



Premiers pas avec le stockage et la virtualisation

NetApp virtualization solutions

NetApp
July 21, 2026

Sommaire

Premiers pas avec le stockage et la virtualisation	1
En savoir plus sur l'intégration de NetApp Trident avec Red Hat OpenShift	1
NetApp Trident	1
Trident Protect	1
NetApp ONTAP Modèles de matériel et protocoles pris en charge	2
Pilotes Trident pour le stockage ONTAP	3
Installez Trident sur un cluster OpenShift Red Hat et créez des objets de stockage	4
Étape 1 : Installer Trident	5
Étape 2 : Préparez le système de stockage et les fichiers de configuration StorageClass pour votre environnement	9
Étape 3 : Créez un fichier de configuration VolumeSnapshotClass	29
Étape 4 : Appliquez les fichiers de configuration à votre cluster	29
Étape 5 : Définir les classes de stockage et d'instantanés Trident par défaut	31
Conditions requises pour déployer Red Hat OpenShift Virtualization avec ONTAP	32
Prérequis	32
NetApp Trident CSI, profils de stockage, modes d'accès aux volumes pour la virtualisation OpenShift	33
Créez une machine virtuelle à l'aide du stockage ONTAP avec Red Hat OpenShift Virtualization	38
Créer VM	39
Démonstration vidéo	42
Créer une machine virtuelle avec Red Hat OpenShift Virtualization en utilisant le stockage ONTAP par des méthodes alternatives	42
Créer une machine virtuelle à partir d'un fichier YAML avec Red Hat OpenShift Virtualization	42
Créer une machine virtuelle à partir d'une copie instantanée avec Red Hat OpenShift Virtualization	45
Cloner une machine virtuelle dans OpenShift Virtualization à l'aide de Trident	49
Migrer une machine virtuelle entre deux nœuds dans un cluster Red Hat OpenShift	53
Migration en direct de machines virtuelles	53
Migration du stockage des machines virtuelles entre classes de stockage dans OpenShift Virtualization	55

Premiers pas avec le stockage et la virtualisation

En savoir plus sur l'intégration de NetApp Trident avec Red Hat OpenShift

Le stockage est crucial dans les applications conteneurisées, garantissant que les données sont disponibles de manière persistante et efficacement gérées dans des environnements dynamiques et souvent éphémères. Les solutions de stockage modernes pour conteneurs sont conçues pour évoluer de manière transparente avec les applications, offrant la flexibilité d'allouer dynamiquement les ressources de stockage en fonction des besoins des applications, tout en prenant en charge la croissance et la variation des charges de travail. Puisque la OpenShift Virtualization permet aux machines virtuelles de s'exécuter comme des conteneurs, elles utilisent le même stockage que les autres applications dans les clusters OpenShift.

NetApp ONTAP est un logiciel de gestion des données robuste et hautement évolutif qui constitue la base des solutions de stockage d'entreprise de NetApp. Il offre une architecture de stockage unifiée qui s'intègre parfaitement à divers environnements de stockage, notamment sur site, dans le cloud et dans des configurations de cloud hybride. ONTAP prend en charge une variété de protocoles tels que NFS, CIFS/SMB, iSCSI, FC et NVMe sur TCP, permettant une intégration fluide avec différentes applications et charges de travail. Il permet la consolidation du stockage SAN, NAS et objet sur une plateforme unique, simplifiant la gestion et réduisant les coûts. NetApp ONTAP est disponible sur site ainsi que sous forme de services gérés dans le cloud.

NetApp Trident

Trident est un orchestrateur de stockage open source développé par NetApp, conçu pour faciliter l'utilisation du stockage ONTAP dans les environnements conteneurisés. Il permet le provisionnement et la gestion dynamiques du stockage persistant pour les applications conteneurisées.

Fonctionnalités clés de Trident

Conformité CSI Garantit la compatibilité avec les dernières fonctionnalités et normes de stockage Kubernetes. **Configuration déclarative** Permet aux utilisateurs de définir les exigences de stockage dans des fichiers YAML, qui sont ensuite automatiquement gérés par Trident. **Pilotes** Trident prend en charge les pilotes pour les protocoles NFS, CIFS/SMB, iSCSI, FC et NVMe/TCP pris en charge par ONTAP. **Prise en charge du cloud hybride et multicloud** Trident prend en charge les pilotes pour le stockage ONTAP sur site et les services de stockage gérés chez les hyperscalers, à savoir FSx for NetApp ONTAP sur AWS, Azure NetApp Files sur Azure et Google Cloud NetApp Volumes sur Google Cloud. **Gestion avancée des données** Tire parti des fonctionnalités de snapshot et de clonage d'ONTAP pour une protection des données efficace et un provisionnement rapide des environnements de test et de développement.

Pour plus de détails sur Trident, consultez NetApp Trident ["documentation ici"](#)

Trident Protect

Trident Protect est une fonctionnalité avancée de Trident qui offre une protection des données et une sécurité renforcées pour les environnements conteneurisés. Elle est conçue pour garantir que les données stratégiques sont protégées contre les menaces potentielles et les scénarios de perte de données. Elle est disponible gratuitement pour tous les utilisateurs de Trident et peut être facilement activée dans n'importe quel cluster

OpenShift avec Trident installé.

Principales caractéristiques de Trident Protect

Sauvegardes automatisées Trident Protect permet des sauvegardes automatisées, pilotées par des politiques, des applications conteneurisées et des machines virtuelles, garantissant qu'elles sont régulièrement sauvegardées sans intervention manuelle.

Gestion des instantanés Elle exploite les capacités d'instantanés d'ONTAP pour créer des copies fréquentes et ponctuelles des applications conteneurisées et des machines virtuelles, fournissant ainsi un mécanisme de récupération fiable.

Chiffrement des données Garantit que les données sont chiffrées aussi bien au repos qu'en transit, répondant ainsi à des exigences de sécurité strictes et protégeant les informations sensibles.

Conformité et audit Fournit des outils et des fonctionnalités qui aident les organisations à respecter les exigences de conformité et à effectuer des audits réguliers de leurs pratiques en matière de protection des données.

Reprise après sinistre S'intègre aux fonctionnalités de réplication d'ONTAP pour fournir des solutions de reprise après sinistre robustes, garantissant la continuité des activités même en cas d'événements catastrophiques.

Pour plus de détails sur Trident Protect, veuillez consulter NetApp Trident Protect ["documentation ici"](#).

NetApp ONTAP Modèles de matériel et protocoles pris en charge

NetApp ONTAP fonctionne sur différents modèles de matériel, chacun conçu pour répondre à des exigences spécifiques en matière de performances et de capacité. Trident étend ses fonctionnalités robustes pour prendre en charge divers systèmes NetApp ONTAP, notamment All Flash FAS (AFF), All SAN Array (ASA) et ASA R2. Cette prise en charge garantit que les organisations peuvent exploiter pleinement le potentiel de leurs solutions de stockage haute performance dans des environnements conteneurisés.

Systèmes All Flash FAS (AFF) Les systèmes AFF offrent des performances exceptionnelles et une faible latence, ce qui les rend idéaux pour les applications à forte demande.

All SAN Array (ASA) Systems Les systèmes ASA sont conçus pour les environnements SAN dédiés, offrant des performances et une fiabilité robustes.

ASA R2 Systems Les systèmes ASA R2 offrent des fonctionnalités et des capacités améliorées pour les environnements SAN.

AFX NetApp AFX est une architecture de stockage 100 % Flash, désagrégée et spécialement conçue pour les charges de travail d'IA en entreprise. Elle offre un NAS haute performance, permettant une mise à l'échelle indépendante de la puissance de calcul et de la capacité. Les systèmes de stockage NetApp AFX se distinguent des autres systèmes basés sur ONTAP (ASA, AFF et FAS) par l'implémentation de leur couche de stockage. AFX utilise un système de fichiers distribué qui permet des performances et une évolutivité élevées, tandis que les autres systèmes basés sur ONTAP utilisent une architecture de stockage traditionnelle.

1. Protocoles pris en charge par AFF : NFS, CIFS/SMB, iSCSI, FC, NVMe/TCP
2. Protocoles pris en charge pour ASA : iSCSI, FC, NVMe/TCP
3. Protocoles pris en charge pour ASA R2 : iSCSI, FC, NVMe/TCP
4. Protocoles pris en charge pour AFX : à partir de Trident 25.10, seul le protocole NFS est pris en charge ; le protocole SMB n'est pas pris en charge.

