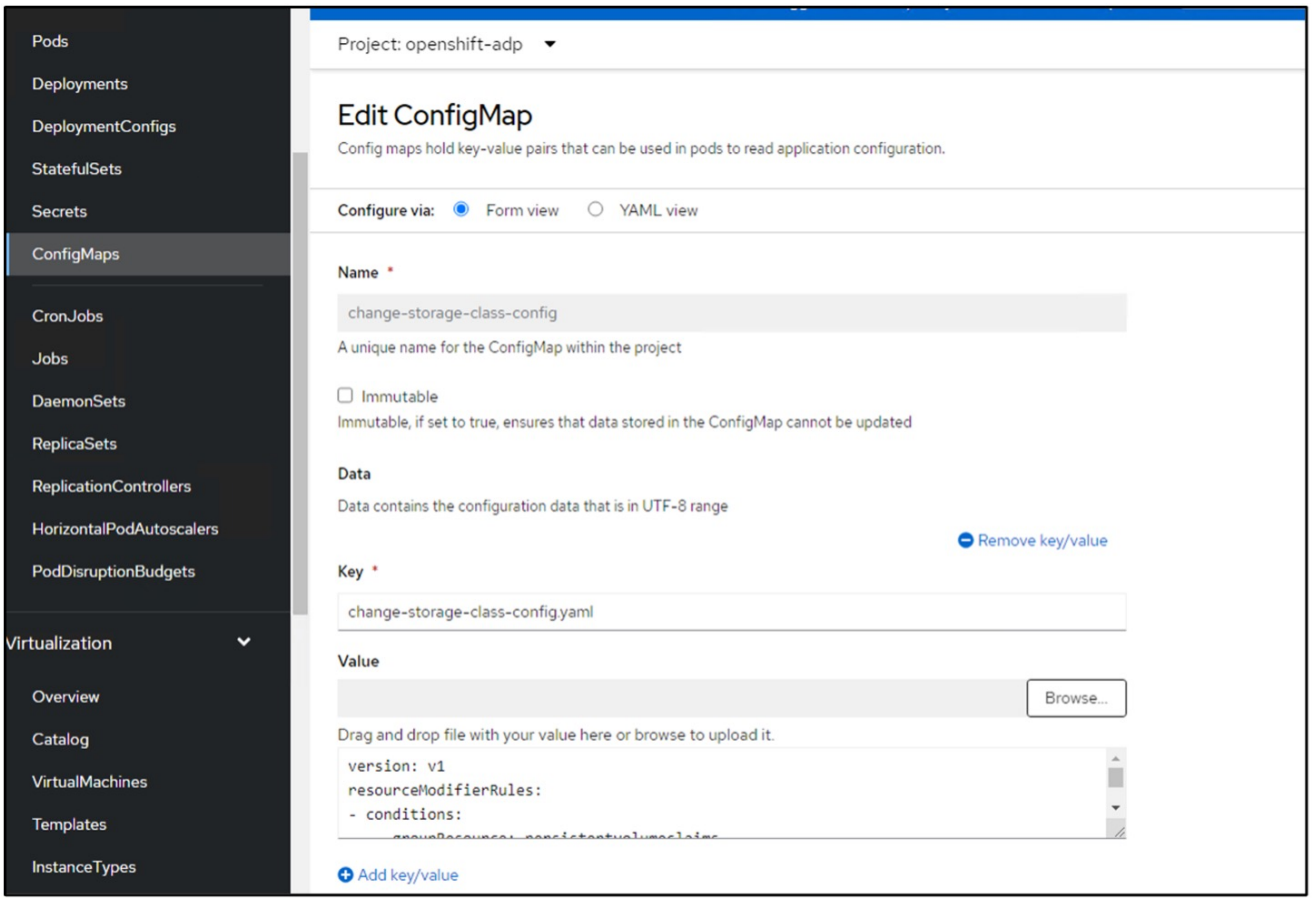
Pour restaurer la machine virtuelle à l'aide d'une classe de stockage différente, par exemple, la classe de stockage ontap-nas-ECO, vous devez effectuer les deux étapes suivantes :

Étape 1

Créez un mappage de configuration (console) dans l'espace de noms openshift-adp comme suit :
Renseignez les détails comme indiqué dans la capture d'écran :
Sélectionnez namespace : openshift-adp
Name : change-Storage-class-config (peut être n'importe quel nom)

Clé : change-Storage-class-config.yaml :
Valeur :

```
version: v1
    resourceModifierRules:
    - conditions:
        groupResource: persistentvolumeclaims
        resourceNameRegex: "^rhel*"
        namespaces:
        - virtual-machines-demo
      patches:
      - operation: replace
        path: "/spec/storageClassName"
        value: "ontap-nas-eco"
```



L'objet de mappage de configuration résultant doit ressembler à ceci (CLI) :

```
# kubectl describe cm/change-storage-class-config -n openshift-
adp
Name:         change-storage-class-config
Namespace:    openshift-adp
Labels:       velero.io/change-storage-class=RestoreItemAction
              velero.io/plugin-config=
Annotations:  <none>

Data
====
change-storage-class-config.yaml:
----
version: v1
resourceModifierRules:
- conditions:
     groupResource: persistentvolumeclaims
     resourceNameRegex: "^rhel*"
     namespaces:
     - virtual-machines-demo
  patches:
  - operation: replace
    path: "/spec/storageClassName"
    value: "ontap-nas-eco"



BinaryData
====

Events:  <none>
```

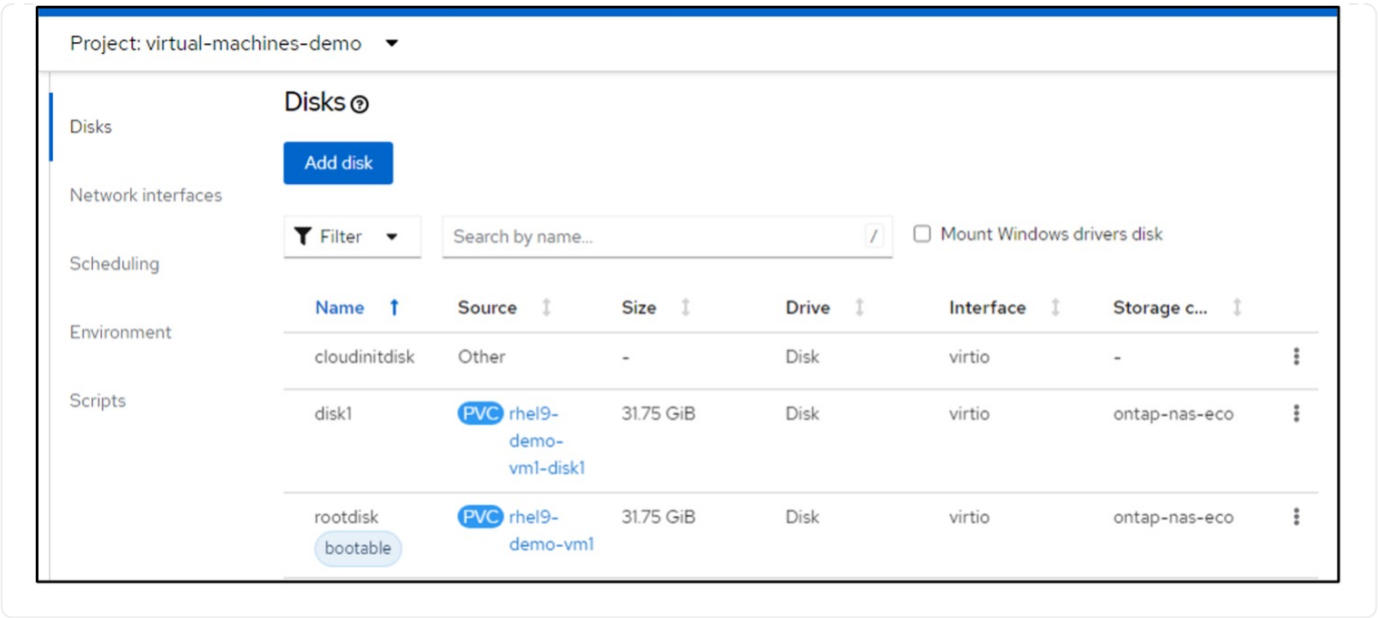
Cette carte de configuration applique la règle de modificateur de ressource lors de la création de la restauration. Un correctif sera appliqué pour remplacer le nom de classe de stockage par ontap-nas-eco pour toutes les demandes de volume persistant commençant par rhel.

Étape 2

Pour restaurer la machine virtuelle, utilisez la commande suivante depuis l'interface de ligne de commande Velero :

```
#velero restore create restore1 --from-backup backup1 --resource
-modifier-configmap change-storage-class-config -n openshift-adp
```

La machine virtuelle est restaurée dans le même namespace avec les disques créés à l'aide de la classe de stockage ontap-nas-eco.



Suppression des sauvegardes et des restaurations dans à l'aide de Velero

Suppression d'une sauvegarde

Vous pouvez supprimer une CR de sauvegarde sans supprimer les données de stockage d'objet à l'aide de l'outil CLI OC.

```
oc delete backup <backup_CR_name> -n <velero_namespace>
```

Si vous souhaitez supprimer la CR de sauvegarde et supprimer les données de stockage d'objets associées, vous pouvez le faire à l'aide de l'outil CLI de Velero.

Téléchargez l'interface de ligne de commande comme indiqué dans les instructions du "Documentation Velero".

Exécutez la commande DELETE suivante à l'aide de l'interface de ligne de commande Velero

```
velero backup delete <backup_CR_name> -n <velero_namespace>
```

Vous pouvez également supprimer la CR de restauration à l'aide de l'interface de ligne de commande Velero

```
velero restore delete restore --namespace openshift-adp
```

Vous pouvez utiliser la commande oc ainsi que l'interface utilisateur pour supprimer la CR de restauration

```
oc delete backup <backup_CR_name> -n <velero_namespace>
```

Surveillance à l'aide de Cloud Insights

Surveillance à l'aide de Cloud Insights pour les VM dans Red Hat OpenShift Virtualization

Auteur: Banu Sundhar, NetApp

Cette section du document de référence décrit en détail l'intégration de NetApp Cloud Insights à un cluster Red Hat OpenShift pour surveiller les machines virtuelles OpenShift Virtualization.

NetApp Cloud Insights est un outil de surveillance de l'infrastructure cloud qui permet de bénéficier d'une grande visibilité sur l'ensemble de l'infrastructure. Avec Cloud Insights, vous pouvez surveiller toutes les ressources, les optimiser et résoudre les problèmes, y compris dans les clouds publics et dans vos data centers privés. Pour plus d'informations sur NetApp Cloud Insights, reportez-vous au "Documentation Cloud Insights".

Pour commencer à utiliser Cloud Insights, vous devez vous inscrire sur le portail NetApp BlueXP. Pour plus de détails, reportez-vous à la "Intégration de Cloud Insights"

Cloud Insights offre plusieurs fonctionnalités qui vous permettent de trouver des données, de résoudre des problèmes et d'obtenir des informations exploitables sur votre environnement, rapidement et facilement. Vous pouvez facilement trouver les données à l'aide de requêtes puissantes, visualiser les données dans des tableaux de bord et envoyer des alertes par e-mail pour les seuils de données que vous avez définis. Reportez-vous à la "didacticiels vidéo" pour vous aider à comprendre ces fonctionnalités.