Pilotes Trident pour le stockage ONTAP

Trident inclut les pilotes CSI suivants, chacun capable de provisionner un volume dans le stockage backend

Pour les protocoles NAS sur les modèles matériels pris en charge

1. **ontap-nas** : Un PVC correspond à un PV, qui est un flexvolume dédié.
2. **ontap-nas-economy** : Chaque PV provisionné est un qtree, avec un nombre configurable de qtrees par FlexVolume (200 par défaut, configurable entre 50 et 300). Un PVC correspond à un PV, qui est un qtree sur un flexvolume partagé.



Vous pouvez placer tous les disques des machines virtuelles sous forme de qtrees dans un seul volume. Cependant, veuillez noter que les fonctionnalités Snapshots, Clones et Replication ne sont pas prises en charge par Trident. Lorsque ces fonctionnalités sont gérées par l'administrateur de stockage, elles ne sont pas granulaires au niveau des PV.

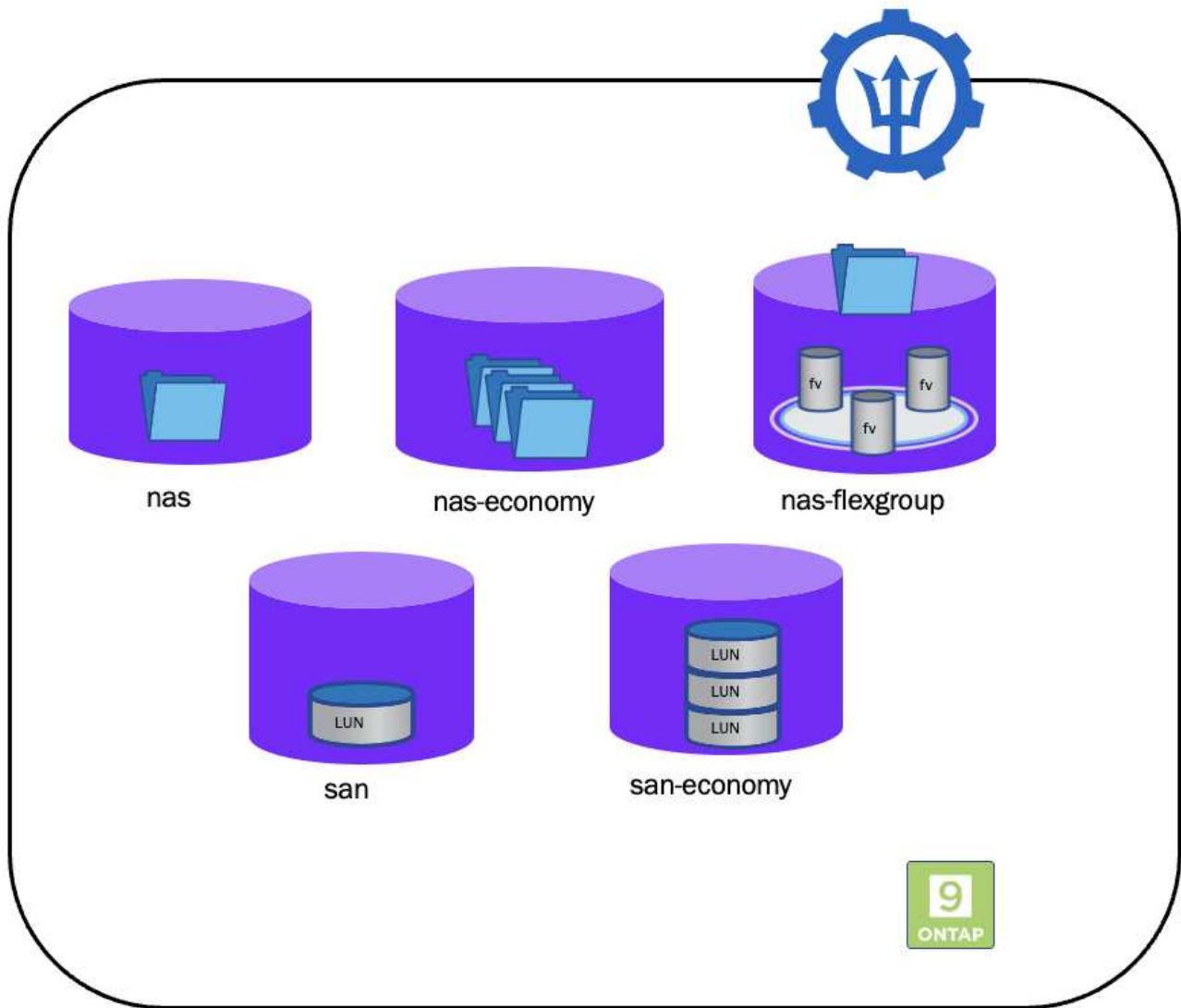
1. **ontap-nas-flexgroup** : Chaque PV provisionné est un ONTAP FlexGroup, et tous les agrégats affectés à un SVM sont utilisés.

Pour les protocoles SAN sur les modèles matériels pris en charge

1. **ontap-san** : Un PVC correspond à un PV, qui est un LUN sur un flexvolume dédié.
2. **ontap-san-economy** : Un PVC correspond à un PV, qui est un LUN sur un flexvolume partagé. Vous pouvez avoir plusieurs LUN dans le flexvolume partagé (100 par défaut, configurable entre 50 et 200 LUN/PV par volume flexvol).



Vous pouvez placer tous les disques des machines virtuelles sous forme de LUN dans un seul volume. Toutefois, bien que ce pilote prenne en charge les Snapshots et les Clones, il ne prend pas en charge la réplication. Lorsque la réplication est gérée par l'administrateur de stockage, elle n'est pas granulaire au niveau des PV.



Pour obtenir la liste complète des fonctionnalités exposées par les pilotes Trident et celles qui doivent être appliquées par l'administrateur de stockage, consultez la section "[Pilotes backend ONTAP](#)" dans la documentation Trident.

Installez Trident sur un cluster OpenShift Red Hat et créez des objets de stockage

Installez Trident avec l'opérateur Trident certifié Red Hat et créez des objets de stockage pour ONTAP et Amazon FSx for NetApp ONTAP afin d'activer le provisionnement dynamique de volumes pour les conteneurs et les machines virtuelles. Préparez les nœuds de travail pour l'accès bloc si nécessaire.

Avant de commencer

- Effectuez les procédures de cette page avant d'installer OpenShift Virtualization. OpenShift Virtualization nécessite un StorageClass et un VolumeSnapshotClass par défaut pris en charge par Trident pour créer des images de référence pour les modèles de machines virtuelles.
- Si vous avez déjà installé OpenShift Virtualization avant de configurer Trident, supprimez toutes les images

de référence créées avec une autre classe de stockage. Après avoir défini Trident comme stockage par défaut, OpenShift Virtualization recrée les images de référence en utilisant le stockage Trident.

```
oc delete dv,VolumeSnapshot -n openshift-virtualization-os-images
--selector=cdi.kubevirt.io/dataImportCron
```