Pour que Cloud Insights commence à collecter des données, vous avez besoin des éléments suivants

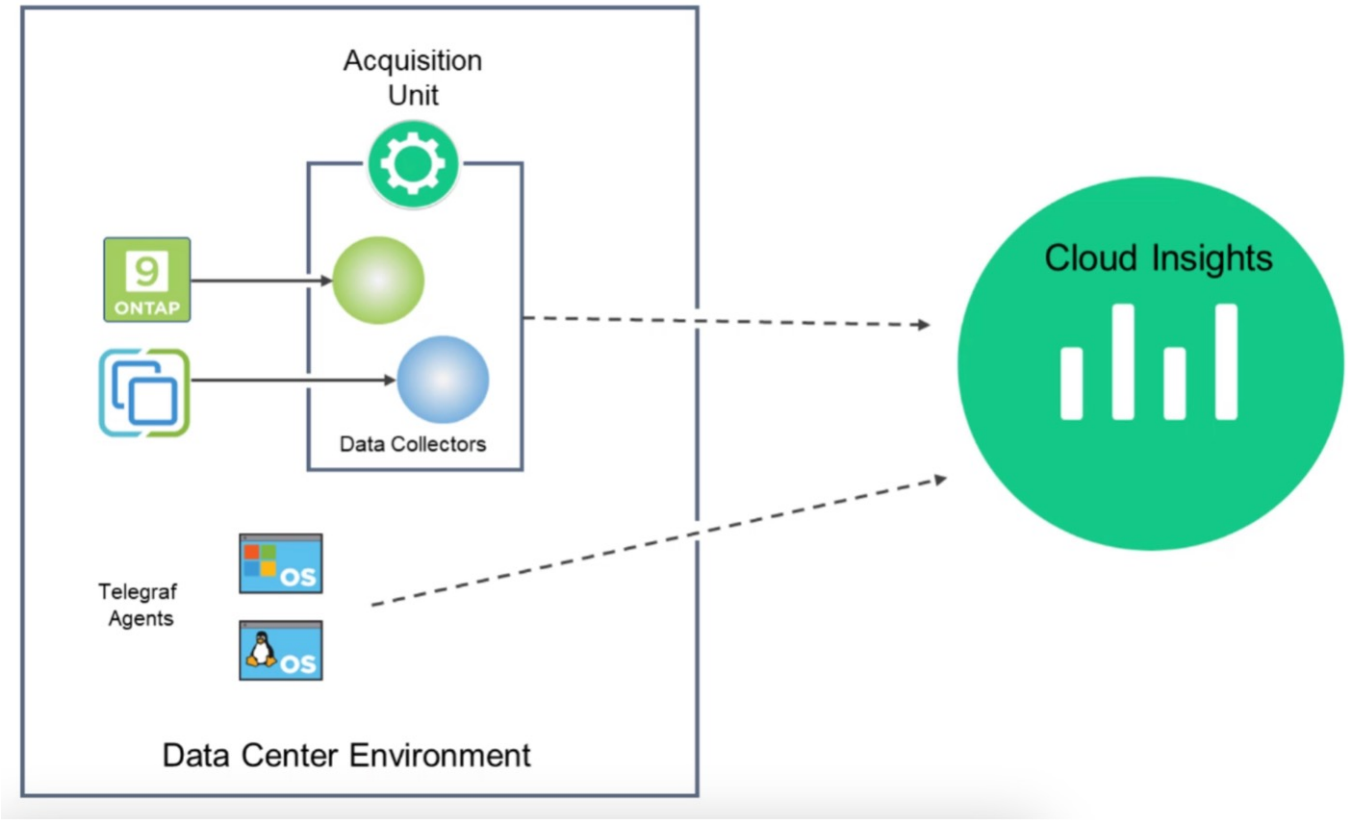
Collecteurs de données
Il existe 3 types de collecteurs de données :
* Infrastructure (périphériques de stockage, commutateurs réseau, infrastructure informatique)
* Systèmes d'exploitation (tels que VMware ou Windows)
* Services (tels que Kafka)

Les collecteurs de données détectent les informations des sources de données, telles que les périphériques de stockage ONTAP (collecteur de données d'infrastructure). Les informations collectées sont utilisées pour l'analyse, la validation, la surveillance et le dépannage.

Unité d'acquisition
Si vous utilisez un Data Collector d'infrastructure, vous avez également besoin d'une unité d'acquisition pour injecter des données dans Cloud Insights. Une unité d'acquisition est un ordinateur dédié à l'hébergement de collecteurs de données, généralement une machine virtuelle. Cet ordinateur se trouve généralement dans le même data Center/VPC que les éléments surveillés.

Agents Telegraf
Cloud Insights prend également en charge Telegraf comme agent de collecte des données d'intégration. Telegraf est un agent serveur piloté par plug-in qui peut être utilisé pour collecter et signaler des mesures, des événements et des journaux.

Architecture Cloud Insights



Intégration à Cloud Insights pour les VM dans Red Hat OpenShift Virtualization

Pour commencer à collecter des données pour les machines virtuelles dans OpenShift Virtualization, vous devez installer :

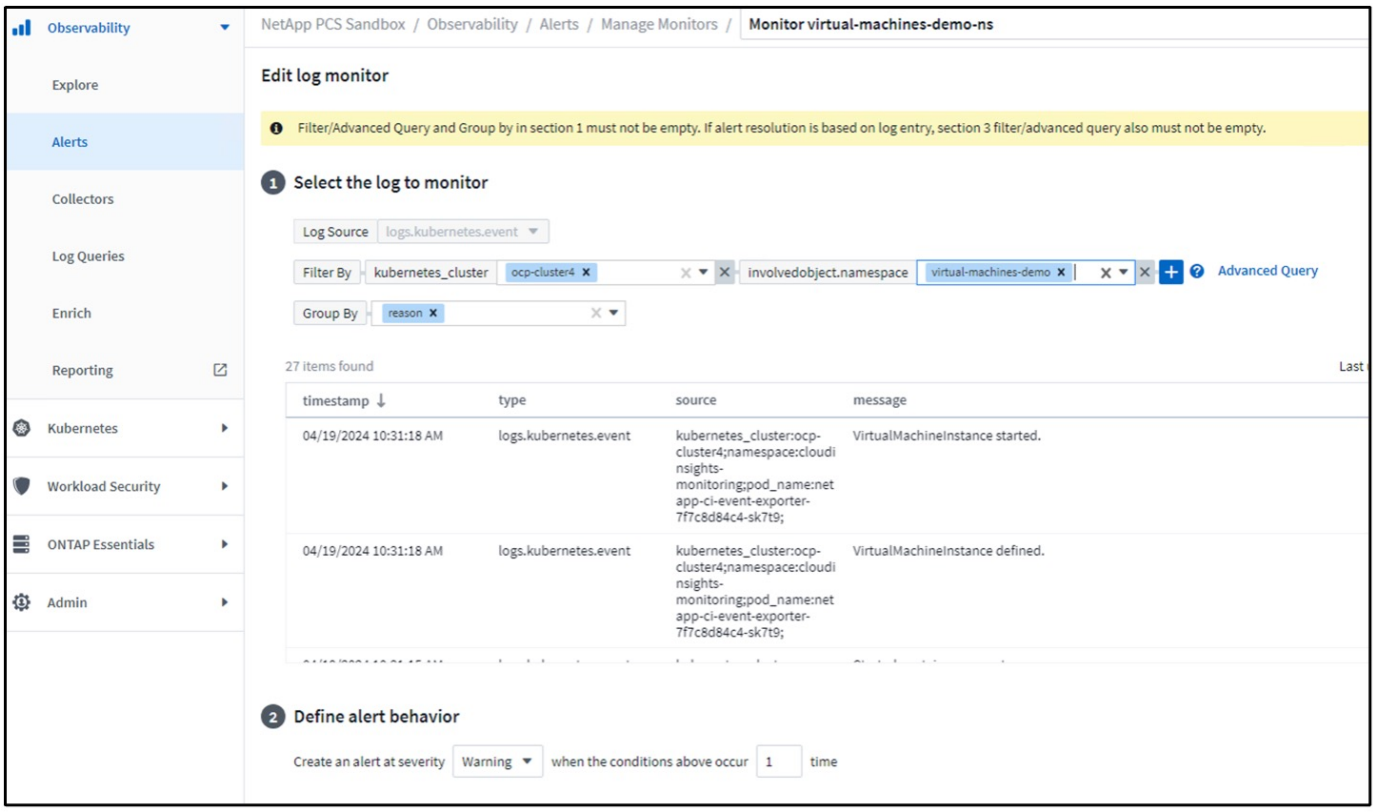
1. Opérateur de surveillance Kubernetes et collecteur de données pour collecter les données Kubernetes
   Pour obtenir des instructions complètes, reportez-vous au "documentation".
2. Unité d'acquisition permettant de collecter les données du stockage ONTAP qui fournit un stockage persistant pour les disques de la machine virtuelle
   Pour obtenir des instructions complètes, reportez-vous au "documentation".
3. Collecteur de données pour ONTAP
   Pour obtenir des instructions complètes, reportez-vous au "documentation"

En outre, si vous utilisez StorageGRID pour les sauvegardes de machines virtuelles, vous avez également besoin d'un collecteur de données pour le StorageGRID.

Exemples de fonctionnalités de surveillance pour les machines virtuelles dans Red Hat OpenShift Virtualization

Surveillance basée sur les événements et création d'alertes

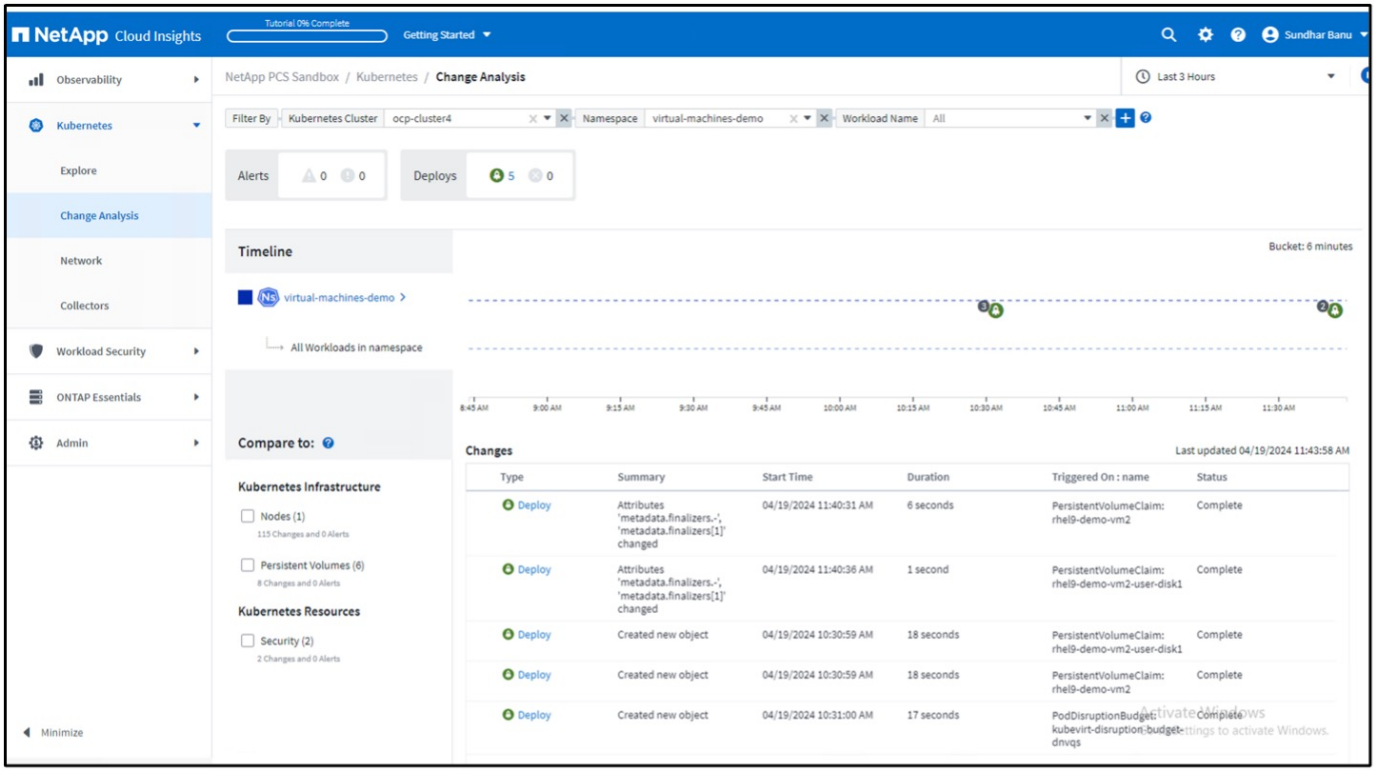
Voici un exemple dans lequel l'espace de noms qui contient une machine virtuelle dans OpenShift Virtualization est contrôlé en fonction des événements. Dans cet exemple, un moniteur est créé sur la base de **logs.kubernetes**.event pour l'espace de noms spécifié dans le cluster.