Étape 1 : Installer Trident

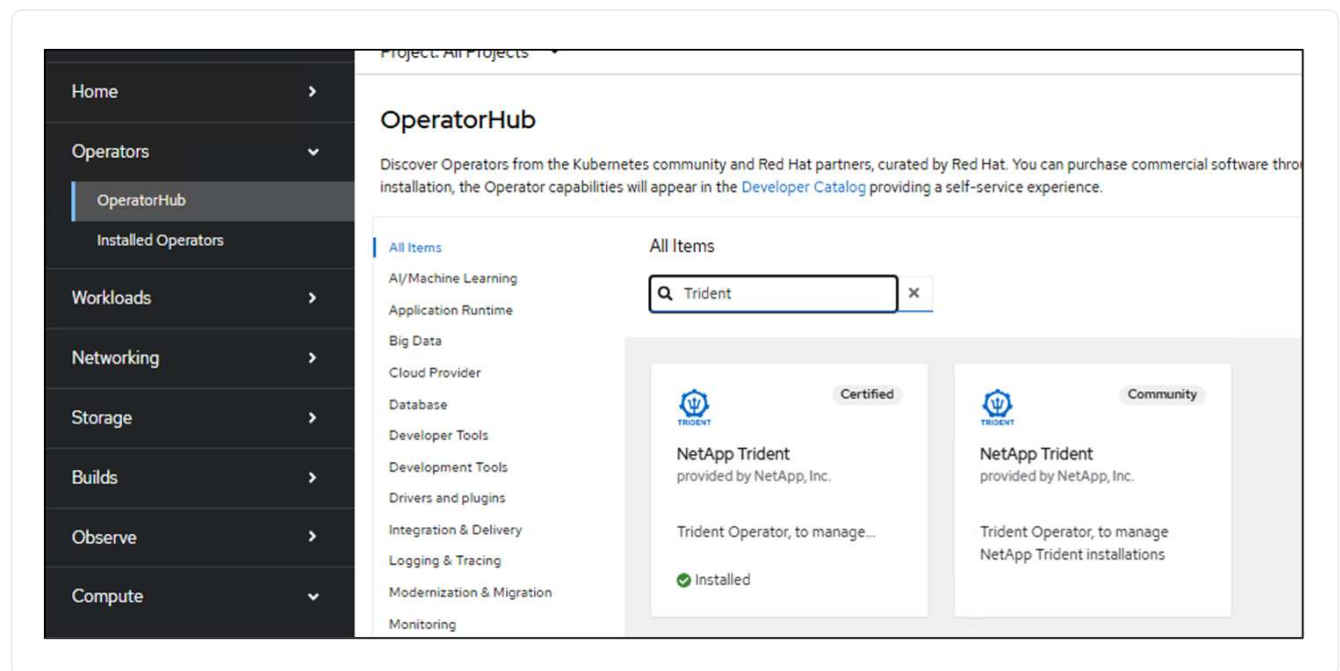
L'opérateur Trident certifié Red Hat est pris en charge par NetApp pour OpenShift sur site, dans les clouds publics et dans les services gérés tels que ROSA. À partir de Trident 25.02, l'opérateur peut également préparer les nœuds de travail pour iSCSI lorsque vous utilisez Amazon FSx for NetApp ONTAP et prévoyez d'exécuter des charges de travail de machines virtuelles OpenShift Virtualization.

Pour d'autres options d'installation, voir ["la documentation Trident"](#).

Étapes

1. Dans **OperatorHub**, sélectionnez **Certified NetApp Trident**.

Afficher un exemple



2. Sur la page **Install**, conservez la dernière version et sélectionnez **Install**.

Afficher un exemple

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 * ⓘ

stable

Version *

25.2.1

25.2.1

25.2.0

Operator will be available in all namespaces.

☐ A specific namespace on the cluster
This mode is not supported by this Operator

Installed Namespace *

PR openshift-operators

Update approval * ⓘ

☒ Automatic

☐ Manual

Install **Cancel**

3. Après l'installation de l'opérateur, sélectionnez **View operator** et créez une instance de Trident Orchestrator.

Si vous souhaitez préparer les nœuds de travail pour iSCSI, passez à la vue YAML et ajoutez `iscsi` à `nodePrep`.

Afficher un exemple

Create TridentOrchestrator

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

Configure via: ☐ Form view ☒ YAML view

```
1 kind: TridentOrchestrator
2 apiVersion: trident.netapp.io/v1
3 metadata:
4   name: trident
5 spec:
6   IPv6: false
7   debug: true
8   nodePrep:
9     - iscsi
10  imagePullSecrets: []
11  imageRegistry: ''
12  namespace: trident
13  silenceAutosupport: false
14
```

4. Confirmez que tous les pods Trident sont en cours d'exécution dans le cluster.

Afficher un exemple

```
[root@localhost ~]# oc get pods -n trident
NAME                                READY   STATUS    RESTARTS   AGE
trident-controller-84cb9bff89-lkx6k 6/6     Running   0           16h
trident-node-linux-d88b9             2/2     Running   0           16h
trident-node-linux-ld4b8             2/2     Running   0           16h
trident-node-linux-mj5r8             2/2     Running   0           16h
trident-node-linux-mkmmmp            2/2     Running   0           16h
trident-node-linux-qhgr7             2/2     Running   0           16h
trident-node-linux-vt9tp             2/2     Running   0           16h
[root@localhost ~]#
```

5. Si vous avez activé la préparation des nœuds iSCSI, connectez-vous aux nœuds de travail et vérifiez que `iscsid` et `multipathd` sont actifs et que `multipath.conf` contient des entrées.

Afficher un exemple

```
sh-5.1# systemctl status iscsid
● iscsid.service - Open-iSCSI
   Loaded: loaded (/usr/lib/systemd/system/iscsid.service; enabled; preset: disabled)
   Active: active (running) since Fri 2025-04-25 00:23:49 UTC; 3 days ago
 TriggeredBy: ● iscsid.socket
    Docs: man:iscsid(8)
          man:iscsiuio(8)
          man:iscsiadm(8)
   Main PID: 74787 (iscsid)
   Status: "Ready to process requests"
   Tasks: 1 (limit: 410912)
  Memory: 1.8M
     CPU: 6ms
   CGroup: /system.slice/iscsid.service
           └─74787 /usr/sbin/iscsid -f

Apr 25 00:23:49 ocp11-worker1 systemd[1]: Starting Open-iSCSI...
Apr 25 00:23:49 ocp11-worker1 systemd[1]: Started Open-iSCSI.
sh-5.1#
```

Afficher un exemple

```
sh-5.1# systemctl status multipathd
● multipathd.service - Device-Mapper Multipath Device Controller
   Loaded: loaded (/usr/lib/systemd/system/multipathd.service; enabled; preset: enabled)
   Active: active (running) since Fri 2025-04-25 00:23:50 UTC; 3 days ago
 TriggeredBy: ● multipathd.socket
   Process: 74905 ExecStartPre=/sbin/modprobe -a scsi_dh_alua scsi_dh_emc scsi_dh_rdac dm-multipath (code=exited, status=0/SUCCESS)
   Process: 74906 ExecStartPre=/sbin/multipath -A (code=exited, status=0/SUCCESS)
   Main PID: 74907 (multipathd)
   Status: "up"
   Tasks: 7
  Memory: 18.3M
     CPU: 23.008s
   CGroup: /system.slice/multipathd.service
           └─74907 /sbin/multipathd -d -s

Apr 25 00:23:50 ocp11-worker1 systemd[1]: Starting Device-Mapper Multipath Device Controller...
Apr 25 00:23:50 ocp11-worker1 multipathd[74907]: -----start up-----
Apr 25 00:23:50 ocp11-worker1 multipathd[74907]: read /etc/multipath.conf
Apr 25 00:23:50 ocp11-worker1 multipathd[74907]: path checkers start up
Apr 25 00:23:50 ocp11-worker1 systemd[1]: Started Device-Mapper Multipath Device Controller.
sh-5.1#
```

Afficher un exemple

```
sh-5.1# cat /etc/multipath.conf
defaults {
    find_multipaths no
}
blacklist {
    device {
        product .*
        vendor  .*
    }
}
blacklist_exceptions {
    device {
        product LUN
        vendor  NETAPP
    }
}
sh-5.1# █
```

Démonstration vidéo

La vidéo suivante présente une démonstration de l'installation de Trident à l'aide de l'opérateur Trident certifié Red Hat.

[Installation de Trident 25.02.1 à l'aide de l'opérateur Trident certifié dans OpenShift](#)

Étape 2 : Préparez le système de stockage et les fichiers de configuration StorageClass pour votre environnement

Créez des définitions TridentBackendConfig et StorageClass pour votre environnement. Vous pouvez configurer plusieurs protocoles de stockage dans votre environnement. Créez des fichiers YAML pour chaque protocole que vous souhaitez utiliser et remplacez les valeurs de remplacement par les détails spécifiques de votre configuration.



Remplissez soit la section sur site, soit la section ROSA ci-dessous en fonction de votre environnement, puis passez à l'étape 3.

Clusters OpenShift sur site

Créez des fichiers YAML pour chaque protocole que vous souhaitez configurer. Vous pouvez configurer un ou plusieurs des protocoles suivants : NAS pour le stockage de fichiers basé sur NFS, iSCSI pour le stockage bloc iSCSI, NVMe/TCP pour le stockage bloc NVMe over TCP haute performance, ou FC pour le stockage bloc Fibre Channel.

Créez un TridentBackendConfig et un StorageClass pour chaque protocole. La configuration du backend inclut des informations d'identification stockées dans un Secret Kubernetes.