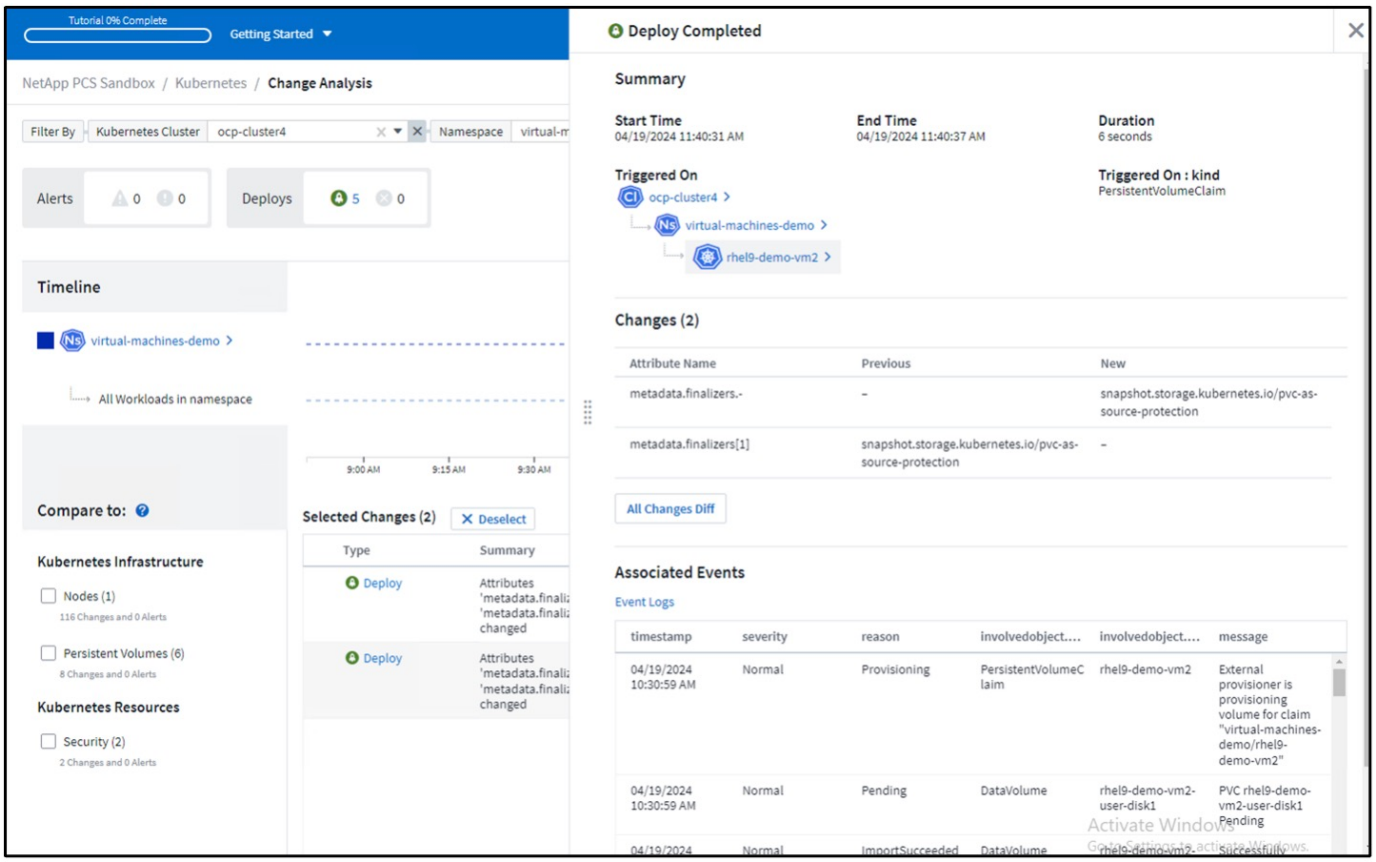
Cette requête fournit tous les événements de la machine virtuelle dans l'espace de nom. (Il n'y a qu'une seule machine virtuelle dans l'espace de noms). Une requête avancée peut également être construite pour filtrer sur la base de l'événement où la raison est « échec » ou « montage en panne ». Ces événements sont généralement créés en cas de problème lors de la création d'un volume persistant ou du montage du volume persistant sur un pod dynamique pour indiquer des problèmes dans le mécanisme de provisionnement dynamique afin de créer un volume persistant volumes pour la machine virtuelle.
Lors de la création du moniteur d'alertes comme indiqué ci-dessus, vous pouvez également configurer la notification aux destinataires. Vous pouvez également fournir des actions correctives ou des informations supplémentaires qui peuvent être utiles pour résoudre l'erreur. Dans l'exemple ci-dessus, des informations supplémentaires peuvent être utiles pour examiner la configuration back-end Trident et les définitions des classes de stockage afin de résoudre le problème.

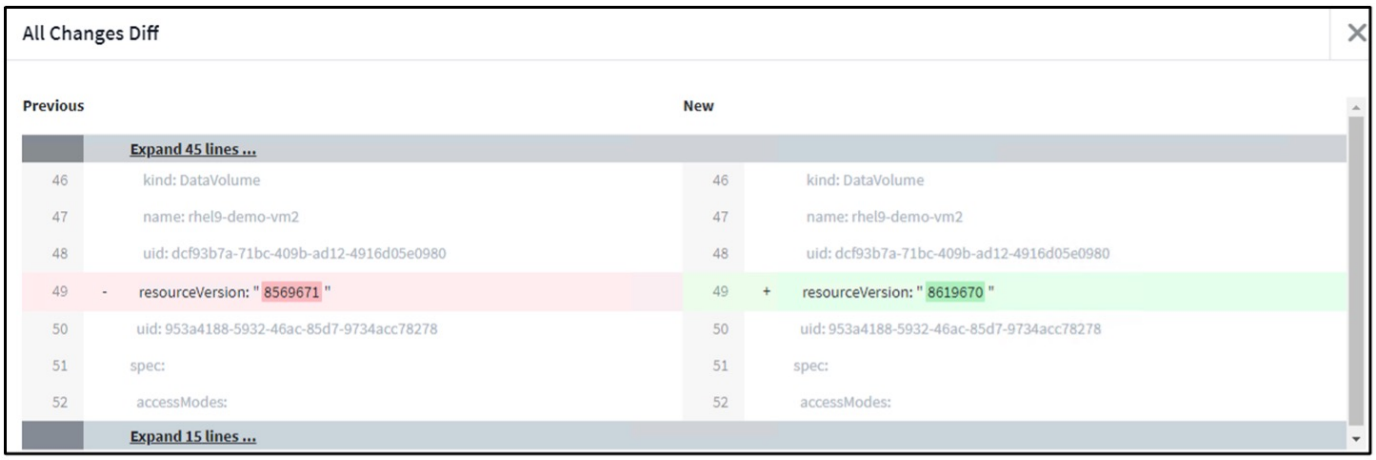
Analyse des changements

Avec change Analytics, vous pouvez afficher les modifications apportées à l'état de votre cluster, y compris les personnes qui ont apporté ces modifications qui peuvent vous aider à résoudre les problèmes.



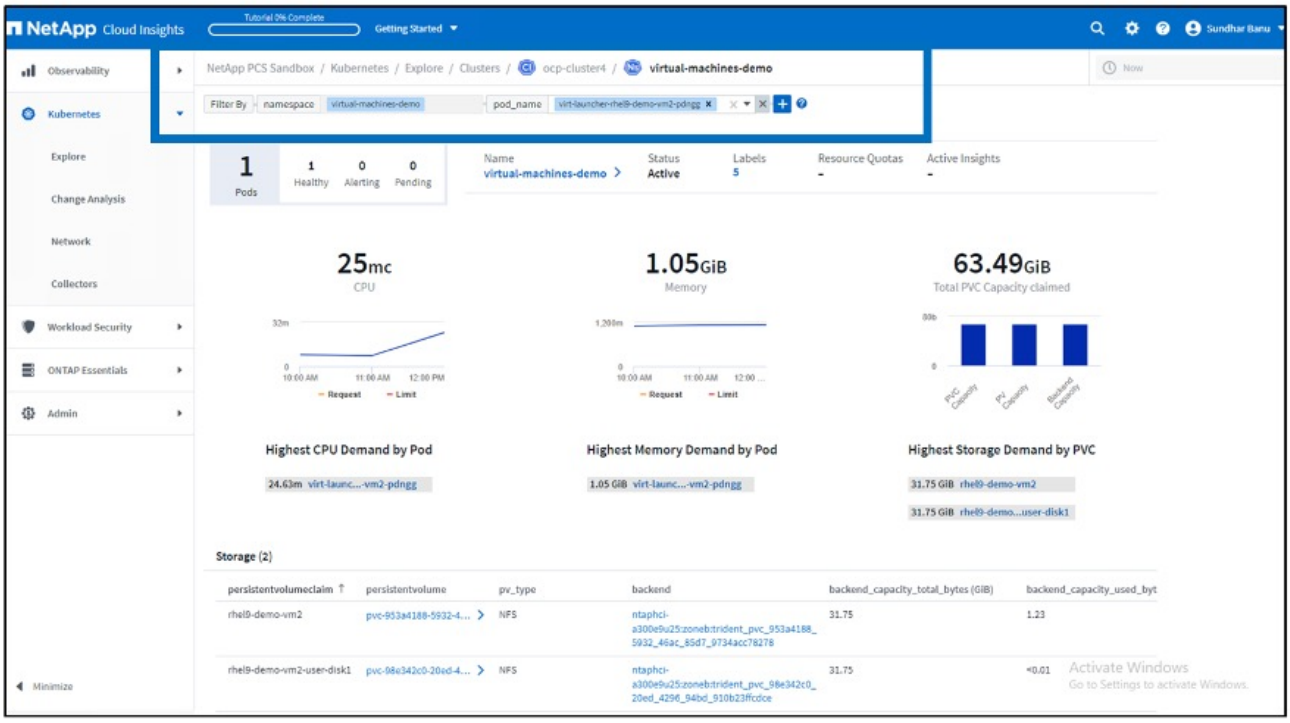
Dans l'exemple ci-dessus, change Analysis est configuré sur le cluster OpenShift pour l'espace de noms contenant une VM OpenShift Virtualization. Le tableau de bord affiche les modifications par rapport à la chronologie. Vous pouvez explorer pour voir ce qui a changé et cliquer sur All Changes Diff pour voir la différence des manifestes. Dans le manifeste, vous pouvez voir qu'une nouvelle sauvegarde des disques persistants a été créée.



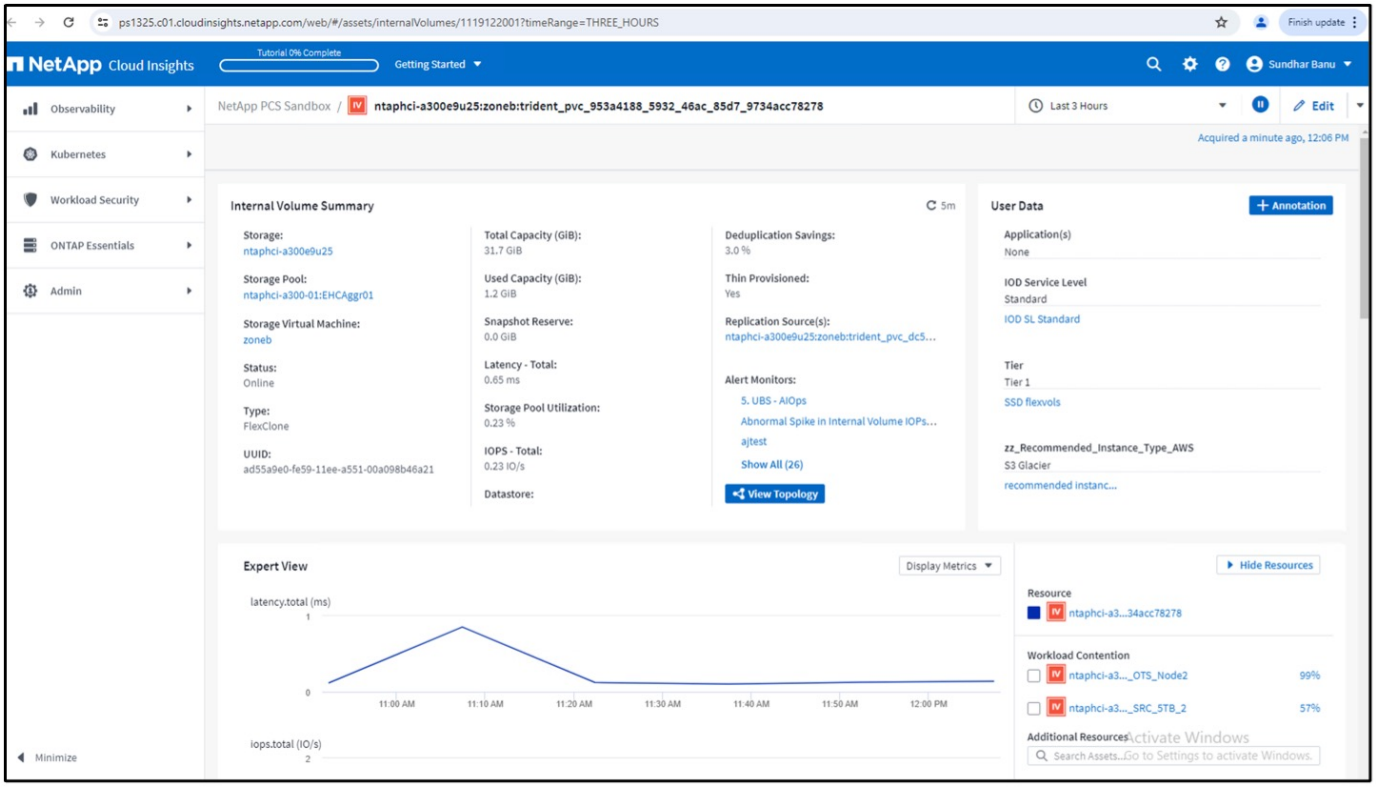


Mappage de stockage back-end

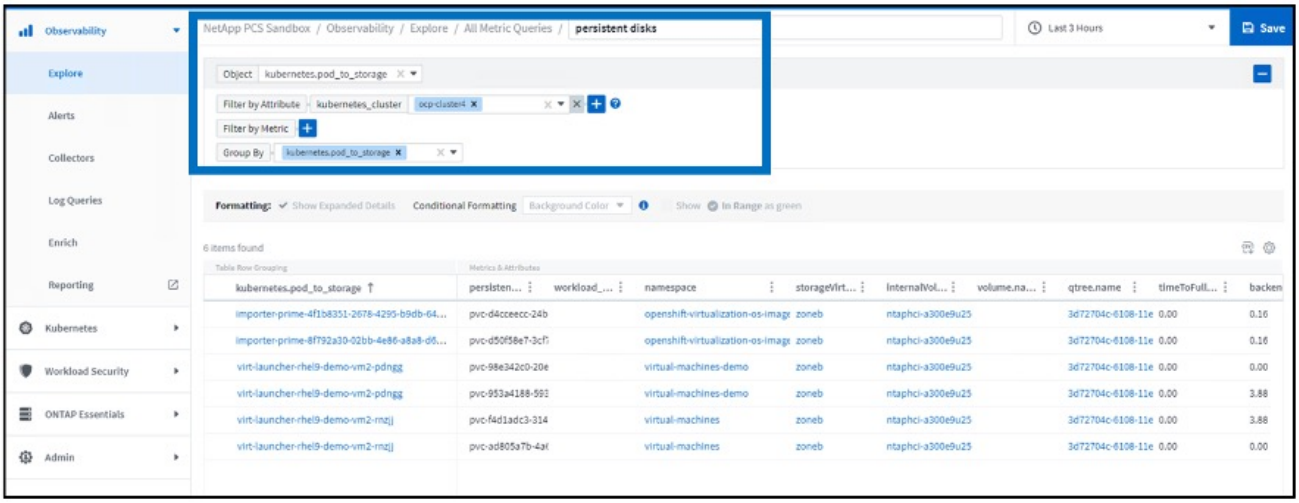
Avec Cloud Insights, vous pouvez facilement consulter le stockage back-end des disques de VM et plusieurs statistiques sur les demandes de volume virtuel.



Vous pouvez cliquer sur les liens sous la colonne backend, qui extront les données directement depuis le système de stockage ONTAP back-end.

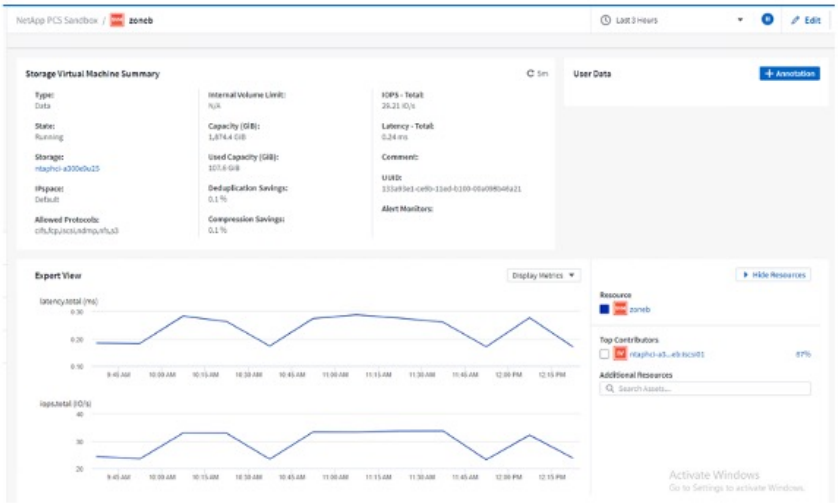


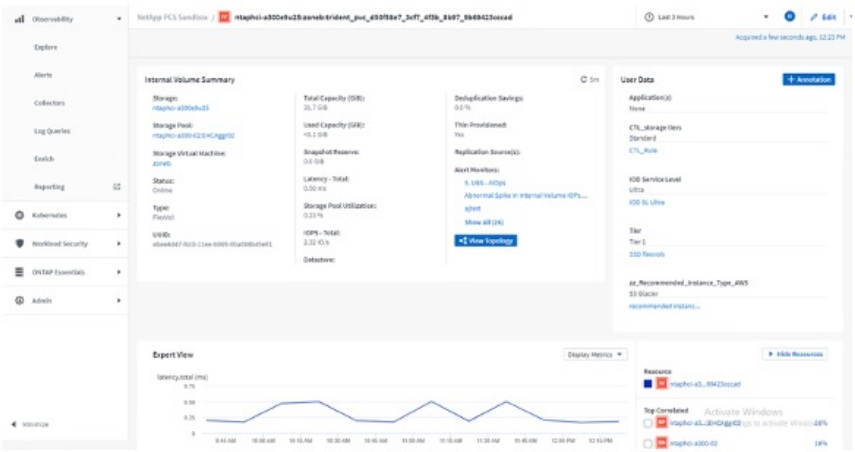
Une autre façon d'examiner le mappage du pod au stockage est de créer une requête All Metrics à partir du menu d'observabilité sous Explore.



Cliquez sur l'un des liens pour obtenir les détails correspondants sur le stockage ONTAP. Par exemple, en cliquant sur le nom d'un SVM dans la colonne storageVirtualmachine, on extrait les détails sur le SVM de ONTAP. Si vous cliquez sur le nom d'un volume interne, les détails relatifs au volume dans ONTAP seront détaillés.

	storageVirtualMachin...	internalVolume.name	volume.na..
zation-os-image	zoneb	ntaphci-a300e9u25:zoneb:trident_p	
zation-os-image	zoneb	ntaphci-a300e9u25:zoneb:trident_p	
demo	zoneb	ntaphci-a300e9u25:zoneb:trident_p	
demo	zoneb	ntaphci-a300e9u25:zoneb:trident_p	
	zoneb	ntaphci-a300e9u25:zoneb:trident_p	
	zoneb	ntaphci-a300e9u25:zoneb:trident_p	





Solution NetApp de gestion avancée des clusters pour Kubernetes sur Red Hat OpenShift

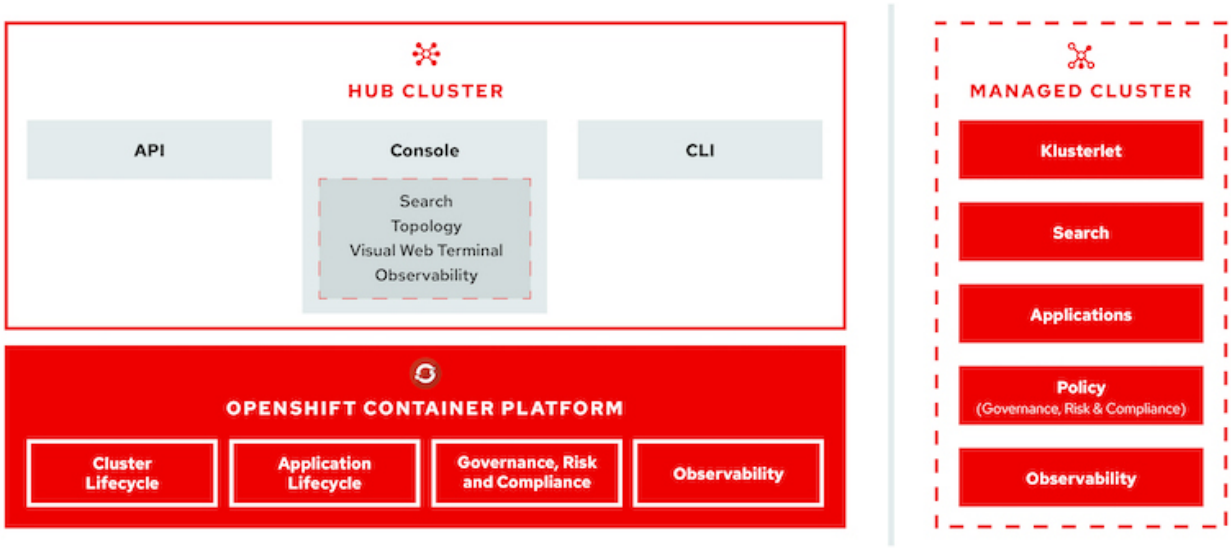
Gestion avancée des clusters pour Kubernetes : Red Hat OpenShift avec NetApp

Lorsqu'une application conteneurisée passe du développement à la production, de nombreuses entreprises ont besoin de plusieurs clusters Red Hat OpenShift pour prendre en charge les tests et le déploiement de cette application. Parallèlement, les entreprises hébergent généralement plusieurs applications ou charges de travail sur les clusters OpenShift. Par conséquent, chaque entreprise finit par gérer un ensemble de clusters. Les administrateurs OpenShift doivent donc faire face au défi que représente la gestion et la maintenance de plusieurs clusters sur un large éventail d'environnements répartis sur plusieurs data centers sur site et clouds publics. Pour relever ces défis, Red Hat a introduit la solution avancée de gestion de clusters pour Kubernetes.

Red Hat Advanced Cluster Management pour Kubernetes vous permet d'effectuer les tâches suivantes :

1. Créez, importez et gérez plusieurs clusters entre les data centers et les clouds publics
2. Déployer et gérer des applications ou des charges de travail sur plusieurs clusters à partir d'une console unique
3. Contrôler et analyser l'état et l'état des différentes ressources du cluster
4. Surveillez et appliquez la conformité aux règles de sécurité dans plusieurs clusters

Red Hat Advanced Cluster Management pour Kubernetes est installé en tant qu'extension d'un cluster Red Hat OpenShift, et ce cluster est utilisé comme contrôleur central pour toutes ses opérations. Ce cluster est connu sous le nom de cluster de concentrateur et expose un plan de gestion permettant aux utilisateurs de se connecter à Advanced Cluster Management. Tous les autres clusters OpenShift importés ou créés via la console Advanced Cluster Management sont gérés par le cluster Hub et appelés clusters gérés. Il installe un agent appelé Klausterlet sur les clusters gérés afin de les connecter au cluster Hub et de répondre aux demandes différentes activités liées à la gestion du cycle de vie des clusters, à la gestion du cycle de vie des applications, à l'observabilité et à la conformité de la sécurité.



Pour plus d'informations, reportez-vous à la documentation "ici".

Déploiement

Déploiement de la gestion avancée des clusters pour Kubernetes

Prérequis

1. Un cluster Red Hat OpenShift (supérieur à la version 4.5) pour le cluster Hub
2. Clusters Red Hat OpenShift (supérieurs à la version 4.4.3) pour les clusters gérés
3. L'accès cluster-admin au cluster Red Hat OpenShift
4. Un abonnement Red Hat à Advanced Cluster Management pour Kubernetes

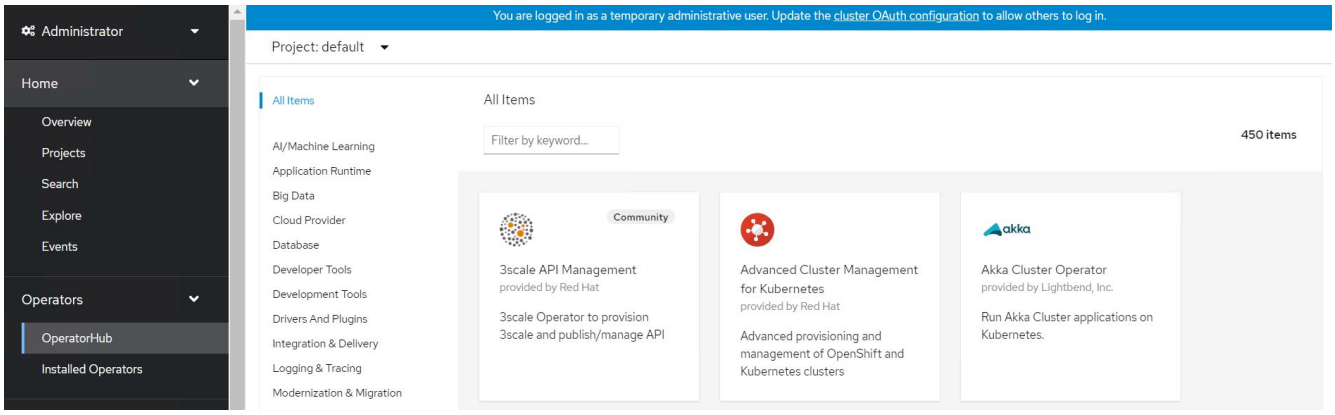
Advanced Cluster Management est un module complémentaire du cluster OpenShift. Il existe donc certaines conditions et restrictions sur les ressources matérielles en fonction des fonctionnalités utilisées sur le concentrateur et les clusters gérés. Vous devez tenir compte de ces problèmes lors du dimensionnement des clusters. Voir la documentation "ici" pour en savoir plus.