Directives essentielles pour les options de configuration de base du backend

1. Les fichiers de configuration du backend font référence à votre **ONTAP SVM** et à votre **LIF de gestion**. +
2. Vous pouvez utiliser un **nom d'utilisateur et un mot de passe** pour vous authentifier auprès d'ONTAP. Il peut s'agir du nom d'utilisateur et du mot de passe d'administrateur de cluster ou d'administrateur de vserver. +
3. Vous pouvez également utiliser un **certificat client** pour l'authentification avec ONTAP. Pour plus de détails sur la génération du certificat client, de la clé privée et du certificat d'autorité de certification, veuillez consulter "[Documentation Trident ici](#)".

Directives essentielles pour les options de configuration du backend pour le pilote NAS economy

Les paramètres suivants peuvent être inclus pour la configuration du **NAS economy driver** :

1. **qtreesPerFlexvol** La valeur doit être comprise entre 50 et 300, la valeur par défaut est 200. +
2. **limitVolumePoolSize** La valeur correspond à la taille maximale de FlexVol que vous demandez lors de l'utilisation de qtrees dans le backend ontap-nas-economy. Aucune valeur par défaut n'est définie et ce paramètre n'est pas appliqué par défaut. Si vous définissez ce paramètre, l'opérateur Trident crée un nouveau FlexVol lorsque le FlexVol actuel atteint la taille spécifiée. Ce paramètre peut vous aider à contrôler l'utilisation de la capacité de votre cluster en limitant la taille des FlexVols utilisés pour le provisionnement des PV. +
3. **denyNewVolumePools** Ce paramètre empêche les backends ontap-nas-economy de créer de nouveaux volumes FlexVol® pour contenir leurs qtrees. Seuls les FlexVol® préexistants sont utilisés pour le provisionnement de nouveaux PV.

Directives essentielles pour les options de configuration du backend pour le pilote SAN

Les paramètres suivants peuvent être inclus pour la configuration du **pilote SAN** :

1. **limitVolumeSize** Cette valeur correspond à la taille maximale de volume que vous demandez lors de l'utilisation d'un backend SAN. Aucune valeur par défaut n'est définie et ce paramètre n'est pas appliqué par défaut. Si vous définissez ce paramètre, l'opérateur Trident échouera lors du provisionnement si la taille de volume demandée dépasse cette valeur. Vous pouvez utiliser ce paramètre pour restreindre la taille maximale des volumes qu'il gère pour les LUN. +
2. **lunsPerFlexvol** Nombre maximal de LUN par FlexVol®, doit être compris entre 50 et 200, la valeur par défaut étant 100. Si vous définissez ce paramètre, l'opérateur Trident crée un nouveau FlexVol lorsque le FlexVol actuel atteint le nombre de LUN spécifié. Ce paramètre peut vous aider à contrôler l'utilisation de la capacité de votre cluster en limitant le nombre de LUN par FlexVol utilisé pour le provisionnement des PV.

Recommandations essentielles pour les options de configuration du backend pour le pilote SAN economy

Les paramètres suivants peuvent être inclus pour la configuration du **SAN economy driver** :

1. **limitVolumePoolSize** La valeur correspond à la taille maximale de FlexVol® que vous demandez lors de l'utilisation du backend SAN economy. Aucune valeur par défaut n'est définie et ce paramètre n'est pas appliqué par défaut. Si vous définissez ce paramètre, l'opérateur Trident limite la taille maximale de FlexVol® qui peut être créée pour la création de LUN dans le FlexVol®. Ce paramètre peut vous aider à contrôler l'utilisation de la capacité dans votre cluster en limitant la taille des FlexVol® utilisés pour le provisionnement de PV. +
2. **denyNewVolumePools** Ce paramètre empêche les back-ends SAN economy de créer de nouveaux volumes FlexVol® pour contenir leurs LUN. Seuls les FlexVol® préexistants sont utilisés pour le provisionnement de nouveaux PV.

Directives essentielles concernant les options de configuration du backend pour le provisionnement des volumes



Dans les exemples de fichiers YAML fournis pour chaque protocole, si storagePrefix n'est pas spécifié, le préfixe par défaut « trident » est utilisé. Lorsque vous utilisez ontap-nas-economy et un storagePrefix de 24 caractères ou plus, les qtrees n'incluent pas le préfixe de stockage, bien qu'il figure dans le nom du volume.

Lorsque vous créez un volume dans ONTAP directement depuis le gestionnaire système ou depuis l'interface de ligne de commande (CLI), sans fournir de valeurs pour tous les paramètres optionnels, certaines valeurs par défaut sont définies automatiquement lors de la mise en service d'un volume, comme indiqué ci-dessous.

```
NSOL-NetApp-A70-T19U05::> volume create -vserver oc-test-automation -volume test_ontap_default -aggregate NSOL_NetApp_A70_T19U05a_NVME_SSD_1 -size 10GB
[Job 22836] Job succeeded: Successful
```

```
NSOL-NetApp-A70-T19U05::> volume show -vserver oc-test-automation -volume test_ontap_default -fields snapshot-policy,qos-adaptive-policy-group,snapshot-reserve-available,tiering-policy,unix-permissions,snapdir-access,space-guarantee
vserver      volume      unix-permissions  space-guarantee  snapdir-access  snapshot-policy  qos-adaptive-policy-group  snapshot-reserve-available  tiering-policy
oc-test-automation test_ontap_default ---rwxr-xr-x      none             true             default          -                  511.8MB                  none
```

Voici un exemple de volume créé à l'aide de la configuration du backend Trident. Aucune option n'est explicitement définie dans la section des valeurs par défaut, à l'exception du modèle de nom.

```
oc describe tbc -n trident tbc-nas
### output truncated for brevity
Spec:
  Backend Name:  tbc-nas
  Credentials:
    Name:  tbc-nas-secret
  Defaults: # There are no defaults set except for the volume name
  template.
    Name Template:      {{ .config.StoragePrefix }}_{{
.volume.Namespace }}_{{ .volume.RequestName }}
  Management LIF:      10.192.102.50
  Storage Driver Name:  ontap-nas
  Storage Prefix:      testautomation
  Svm:                  oc-test-automation
```

Lorsque le volume est créé à l'aide de cette configuration backend, les options qui ne sont pas définies dans le fichier de configuration backend sont définies sur les valeurs par défaut, comme indiqué ci-dessous.

```
NSOL-NetApp-A70-T19U05::> volume show -vserver oc-test-automation -volume testautomation_default_test_trident_defaults_91653 -fields snapshot-policy,qos-adaptive-policy-group,snapshot-reserve-available,tiering-policy,unix-permissions,snapdir-access,space-guarantee
vserver      volume      unix-permissions  space-guarantee  snapdir-access  snapshot-policy  qos-adaptive-policy-group  snapshot-reserve-available  tiering-policy
oc-test-automation testautomation_default_test_trident_defaults_91653 ---rwxrwxrwx      none             false            none          -                  0B                  none
```

Cependant, lorsque vous définissez des valeurs dans la section par défaut de la configuration du backend Trident, vous pouvez créer des volumes qui peuvent avoir des valeurs spécifiques pour ces paramètres. Dans l'exemple suivant, snapshotPolicy est définie sur la stratégie par défaut (elle est « none

» si vous ne définissez rien explicitement) et snapshotDir est définie sur « true » dans la section par défaut du fichier de configuration yaml du backend. Avec ces paramètres, vous pouvez constater que le volume du backend reçoit ces valeurs. (snapdir-access est « true » et snapshot-policy est « default »)

```
[root@00-50-56-b5-01-5b pvcs-br]# oc describe tbc -n trident tbc-nas
Name:          tbc-nas
Namespace:     trident
Labels:        <none>
Annotations:   <none>
API Version:   trident.netapp.io/v1
Kind:          TridentBackendConfig
Metadata:
  Creation Timestamp:  2026-03-30T20:06:59Z
  Finalizers:
    trident.netapp.io
  Generation:          3
  Resource Version:    81358056
  UID:                 4dc24339-ff66-4345-acc1-37ec73d14140
Spec:
  Backend Name:  tbc-nas
  Credentials:
    Name:  tbc-nas-secret
  Defaults:
    Name Template:  {{ .config.StoragePrefix }}_{{ .volume.Namespace }}_{{ .volume.RequestName }}
    Snapshot Dir:   true
    Snapshot Policy: default
    Management LIF: 10.192.102.50
    Storage Driver Name: ontap-nas
    Storage Prefix:  testautomation
    Svm:              oc-test-automation
    Version:           1
Status:
  Backend Info:
    Backend Name:  tbc-nas
    Backend UUID:  210cc722-b394-489d-8e9e-49018d3e8710
    Deletion Policy: delete
    Last Operation Status: Success
    Message:          Backend 'tbc-nas' updated
    Phase:            Bound
```

```
NSOL-NetApp-A70-T19U05:> volume show -vserver oc-test-automation -volume testautomation_default_test_trident_set_defaults_7567a -fields snapshot-policy,qos-adaptive-policy-group,snapshot-reserve-available,tiering-policy,unix-permissions,snapdir-access,space-guarantee
vserver      volume                                     unix-permissions space-guarantee snapdir-access snapshot-policy qos-adaptive-policy-group snapshot-reserve-availab
le tiering-policy
-----
oc-test-automation testautomation_default_test_trident_set_defaults_7567a ---rwxrwxrwx none true default - 269.5MB
none
```

```
NSOL-NetApp-A70-T19U05:> volume show -vserver oc-test-automation -volume testautomation_default_test_trident_set_defaults_7567a -fields snapshot-policy,snapdir-access
vserver      volume                                     snapdir-access snapshot-policy
oc-test-automation testautomation_default_test_trident_set_defaults_7567a true default
```