Si le cluster Hub dispose de nœuds dédiés pour héberger les composants de l'infrastructure et que vous souhaitez installer les ressources Advanced Cluster Management uniquement sur ces nœuds, vous devez ajouter des tolérances et des sélecteurs à ces nœuds en conséquence. Pour plus de détails, consultez la documentation "ici".

Déploiement de la gestion avancée des clusters pour Kubernetes

Pour installer Advanced Cluster Management pour Kubernetes sur un cluster OpenShift, effectuez les opérations suivantes :

1. Choisissez un cluster OpenShift en tant que cluster Hub et connectez-vous avec les privilèges cluster-admin.
2. Accédez à Operators > Operators Hub et recherchez Advanced Cluster Management pour Kubernetes.



3. Sélectionnez Advanced Cluster Management pour Kubernetes et cliquez sur Install.

4. Dans l'écran Install Operator, indiquez les détails nécessaires (NetApp recommande de conserver les paramètres par défaut) et cliquez sur Install.

OperatorHub > Operator Installation

Install Operator

Install your Operator by subscribing to one of the update channels to keep the Operator up to date. The strategy determines either manual or automatic updates.

Update channel *

- ○ release-2.0
- ○ release-2.1
- ◉ release-2.2

Installation mode *

- ○ All namespaces on the cluster (default)
  This mode is not supported by this Operator
- ◉ A specific namespace on the cluster
  Operator will be available in a single Namespace only.

Installed Namespace *

- ◉ Operator recommended Namespace: PR open-cluster-management

Namespace creation
Namespace **open-cluster-management** does not exist and will be created.

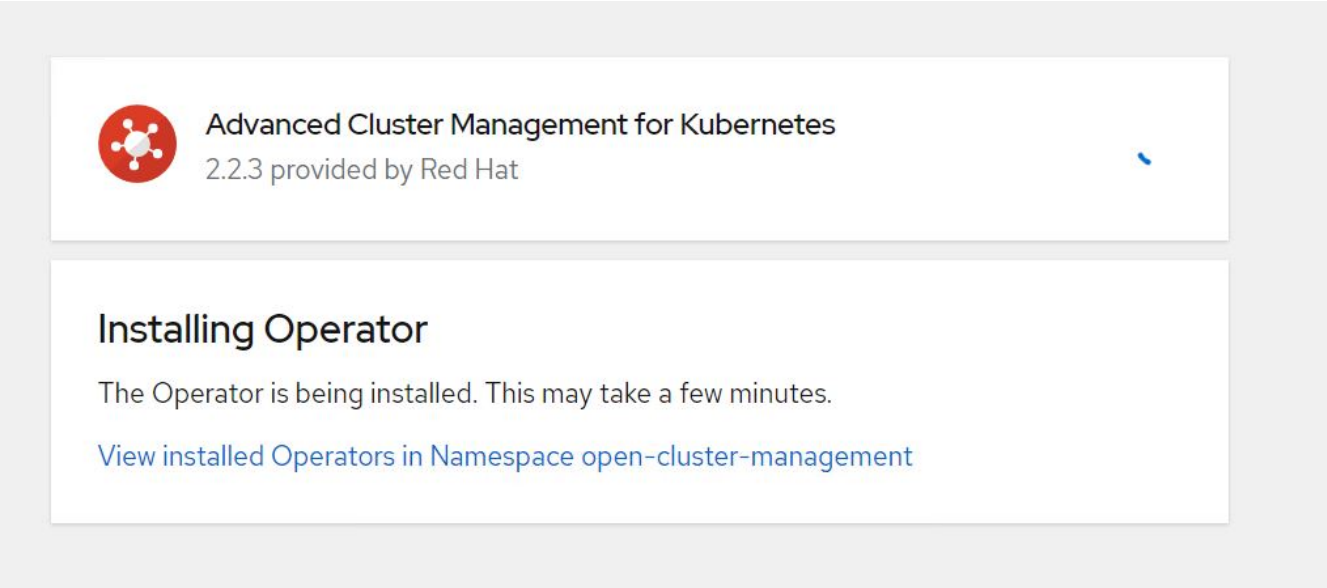
- ○ Select a Namespace

Approval strategy *

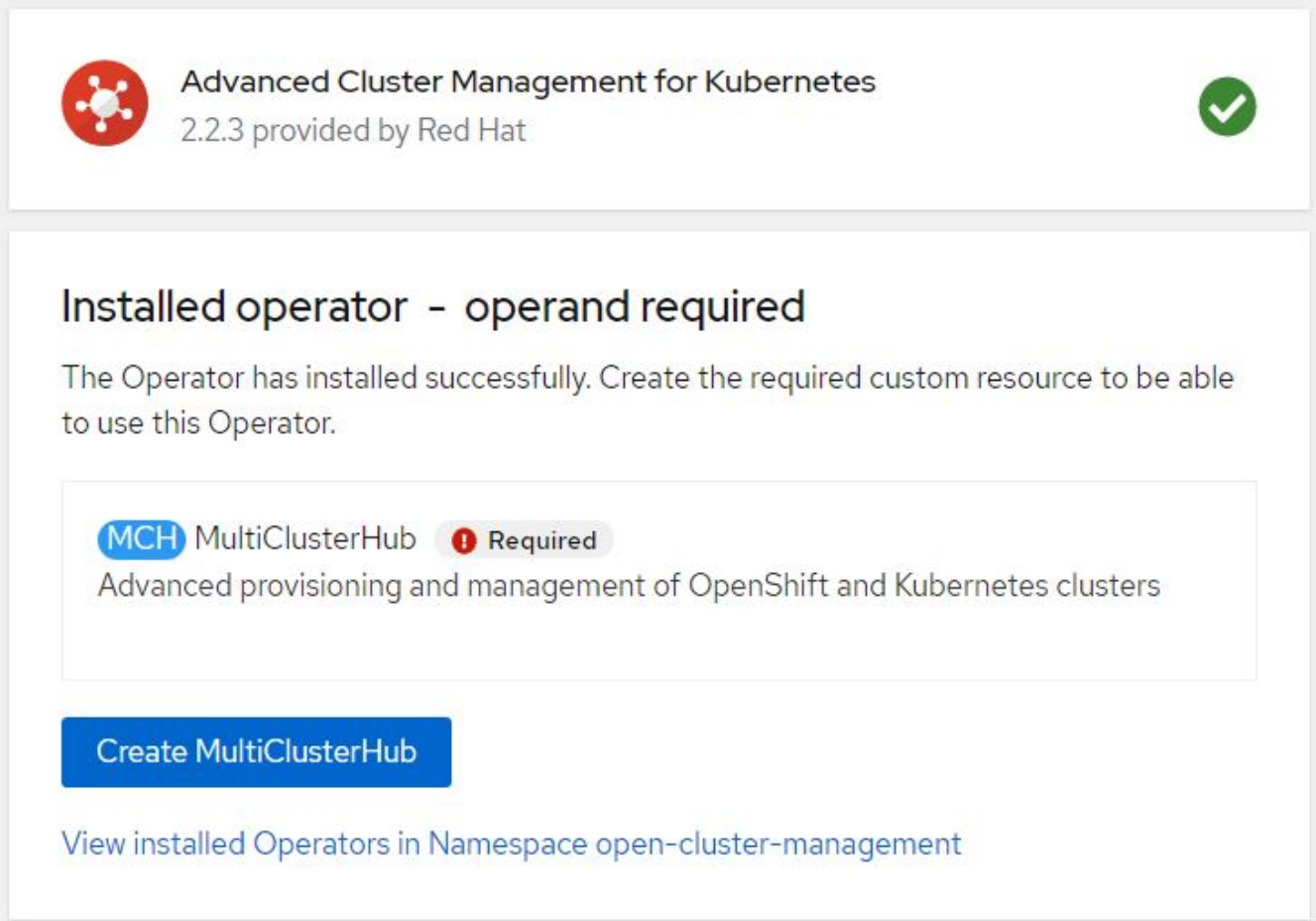
- ◉ Automatic
- ○ Manual

Install Cancel

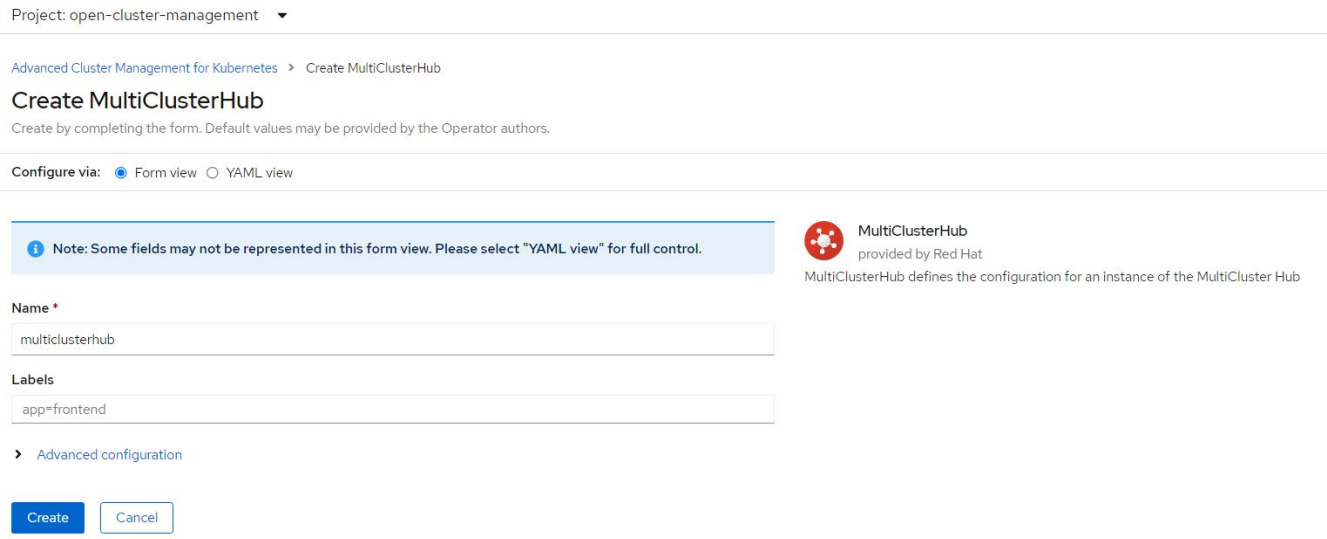
5. Attendre la fin de l'installation par l'opérateur.



6. Une fois l'opérateur installé, cliquez sur Créer MultiClusterHub.



7. Dans l'écran Créer MultiClusterHub, cliquez sur Créer après avoir donné les détails. Cela initie l'installation d'un hub multi-cluster.



8. Une fois que tous les pods passent à l'état d'exécution dans l'espace de noms d'open-cluster-management et que l'opérateur passe à l'état « réussi », Advanced Cluster Management pour Kubernetes est installé.

Project: open-cluster-management

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	Managed Namespaces	Status	Provided APIs
Advanced Cluster Management for Kubernetes 2.2.3 provided by Red Hat	NS open-cluster-management	Succeeded Up to date	MultiClusterHub ClusterManager ClusterDeployment ClusterState View 25 more...

9. L'installation du concentrateur prend un certain temps et, une fois cette opération effectuée, le concentrateur MultiCluster passe à l'état d'exécution.

Installed Operators > Operator details

Advanced Cluster Management for Kubernetes
2.2.3 provided by Red Hat

Actions

Details | YAML | Subscription | Events | All instances | MultiClusterHub | ClusterManager | ClusterDeployment | ClusterSt

MultiClusterHubs

Create MultiClusterHub

Name | Search by name...

Name	Kind	Status	Labels
MCH multiclusterhub	MultiClusterHub	Phase: Running	No labels

10. Elle crée une route dans l'espace de noms Open-cluster-management. Connectez-vous à l'URL de la route pour accéder à la console Advanced Cluster Management.

Project: open-cluster-management

Routes

Create Route

Filter | Name | mul

Name mul ✕ | Clear all filters

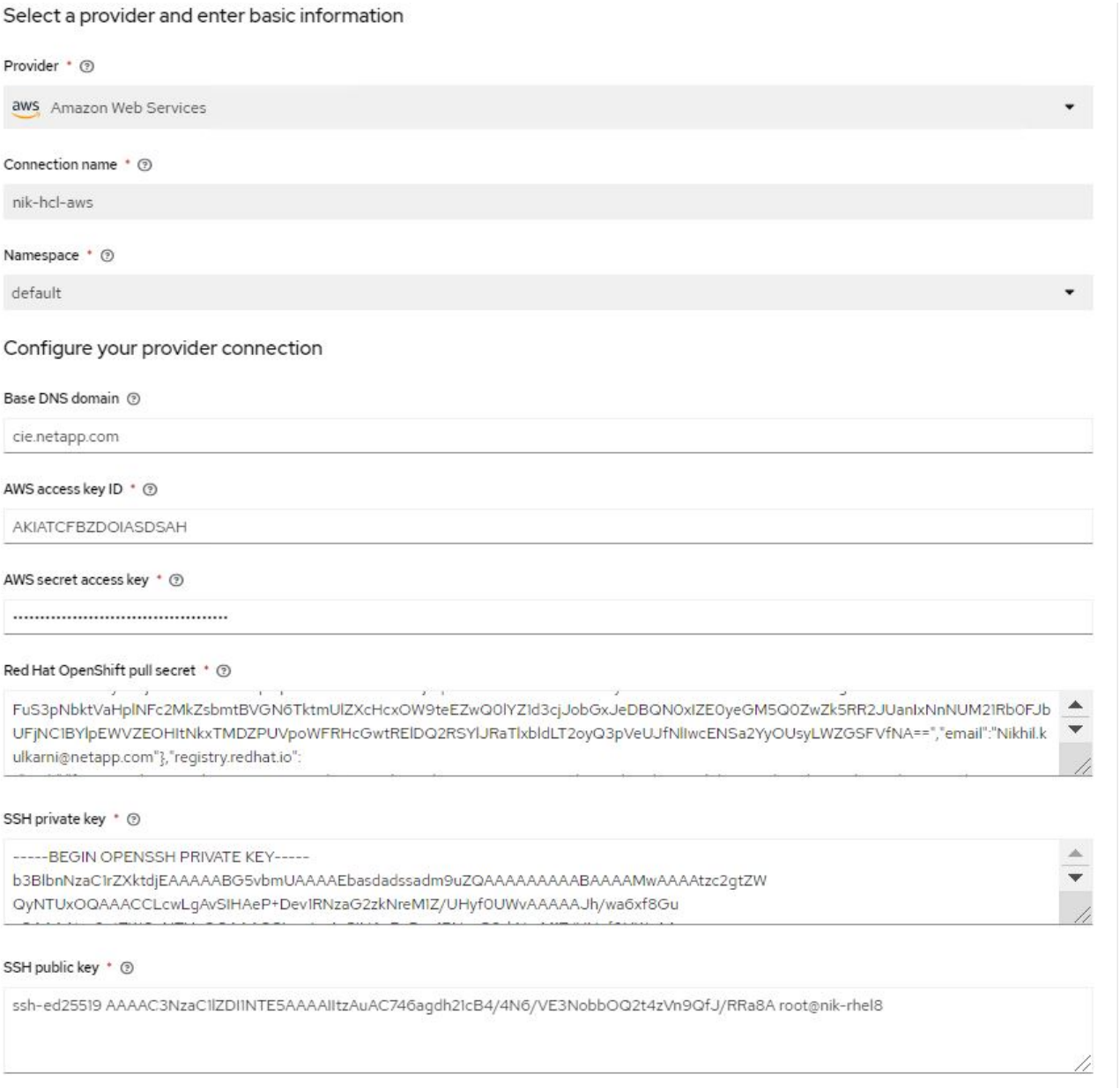
Name	Status	Location	Service
RT multicloud-console	Accepted	https://multicloud-console.apps.ocp-vmware2.cie.netapp.com	S management-ingress

Fonctionnalités : gestion avancée des clusters pour Kubernetes sur Red Hat OpenShift avec NetApp

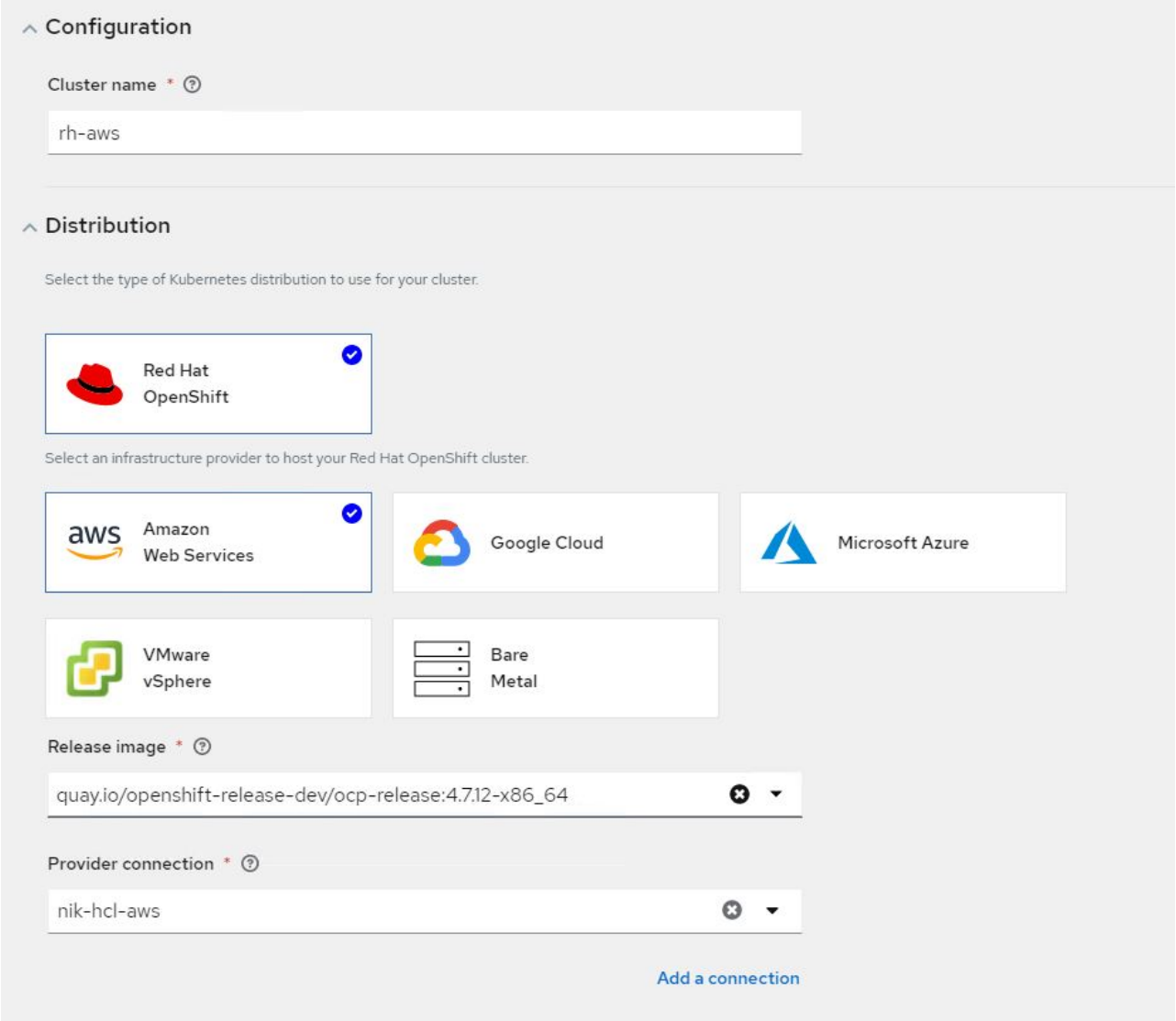
Gestion du cycle de vie des clusters

Pour gérer différents clusters OpenShift, vous pouvez les créer ou les importer dans Advanced Cluster Management.

1. Commencez par automatiser les infrastructures > clusters.
2. Pour créer un cluster OpenShift, effectuez les opérations suivantes :
   a. Créer une connexion fournisseur : accédez à connexions fournisseur et cliquez sur Ajouter une connexion, fournissez tous les détails correspondant au type de fournisseur sélectionné et cliquez sur Ajouter.

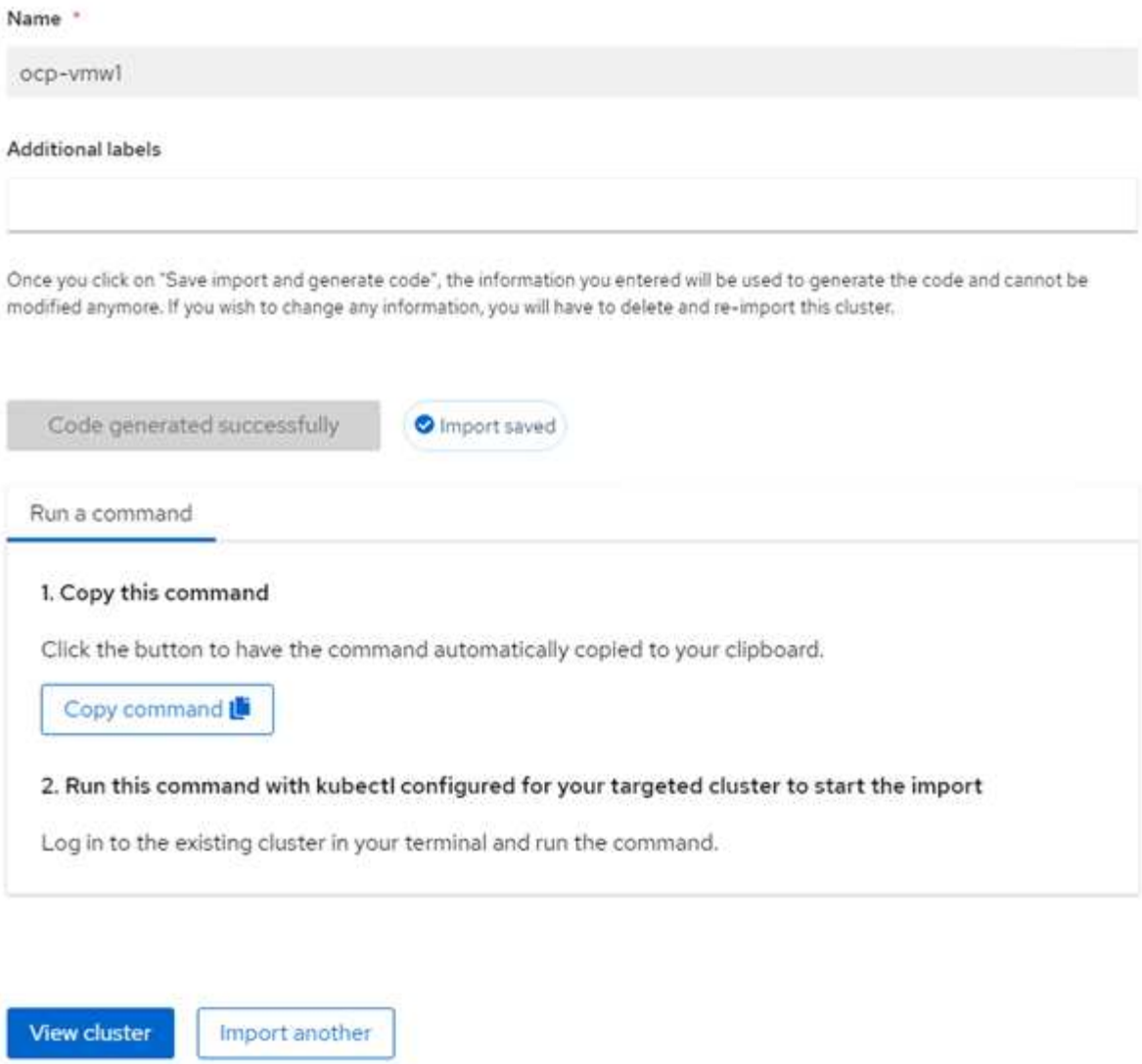


   b. Pour créer un nouveau cluster, accédez à clusters et cliquez sur Ajouter un cluster > Créer un cluster. Fournissez les détails du cluster et du fournisseur correspondant, puis cliquez sur Créer.



c. Une fois le cluster créé, il apparaît dans la liste des clusters avec l'état prêt.