Pour tous les autres paramètres supplémentaires, veuillez consulter la documentation Trident :

1. ["Options de configuration du backend NAS Trident"](#) +
2. ["Options de configuration du backend SAN Trident"](#)

NAS

Créez un `TridentBackendConfig` et un `StorageClass` pour ONTAP NAS afin d'activer le provisionnement de stockage persistant basé sur NFS. La configuration du backend inclut des informations d'identification stockées dans un secret Kubernetes et fait référence à votre ONTAP SVM et à votre LIF de gestion.

Exemple de secret backend et de fichier de configuration backend utilisant la méthode d'authentification par nom d'utilisateur et mot de passe (enregistrez sous `tbc-nas.yaml`) :

```
# tbc-nas.yaml
apiVersion: v1
kind: Secret
metadata:
  name: tbc-nas-secret
type: Opaque
stringData:
  username: <cluster admin username>
  password: <cluster admin password>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: tbc-nas
spec:
  version: 1
  storageDriverName: ontap-nas
  managementLIF: <ONTAP management LIF>
  backendName: tbc-nas
  svm: zoneb #<replace with your SVM name>
  storagePrefix: testzoneb #<replace with your prefix>
  defaults:
    nameTemplate: "{{ .config.StoragePrefix }}_{{ .volume.Namespace }}"
    credentials:
      name: tbc-nas-secret
```

Exemple de secret backend et de fichier de configuration backend utilisant la méthode d'authentification par certificat client (enregistrez sous `tbc-nas.yaml`) :


```
# tbc-nas.yaml
apiVersion: v1
kind: Secret
metadata:
  name: ontap-nas-secret
  namespace: trident
type: Opaque
stringData:
  clientPrivateKey: <client private>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: tbc-nas
  namespace: trident
spec:
  version: 1
  storageDriverName: ontap-nas
  managementLIF: <ONTAP management LIF>
  backendName: tbc-nas
  svm: zoneb #<replace with your SVM name>
  storagePrefix: testzoneb #<replace with your prefix>
  credentials:
    name: ontap-nas-secret
    clientCertificate: <client certificate>
```

StorageClass definition (enregistrer sous sc-nas.yaml):

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

NAS économique

Créez un TridentBackendConfig et un StorageClass pour le pilote d'économie NAS ONTAP afin d'activer le provisionnement de stockage persistant basé sur NFS à l'aide de qtrees. La configuration du backend

inclut des informations d'identification stockées dans un secret Kubernetes et fait référence à votre SVM ONTAP et à votre LIF de gestion.

Exemple de secret backend et de fichier de configuration backend utilisant la méthode d'authentification par nom d'utilisateur et mot de passe (enregistrez sous `tbc-nas-economy.yaml`) :

```
# tbc-nas-economy.yaml
apiVersion: v1
kind: Secret
metadata:
  name: tbc-nas-economy-secret
type: Opaque
stringData:
  username: <cluster admin username>
  password: <cluster admin password>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: tbc-nas-economy
spec:
  version: 1
  storageDriverName: ontap-nas-economy
  managementLIF: <ONTAP management LIF>
  backendName: tbc-nas-eco
  svm: <replace with your SVM name>
  storagePrefix: <replace with your prefix>
  defaults:
    nameTemplate: "{{ .config.StoragePrefix }}_{{ .volume.Namespace
  }}_{{ .volume.RequestName }}"
  credentials:
    name: tbc-nas-economy-secret
```

StorageClass definition (enregistrer sous `sc-nas-economy.yaml`):

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



Veuillez consulter la section « Directives essentielles pour les options de configuration du backend du pilote NAS economy » ci-dessus pour plus de détails sur les paramètres supplémentaires pouvant être inclus dans le fichier de configuration du pilote ontap-nas-economy.

SAN iSCSI

Créez un TridentBackendConfig et un StorageClass pour ONTAP SAN avec iSCSI afin d'activer le provisionnement de stockage bloc basé sur iSCSI. La configuration du backend utilise le pilote ontap-san et inclut des informations d'identification stockées dans un Secret Kubernetes. Le paramètre sanType utilise par défaut le protocole iSCSI s'il est omis.

Secret backend et fichier de configuration backend utilisant la méthode d'authentification par nom d'utilisateur et mot de passe (enregistrez sous `tbc-iscsi.yaml`) :

```

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

```

Secret backend et fichier de configuration backend utilisant la méthode d'authentification par certificat client (enregistrez sous `tbc-iscsi.yaml`):

```
# tbc-iscsi.yaml
apiVersion: v1
kind: Secret
metadata:
  name: backend-tbc-ontap-iscsi-secret
type: Opaque
stringData:
  clientPrivateKey: <client private key>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: ontap-iscsi
spec:
  version: 1
  storageDriverName: ontap-san
  managementLIF: <management LIF>
  backendName: ontap-iscsi
  svm: <SVM name>
  credentials:
    name: backend-tbc-ontap-iscsi-secret
    clientCertificate: <client certificate>
```

StorageClass definition (enregistrer sous `sc-iscsi.yaml`):

```
# sc-iscsi.yaml
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: sc-iscsi
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-san"
  media: "ssd"
  provisioningType: "thin"
  fsType: ext4
  snapshots: "true"
allowVolumeExpansion: true
```

SAN NVMe/TCP

Créez un `TridentBackendConfig` et un `StorageClass` pour ONTAP SAN avec NVMe sur TCP afin d'activer le provisionnement de stockage bloc haute performance. La configuration du backend utilise le pilote `ontap-san` optimisé pour le transport NVMe/TCP et inclut des informations d'identification stockées dans un `Secret` Kubernetes.

Secret backend et fichier de configuration backend utilisant la méthode d'authentification par nom d'utilisateur et mot de passe (enregistrez sous `tbc-nvme.yaml`) :

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

Secret backend et fichier de configuration backend utilisant la méthode d'authentification par certificat client (enregistrez sous `tbc-nvme.yaml`) :

```
# tbc-nvme.yaml
apiVersion: v1
kind: Secret
metadata:
  name: backend-tbc-ontap-nvme-secret
type: Opaque
stringData:
  clientPrivateKey: <client private key>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-tbc-ontap-nvme
spec:
  version: 1
  storageDriverName: ontap-san
  sanType: nvme
  managementLIF: <ONTAP management LIF>
  backendName: backend-tbc-ontap-nvme
  svm: <SVM name>
  credentials:
    name: backend-tbc-ontap-nvme-secret
    clientCertificate: <client certificate>
```

StorageClass definition (enregistrer sous `sc-nvme.yaml`):

```
# sc-nvme.yaml
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: sc-nvme
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-san"
  media: "ssd"
  provisioningType: "thin"
  fsType: ext4
  snapshots: "true"
allowVolumeExpansion: true
```



Si vous prévoyez d'utiliser des volumes ReadWriteMany (RWX) avec le protocole NVMe, un seul volume NVMe RWX ne peut pas être monté par plus de 64 nœuds dans un cluster Kubernetes. Pour plus d'informations sur cette limitation et sur le modèle de super-sous-système introduit dans Trident 26.02 ainsi que sur le modèle de sous-système par volume antérieur à Trident 26.02, consultez la documentation Trident : <https://docs.netapp.com/us-en/trident/trident-use/ontap-san-prep.html#nvme-tcp-considerations>.

FC SAN

Créez un `TridentBackendConfig` et un `StorageClass` pour ONTAP SAN avec Fibre Channel afin d'activer le provisionnement du stockage bloc basé sur FC. La configuration du backend utilise le pilote `ontap-san` avec le protocole FCP spécifié et inclut des informations d'identification stockées dans un `Secret` Kubernetes.