3. Pour importer un cluster existant, procédez comme suit :
   a. Accédez à clusters et cliquez sur Ajouter un cluster > Importer un cluster existant.
   b. Entrez le nom du cluster, puis cliquez sur Enregistrer l'importation et générer le code. Une commande permettant d'ajouter le cluster existant est affichée.
   c. Cliquez sur Copy Command et exécutez la commande sur le cluster à ajouter au cluster Hub. Cette opération lance l'installation des agents nécessaires sur le cluster et, une fois ce processus terminé, le cluster apparaît dans la liste des clusters avec l'état prêt.



4. Une fois que vous avez créé et importé plusieurs clusters, vous pouvez les surveiller et les gérer à partir d'une seule console.

Fonctionnalités : gestion avancée des clusters pour Kubernetes sur Red Hat OpenShift avec NetApp

Gestion du cycle de vie des applications

Pour créer une application et la gérer dans un ensemble de clusters,

1. Accédez à gérer les applications dans la barre latérale et cliquez sur Créer une application. Indiquez les détails de l'application que vous souhaitez créer et cliquez sur Enregistrer.

Applications /

Create an application YAML: Off

Cancel Save

Name*

demo-app

Namespace*

default

Repository location for resources

Repository types

Select the type of repository where resources that you want to deploy are located

Git

URL*

https://github.com/open-cluster-management/acm-hive-openshift-releases.git

Branch

main

Path

clusterImageSets/fast/4.7

2. Une fois les composants de l'application installés, l'application apparaît dans la liste.

Applications

Refresh every 15s

Last update: 7:36:23 PM

Overview Advanced configuration

Create application

Search

Name	Namespace	Clusters	Resource	Time window	Created
demo-app	default	Local	Git		8 days ago

1 - 1 of 1 1 of 1

3. L'application peut désormais être contrôlée et gérée depuis la console.

Fonctionnalités : gestion avancée des clusters pour Kubernetes sur Red Hat OpenShift avec NetApp

Gouvernance et risque

Cette fonctionnalité vous permet de définir les stratégies de conformité des différents clusters et de vous assurer que ces clusters l'adhèrent. Vous pouvez configurer les règles pour les informer ou corriger toute déviation ou violation des règles.

1. Accédez à gouvernance et risque depuis la barre latérale.
2. Pour créer des stratégies de conformité, cliquez sur Créer une stratégie, entrez les détails des normes de stratégie et sélectionnez les clusters qui doivent respecter cette stratégie. Si vous souhaitez corriger automatiquement les violations de cette stratégie, cochez la case appliquer si elle est prise en charge, puis cliquez sur Créer.

Governance and risk / Policies /

Create policy YAML: Off

Name *

policy-complianceoperator

Namespace *

default

Specifications *

1 ✕ ComplianceOperator

Cluster selector

1 ✕ local-cluster: "true"

Standards

1 ✕ NIST-CSF

Categories

1 ✕ PR.IP Information Protection Processes and Procedures

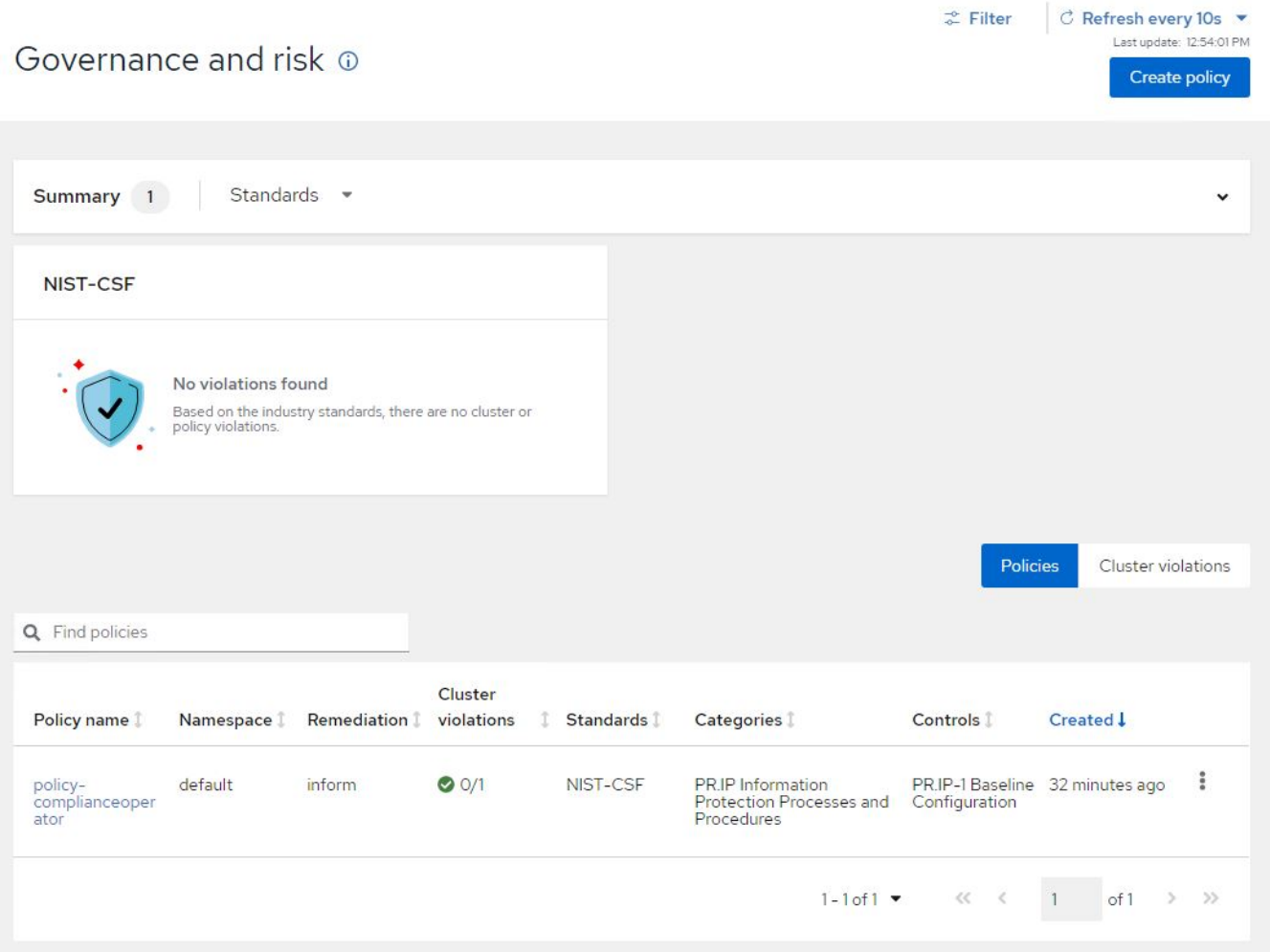
Controls

1 ✕ PR.IP-1 Baseline Configuration

☐ Enforce if supported

☐ Disable policy

3. Une fois toutes les règles requises configurées, toutes les violations des règles ou des clusters peuvent être surveillées et remédier aux problèmes dans Advanced Cluster Management.

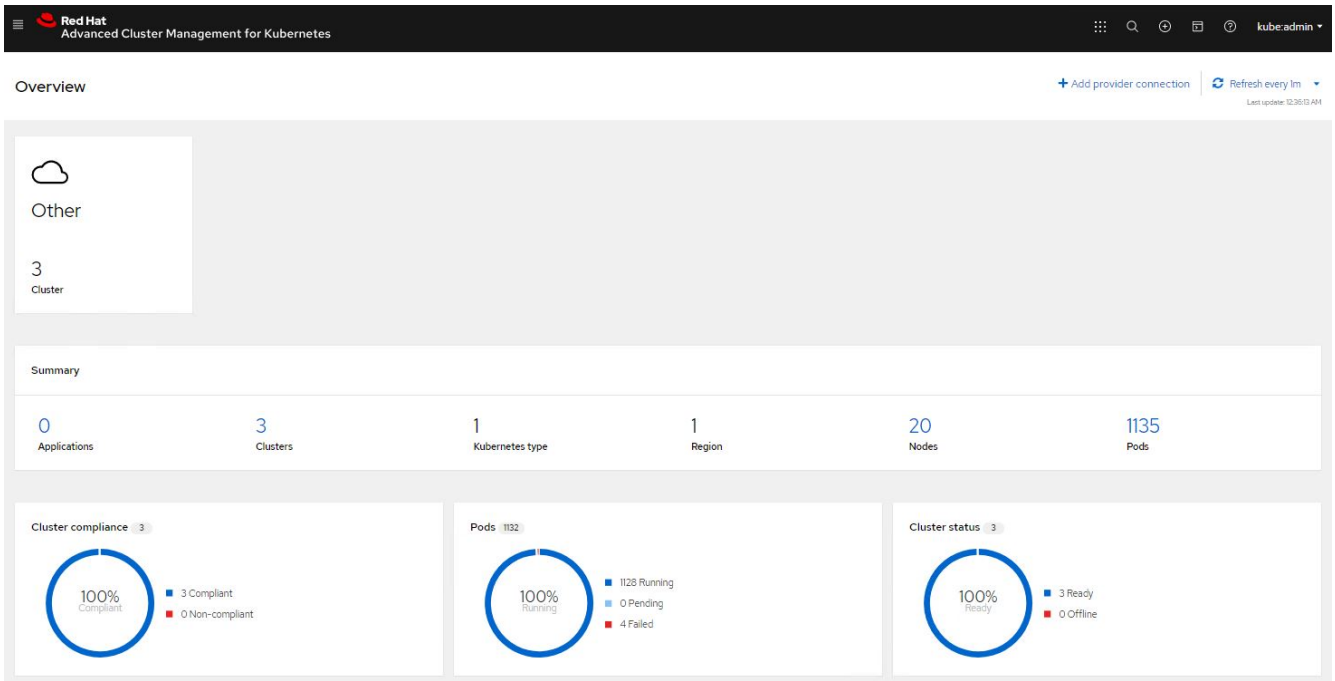


Fonctionnalités : gestion avancée des clusters pour Kubernetes sur Red Hat OpenShift avec NetApp

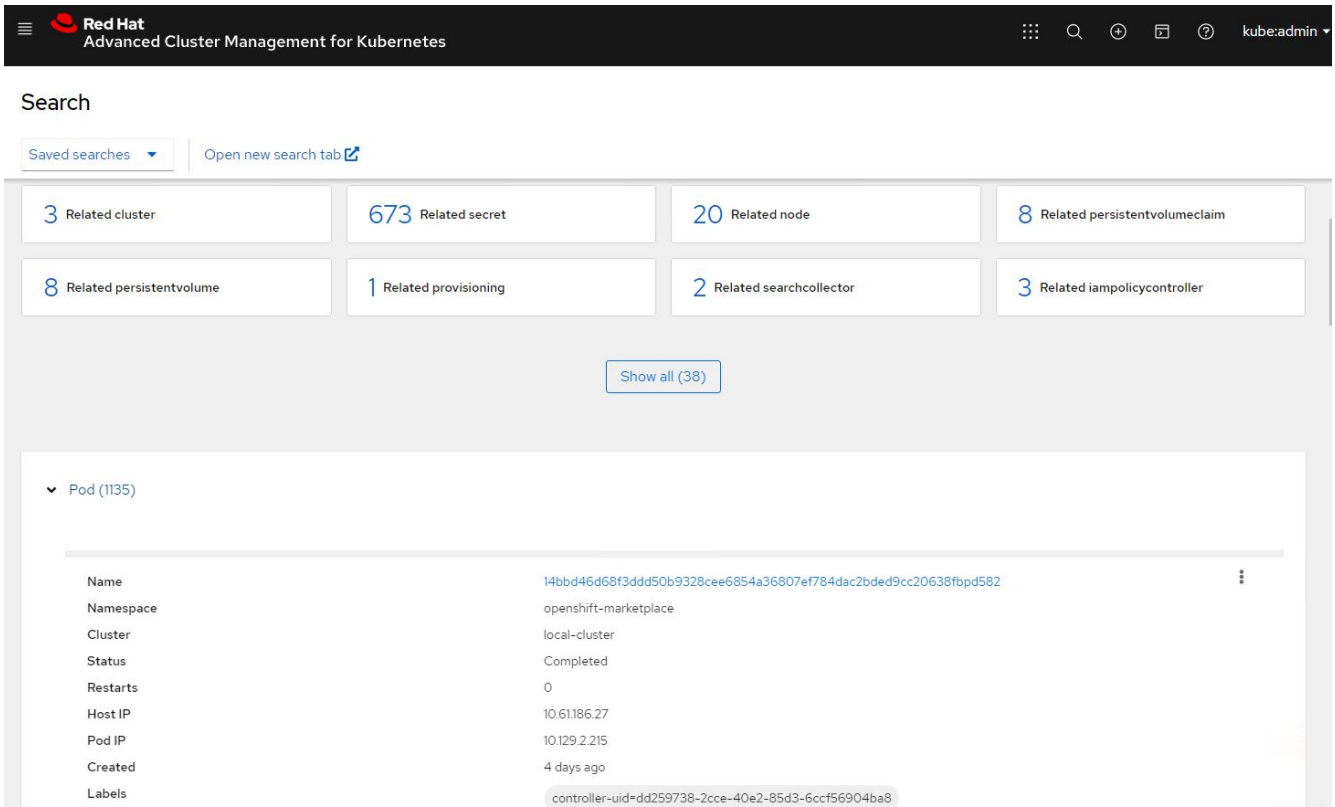
Observabilité

La solution Advanced Cluster Management pour Kubernetes fournit un moyen de surveiller les nœuds, les pods, les applications et les workloads dans l'ensemble des clusters.

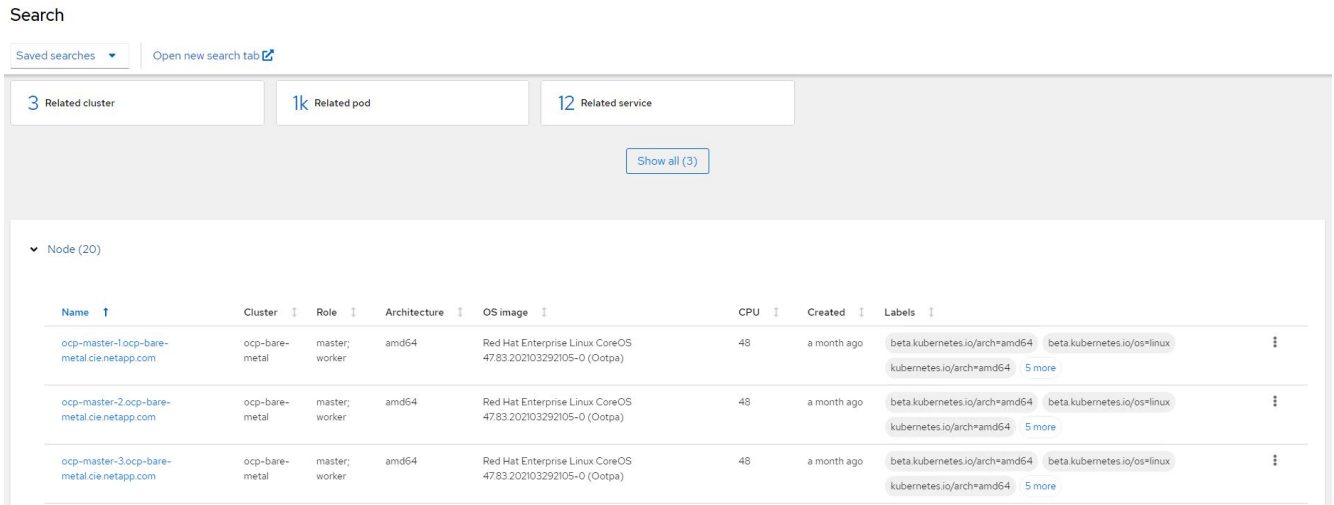
1. Naviguez jusqu'à observer les environnements > Présentation.



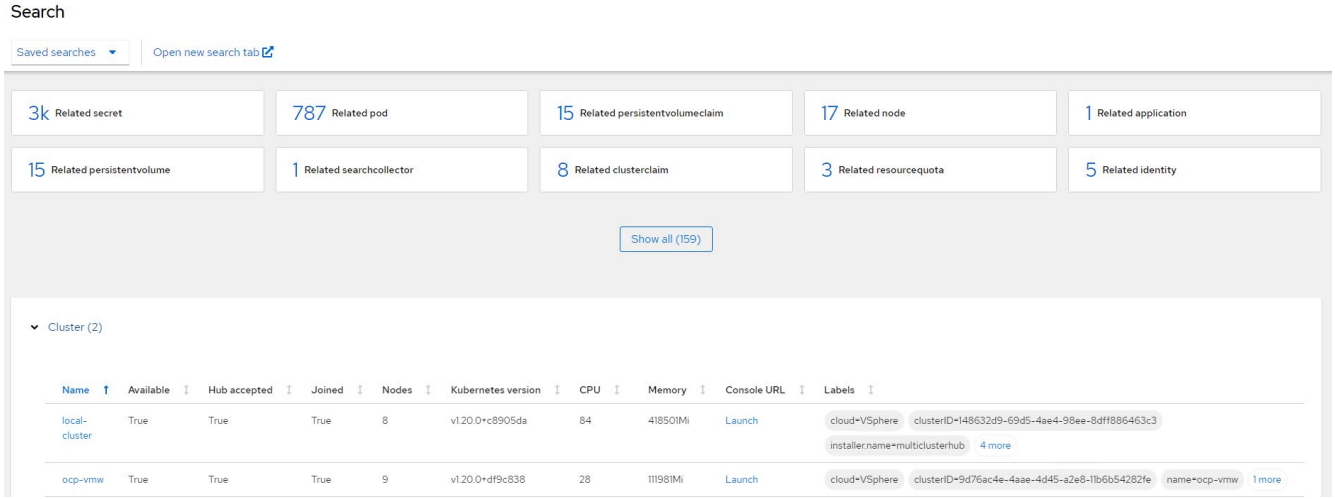
2. Tous les pods et les charges de travail dans tous les clusters sont surveillés et triés en fonction de différents filtres. Cliquez sur Pods pour afficher les données correspondantes.



3. Tous les nœuds des clusters sont surveillés et analysés en fonction de divers points de données. Cliquez sur nœuds pour obtenir plus d'informations sur les détails correspondants.



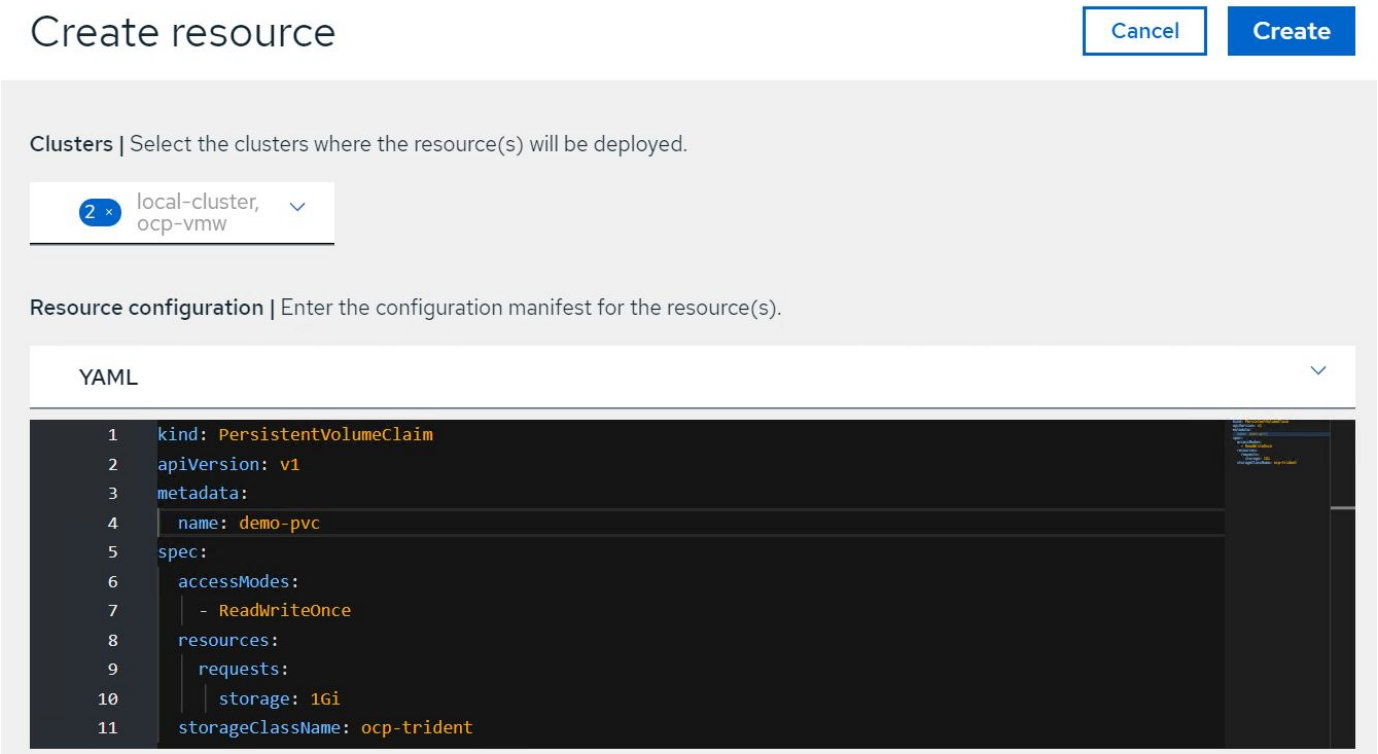
4. Tous les clusters sont surveillés et organisés en fonction de différents paramètres et ressources de cluster. Cliquez sur clusters pour afficher les détails du cluster.



Fonctionnalités : gestion avancée des clusters pour Kubernetes sur Red Hat OpenShift avec NetApp

Créer des ressources sur plusieurs clusters

La gestion avancée des clusters pour Kubernetes permet aux utilisateurs de créer des ressources sur un ou plusieurs clusters gérés simultanément à partir de la console. Par exemple, si vous disposez de clusters OpenShift sur différents sites et que ONTAP vous souhaitez provisionner des PVC sur les deux sites, vous pouvez cliquer sur le signe (+) de la barre d'onglets. Sélectionnez ensuite les clusters sur lesquels vous souhaitez créer la demande de volume persistant, collez la ressource YAML, puis cliquez sur Créer.



Vidéos et démonstrations : Red Hat OpenShift avec NetApp

Les vidéos suivantes présentent certaines des fonctionnalités décrites dans ce document :

Utilisation de Red Hat MTV pour migrer des machines virtuelles vers OpenShift Virtualization avec le stockage NetApp ONTAP

Accélérez le développement logiciel avec Astra Control et la technologie NetApp FlexClone - Red Hat OpenShift avec NetApp

Utilisez l'Astra de NetApp pour effectuer une analyse post-mortem et restaurer votre application

Protection des données dans un pipeline ci/CD avec Astra Control Center

Migration de workloads à l'aide d'Astra Control Center : Red Hat OpenShift avec NetApp

Migration des charges de travail - Red Hat OpenShift avec NetApp

Installation d'OpenShift Virtualization - Red Hat OpenShift avec NetApp

Déploiement d'une machine virtuelle avec OpenShift Virtualization - Red Hat OpenShift avec NetApp

NetApp HCI pour Red Hat OpenShift sur Red Hat Virtualization

Informations complémentaires : Red Hat OpenShift avec NetApp

Pour en savoir plus sur les informations fournies dans ce document, consultez les sites web suivants :

- Documentation NetApp

  "https://docs.netapp.com/"

- Documentation Trident d'Astra

  "https://docs.netapp.com/us-en/trident/index.html"

- Documentation NetApp Astra Control Center

  "https://docs.netapp.com/us-en/astra-control-center/"

- Documentation Red Hat OpenShift

  "https://access.redhat.com/documentation/en-us/openshift_container_platform/4.7/"

- Documentation Red Hat OpenStack Platform

  "https://access.redhat.com/documentation/en-us/red_hat_openstack_platform/16.1/"

- Documentation Red Hat Virtualization

  "https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.4/"

- Documentation VMware vSphere

  "https://docs.vmware.com/"