Utilisez la méthode d'authentification par nom d'utilisateur/mot de passe ou la méthode d'authentification par certificat client pour vous authentifier auprès d'ONTAP. Pour plus d'informations sur la génération du certificat client, de la clé privée et du certificat d'autorité de certification, veuillez consulter "[Documentation Trident ici](#)". Des exemples pour les deux méthodes sont fournis ci-dessous.

Afficher un exemple

Secret backend et fichier de configuration backend utilisant la méthode d'authentification par nom d'utilisateur et mot de passe ONTAP (enregistrez sous `tbc-fc.yaml`) :

```
# tbc-fc.yaml
apiVersion: v1
kind: Secret
metadata:
  name: tbc-fc-secret
type: Opaque
stringData:
  username: <cluster admin username>
  password: <cluster admin password>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: tbc-fc
spec:
  version: 1
  storageDriverName: ontap-san
  managementLIF: <ONTAP management LIF>
  backendName: tbc-fc
  svm: openshift-fc #<replace with your SVM name>
  sanType: fcp
  storagePrefix: demofc #<replace with your prefix>
  defaults:
    nameTemplate: "{{ .config.StoragePrefix }}"_{{ .volume.Namespace
  }}_{{ .volume.RequestName }}"
  credentials:
    name: tbc-fc-secret
```

Secret backend et fichier de configuration backend utilisant la méthode d'authentification par certificat client (enregistrez sous `tbc-fc.yaml`) :

```
# tbc-fc.yaml
apiVersion: v1
kind: Secret
metadata:
  name: tbc-fc-secret
type: Opaque
stringData:
  clientPrivateKey: <client private key>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: tbc-fc
spec:
  version: 1
  storageDriverName: ontap-san
  managementLIF: <ONTAP management LIF>
  backendName: tbc-fc
  svm: openshift-fc #<replace with your SVM name>
  sanType: fcp
  storagePrefix: demofc #<replace with your prefix>
  defaults:
    nameTemplate: "{{ .config.StoragePrefix }}_{{ .volume.Namespace
  }}_{{ .volume.RequestName }}"
  credentials:
    name: tbc-fc-secret
    clientCertificate: <client certificate>
```

```
# sc-fc.yaml
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: sc-fc
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-san"
  media: "ssd"
  provisioningType: "thin"
  fsType: ext4
  snapshots: "true"
allowVolumeExpansion: true
```

Créez un `TridentBackendConfig` et un `StorageClass` pour ONTAP SAN economy avec iSCSI afin d'activer le provisionnement du stockage bloc basé sur iSCSI. La configuration du backend utilise le pilote `ontap-san-economy` avec le protocole iSCSI et inclut des informations d'identification stockées dans un `Secret` Kubernetes.

Secret backend et fichier de configuration backend utilisant la méthode d'authentification par certificat client (enregistrez sous `tbc-san-eco.yaml`) :

```
# tbc-san-eco.yaml
---
apiVersion: v1
kind: Secret
metadata:
  name: ontap-san-cert-auth
  namespace: trident
type: Opaque
stringData:
  clientPrivateKey: <client private key>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: tbc-san-eco-cert
  namespace: trident
spec:
  version: 1
  storageDriverName: ontap-san-economy
  managementLIF: "<ONTAP management LIF>"
  backendName: tbc_san_eco
  svm: <replace with your SVM name>
  storagePrefix: <replace with your prefix>
  credentials:
    name: ontap-san-cert-auth
    clientCertificate: <client certificate>
```

```
# sc-san-eco.yaml
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: sc-san-eco
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-san-economy"
  provisioningType: "thin"
  snapshots: "true"
```

Consultez la documentation Trident pour obtenir des exemples supplémentaires de fichiers yaml, ainsi que des informations sur les modes d'accès et les modes de volume pris en charge par les pilotes SAN et NAS.

1. ["Exemples utilisant des pilotes NAS"](#)
2. ["Exemples utilisant des pilotes SAN"](#)

Clusters ROSA avec Amazon FSx for NetApp ONTAP

Créez des fichiers YAML pour chaque protocole que vous souhaitez configurer. Vous pouvez configurer un ou les deux protocoles suivants : NAS pour le stockage de fichiers basé sur NFS ou iSCSI pour le stockage bloc.

NAS

Créez une `TridentBackendConfig` et une `StorageClass` pour Amazon FSx for NetApp ONTAP avec ONTAP NAS afin d'activer le provisionnement de stockage persistant basé sur NFS sur les clusters ROSA. La configuration backend utilise les noms DNS d'Amazon FSx for NetApp ONTAP pour les LIF de gestion et de données, et inclut des informations d'identification stockées dans un `Secret` Kubernetes dans l'espace de noms `trident`.

Secret backend et fichier de configuration backend (à enregistrer sous `tbc-fsx-nas.yaml`):

```
# tbc-fsx-nas.yaml
apiVersion: v1
kind: Secret
metadata:
  name: backend-fsx-ontap-nas-secret
type: Opaque
stringData:
  username: <FSx for ONTAP, for example fsxadmin>
  password: <FSx for ONTAP password>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: backend-fsx-ontap-nas
spec:
  version: 1
  backendName: fsx-ontap
  storageDriverName: ontap-nas
  managementLIF: <Management DNS name>
  dataLIF: <NFS DNS name>
  svm: <SVM NAME>
  credentials:
    name: backend-fsx-ontap-nas-secret
```

StorageClass definition (enregistrer sous `sc-fsx-nas.yaml`):

```
# sc-fsx-nas.yaml (storage class name is trident-csi)
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: trident-csi
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-nas"
  fsType: "ext4"
allowVolumeExpansion: true
reclaimPolicy: Retain
```

iSCSI

Créez un TridentBackendConfig et un StorageClass pour Amazon FSx for NetApp ONTAP avec ONTAP SAN afin d'activer le provisionnement de stockage bloc basé sur iSCSI sur les clusters ROSA. La configuration du backend utilise le pilote ontap-san et inclut des informations d'identification stockées dans un Secret Kubernetes. Assurez-vous que les nœuds de travail sont préparés pour l'accès iSCSI.

Secret backend et fichier de configuration backend (à enregistrer sous `tbc-fsx-iscsi.yaml`):

```
# tbc-fsx-iscsi.yaml
apiVersion: v1
kind: Secret
metadata:
  name: backend-tbc-fsx-iscsi-secret
type: Opaque
stringData:
  username: <FSx for ONTAP, for example fsxadmin>
  password: <FSx for ONTAP password>
---
apiVersion: trident.netapp.io/v1
kind: TridentBackendConfig
metadata:
  name: fsx-iscsi
spec:
  version: 1
  storageDriverName: ontap-san
  managementLIF: <Management DNS name>
  backendName: fsx-iscsi
  svm: <SVM name>
  credentials:
    name: backend-tbc-fsx-iscsi-secret
```

StorageClass definition (enregistrer sous `sc-fsx-iscsi.yaml`):

```
# sc-fsx-iscsi.yaml
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: sc-fsx-iscsi
provisioner: csi.trident.netapp.io
parameters:
  backendType: "ontap-san"
  media: "ssd"
  provisioningType: "thin"
  fsType: ext4
  snapshots: "true"
allowVolumeExpansion: true
```

Étape 3 : Créez un fichier de configuration VolumeSnapshotClass

Créez une définition VolumeSnapshotClass pour les déploiements sur site et ROSA. Cette configuration active les opérations basées sur les instantanés pour les volumes persistants.

VolumeSnapshotClass definition (enregistrer sous `snapshot-class.yaml`):

```
# snapshot-class.yaml
apiVersion: snapshot.storage.k8s.io/v1
kind: VolumeSnapshotClass
metadata:
  name: trident-snapshotclass
driver: csi.trident.netapp.io
deletionPolicy: Retain
```

Étape 4 : Appliquez les fichiers de configuration à votre cluster

Appliquez les fichiers de configuration que vous avez créés lors des étapes précédentes à votre OpenShift cluster.

Étapes

1. Appliquez les fichiers TridentBackendConfig et StorageClass pour chaque protocole que vous avez configuré.

Pour les clusters sur site :

```
oc create -f tbc-nas.yaml -n trident
oc create -f sc-nas.yaml

oc create -f tbc-nas-economy.yaml -n trident
oc create -f sc-nas-economy.yaml

oc create -f tbc-iscsi.yaml -n trident
oc create -f sc-iscsi.yaml

oc create -f tbc-nvme.yaml -n trident
oc create -f sc-nvme.yaml

oc create -f tbc-fc.yaml -n trident
oc create -f sc-fc.yaml
```

Pour les clusters ROSA :

```
oc create -f tbc-fsx-nas.yaml -n trident
oc create -f sc-fsx-nas.yaml

oc create -f tbc-fsx-iscsi.yaml -n trident
oc create -f sc-fsx-iscsi.yaml
```

2. Appliquez la configuration VolumeSnapshotClass.

```
oc create -f snapshot-class.yaml
```

3. Vérifiez que les ressources ont été créées avec succès.

Vérifiez les objets TridentBackendConfig :

```
oc get tbc -n trident
```

Vérifiez les objets StorageClass :

```
oc get storageclass
```

Vérifiez VolumeSnapshotClass :

```
oc get volumesnapshotclass
```


Étape 5 : Définir les classes de stockage et d'instantanés Trident par défaut

Définissez la StorageClass Trident et la VolumeSnapshotClass Trident comme valeurs par défaut dans le cluster OpenShift. Ceci est nécessaire pour que la virtualisation OpenShift puisse créer des sources d'images de référence pour les modèles de machines virtuelles.

Étapes

1. Définissez le Trident StorageClass par défaut.

Définissez un StorageClass pris en charge par Trident comme valeur par défaut du cluster afin que les PersistentVolumeClaims l'utilisent automatiquement lorsqu'aucune classe de stockage n'est spécifiée. Vous devez configurer deux annotations : une pour la valeur par défaut du cluster et une spécifique à la virtualisation OpenShift.

- a. Définissez l'annotation StorageClass par défaut à l'échelle du cluster.

Assurez-vous qu'une seule StorageClass est définie par défaut. Si une autre StorageClass est déjà définie par défaut, définissez son annotation sur false.

Dans la console, modifiez l'annotation :

```
storageclass.kubernetes.io/is-default-class: "true"
```

Depuis l'interface de ligne de commande :

```
kubectl patch storageclass <storage class name> -p '{"metadata":  
{"annotations":{"storageclass.kubernetes.io/is-default-  
class":"true"}}}'
```

- b. Définissez l'annotation par défaut spécifique à la virtualisation OpenShift.

OpenShift Virtualization utilise une annotation spécifique qui prime sur l'`is-default-class` annotation générale du cluster. Si une autre StorageClass est déjà définie par défaut, définissez son annotation sur false.

Dans la console, modifiez l'annotation :

```
storageclass.kubevirt.io/is-default-virt-class: "true"
```

Depuis l'interface de ligne de commande :

```
kubectl patch storageclass <storage class name> -p '{"metadata":  
{"annotations":{"storageclass.kubevirt.io/is-default-virt-class":  
"true"}}}'
```

2. Définissez la VolumeSnapshotClass Trident par défaut.

Définissez un VolumeSnapshotClass pris en charge par Trident comme valeur par défaut du cluster pour activer les opérations basées sur les instantanés pour les volumes persistants. Cela garantit que les VolumeSnapshots utilisent automatiquement le pilote CSI Trident lorsqu'aucune classe d'instantané n'est spécifiée et permet à la virtualisation OpenShift de créer des instantanés d'images de référence.

Assurez-vous qu'une seule VolumeSnapshotClass est définie par défaut. Si une autre VolumeSnapshotClass est déjà définie par défaut, définissez son annotation sur false.

Dans la console, modifiez l'annotation :

```
snapshot.storage.kubernetes.io/is-default-class: "true"
```

Depuis l'interface de ligne de commande :

```
oc patch volumesnapshotclass <snapshot class name> --type=merge -p
'{"metadata":{"annotations":{"snapshot.storage.kubernetes.io/is-default-
class":"true"}}}'
```

Conditions requises pour déployer Red Hat OpenShift Virtualization avec ONTAP

Consultez les prérequis et les détails supplémentaires pour déployer la virtualisation OpenShift avec les systèmes de stockage ONTAP.

Prérequis

- Un cluster Red Hat OpenShift (version ultérieure à la version 4.6) installé sur une infrastructure bare-metal avec des nœuds de travail RHCOS
- Déployer des contrôles de santé des machines pour maintenir la haute disponibilité des machines virtuelles
- Un cluster NetApp ONTAP , avec SVM configuré avec le protocole correct.
- Trident installé sur le cluster OpenShift
- Une configuration backend Trident créée
- Une StorageClass configurée sur le cluster OpenShift avec Trident comme provisionneur
- Un Trident VolumeSnapshotClass configuré sur le cluster OpenShift avec Trident comme provisionneur
- Trident StorageClass configuré par défaut pour le cluster OpenShift ou OpenShift Virtualization.

Pour les prérequis Trident ci-dessus, voir ["Section d'installation du Trident"](#) pour plus de détails.

- Accès administrateur du cluster au cluster Red Hat OpenShift
- Accès administrateur au cluster NetApp ONTAP
- Un poste de travail administrateur avec les outils tridentctl et oc installés et ajoutés à \$PATH

Comme la virtualisation OpenShift est gérée par un opérateur installé sur le cluster OpenShift, elle impose une

surcharge supplémentaire sur la mémoire, le processeur et le stockage, dont il faut tenir compte lors de la planification des besoins matériels du cluster. Consultez la documentation ["ici"](#) pour plus de détails.

Vous pouvez également spécifier un sous-ensemble des nœuds du cluster OpenShift pour héberger les opérateurs, les contrôleurs et les machines virtuelles 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 le stockage prenant en charge OpenShift Virtualization, NetApp recommande d'avoir une StorageClass dédiée qui demande du stockage à partir d'un backend Trident particulier, qui à son tour est pris en charge par une SVM dédiée. Cela maintient un niveau de multilocation en ce qui concerne les données servies pour les charges de travail basées sur des machines virtuelles sur le cluster OpenShift.

NetApp Trident CSI, profils de stockage, modes d'accès aux volumes pour la virtualisation OpenShift

Dans OpenShift Virtualization, les sources de démarrage sont des images disque utilisées pour provisionner des machines virtuelles (VM) à partir de modèles. Elles sont généralement stockées sous forme de DataVolumes ou de Persistent Volume Claims (PVC) dans l'espace de noms openshift-virtualization-os-images. Pour que OpenShift Virtualization fonctionne correctement, deux tâches de configuration du stockage sont obligatoires avant d'installer OpenShift Virtualization à l'aide de l'opérateur :

1. Vous devez configurer une StorageClass par défaut pour le cluster. Sinon, la virtualisation OpenShift ne pourra pas importer automatiquement les images sources de démarrage.
2. Vous devez configurer des profils de stockage si votre fournisseur de stockage n'est pas reconnu par CDI. Un profil de stockage fournit des paramètres de stockage recommandés en fonction du StorageClass associé. L'une des fonctions essentielles de StorageProfile est de définir les stratégies de clonage pour un StorageClass. Cela détermine comment les volumes sont clonés (par exemple, à l'aide de snapshots, de clones CSI ou de copies assistées par l'hôte), en optimisant les performances et l'utilisation des ressources selon les capacités du fournisseur de stockage sous-jacent. Le pilote CSI Trident est reconnu par CDI, et un profil de stockage par défaut est automatiquement créé pour tout StorageClass Trident créé dans le cluster. La stratégie de clonage de ce profil de stockage par défaut est définie sur snapshot, qui est la stratégie de clonage recommandée pour le pilote CSI Trident. Vous devez créer le VolumeSnapshotClass et le StorageClass pour Trident et définir le StorageClass comme valeur par défaut avant d'installer OpenShift Virtualization. Cela garantit que les images sources de démarrage sont disponibles en tant que VolumeSnapshots dans l'espace de noms openshift-virtualization-os-images, afin de pouvoir provisionner efficacement les machines virtuelles sans erreur.

Pour certains systèmes d'exploitation pris en charge (comme RHEL, Fedora ou CentOS), OpenShift Virtualization fournit et met à jour automatiquement les sources de démarrage. Celles-ci sont téléchargées lors de l'installation de OpenShift Virtualization et elles sont gérées et mises à jour automatiquement vers la dernière version du système d'exploitation. Si un système d'exploitation n'est pas fourni automatiquement, un administrateur doit préparer et attacher manuellement une image personnalisée. Lorsque vous utilisez NetApp Trident comme CSI, vous bénéficiez des avantages supplémentaires suivants :

1. Les images sources de démarrage seront disponibles sous forme d'instantanés dans l'espace de noms openshift-virtualization-os-images. Lorsque les machines virtuelles sont provisionnées, l'image de démarrage est clonée à partir de l'instantané, ce qui permet au démarrage de la machine virtuelle d'être beaucoup plus rapide.
2. Un profil de stockage est automatiquement créé pour chaque Trident StorageClass créé et est automatiquement configuré avec la stratégie de clonage définie sur snapshot. Le mode d'accès est défini sur ReadWriteMany. Aucune configuration manuelle n'est nécessaire.

Il est essentiel que vous créiez un Trident StorageClass et un Trident VolumeSnapshotClass et que vous les

définissiez comme valeurs par défaut avant d'installer la virtualisation OpenShift dans votre cluster. Cela garantira que les images sources de démarrage seront disponibles sous forme de VolumeSnapshots dans l'espace de noms openshift-virtualization-os-images, afin de pouvoir approvisionner les machines virtuelles efficacement et sans erreur.

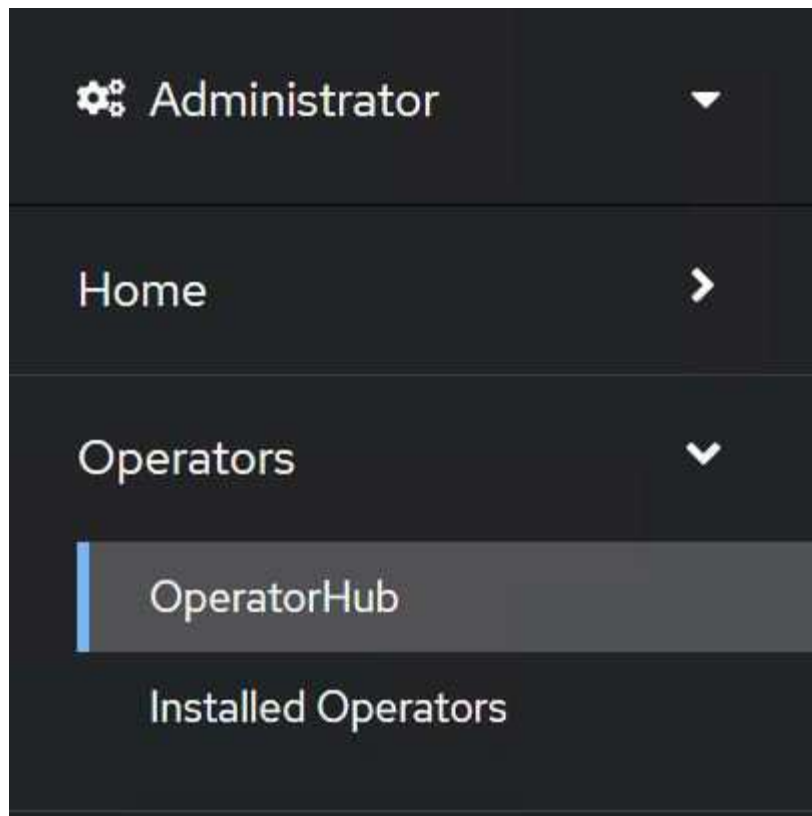
Un autre avantage de l'utilisation de Trident CSI : . Dans OpenShift Virtualization, les disques de démarrage des machines virtuelles doivent disposer des modes d'accès RWX pour permettre la migration à chaud des machines virtuelles. Trident prend en charge le mode d'accès **ReadWriteMany** pour les volumes provisionnés à l'aide des protocoles NAS ainsi que des protocoles SAN (lorsque volumeMode est défini sur Block et non sur FileSystem). Vous pouvez utiliser l'un ou l'autre protocole pour le disque de démarrage. L'objet de profil de stockage sera automatiquement configuré en fonction des paramètres StorageClass et les modes d'accès du disque de démarrage seront correctement définis lors du provisionnement des machines virtuelles.

Après avoir défini le Trident StorageClass comme valeur par défaut pour le cluster, vous pouvez créer des StorageClasses supplémentaires pour Trident avec des paramètres différents, et ils seront automatiquement reconnus par OpenShift Virtualization, avec des profils de stockage appropriés créés pour eux. Vous pouvez ensuite spécifier le StorageClass à utiliser pour le disque de démarrage et les disques de données de la machine virtuelle dans la définition de la VM.

L'installation de OpenShift Virtualization sur le cluster OpenShift est un processus simple utilisant l'opérateur OpenShift Virtualization disponible dans le OpenShift OperatorHub. Pour des étapes détaillées sur la façon d'installer OpenShift Virtualization, consultez la documentation ["ici"](#).

Installez OpenShift Virtualization sur un cluster bare-metal Red Hat OpenShift. Cette procédure comprend la connexion avec l'accès administrateur du cluster, la navigation vers OperatorHub et l'installation de l'opérateur OpenShift Virtualization.

1. Connectez-vous au cluster bare-metal Red Hat OpenShift avec un accès administrateur de cluster.
2. Sélectionnez Administrateur dans la liste déroulante Perspective.
3. Accédez à Opérateurs > OperatorHub et recherchez OpenShift Virtualization.



4. Sélectionnez la mosaïque Virtualisation OpenShift et cliquez sur Installer.



OpenShift Virtualization

2.6.2 provided by Red Hat

✕

Install

Latest version

2.6.2

Capability level

- ☒ Basic Install
- ☒ Seamless Upgrades
- ☒ Full Lifecycle
- ☐ Deep Insights
- ☐ Auto Pilot

Provider type

Red Hat

Provider

Red Hat

Requirements

Your cluster must be installed on bare metal infrastructure with Red Hat Enterprise Linux CoreOS workers.

Details

OpenShift Virtualization extends Red Hat OpenShift Container Platform, allowing you to host and manage virtualized workloads on the same platform as container-based workloads. From the OpenShift Container Platform web console, you can import a VMware virtual machine from vSphere, create new or clone existing VMs, perform live migrations between nodes, and more. You can use OpenShift Virtualization to manage both Linux and Windows VMs.

The technology behind OpenShift Virtualization is developed in the [KubeVirt](#) open source community. The KubeVirt project extends [Kubernetes](#) by adding additional virtualization resource types through [Custom Resource Definitions](#) (CRDs). Administrators can use Custom Resource Definitions to manage [VirtualMachine](#) resources alongside all other resources that Kubernetes provides.

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

Update channel *

- ☐ 2.1
- ☐ 2.2
- ☐ 2.3
- ☐ 2.4
- ☒ stable

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** openshift-cnv

i Namespace creation

Namespace **openshift-cnv** does not exist and will be created.

- ☐ Select a Namespace

Approval strategy *

- ☒ Automatic
- ☐ Manual

Install

Cancel

 OpenShift Virtualization
provided by Red Hat

Provided APIs

HC OpenShift
Virtualization
Deployment **Required**

Represents the deployment of
OpenShift Virtualization

6. Attendez que l'installation de l'opérateur soit terminée.

 OpenShift Virtualization
2.6.2 provided by Red Hat

Installing Operator

The Operator is being installed. This may take a few minutes.

[View installed Operators in Namespace openshift-cnv](#)

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



Installed operator - operand required

The Operator has installed successfully. Create the required custom resource to be able to use this Operator.

HC HyperConverged **Required**

Creates and maintains an OpenShift Virtualization Deployment

Create HyperConverged

[View installed Operators in Namespace openshift-cnv](#)

8. Sur l'écran Créer HyperConverged, cliquez sur Créer, en acceptant tous les paramètres par défaut. Cette étape démarre l'installation d'OpenShift Virtualization.

Name *

Labels

Infra >

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

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.

9. Une fois que tous les pods passent à l'état En cours d'exécution dans l'espace de noms openshift-cnv et que l'opérateur de virtualisation OpenShift est à l'état Réussi, l'opérateur est prêt à être utilisé. Les machines virtuelles peuvent désormais être créées 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
 OpenShift Virtualization 2.6.2 provided by Red Hat	 openshift-cnv	 Succeeded Up to date
		Last updated: May 18, 8:02 pm Provided APIs: OpenShift Virtualization Deployment , HostPathProvisioner deployment

Créez une machine virtuelle à l'aide du stockage ONTAP avec Red Hat OpenShift Virtualization

Créez une machine virtuelle avec OpenShift Virtualization. Cette procédure comprend la sélection d'un modèle de système d'exploitation, la configuration des StorageClasses, et la personnalisation des paramètres de la machine virtuelle afin de répondre à des

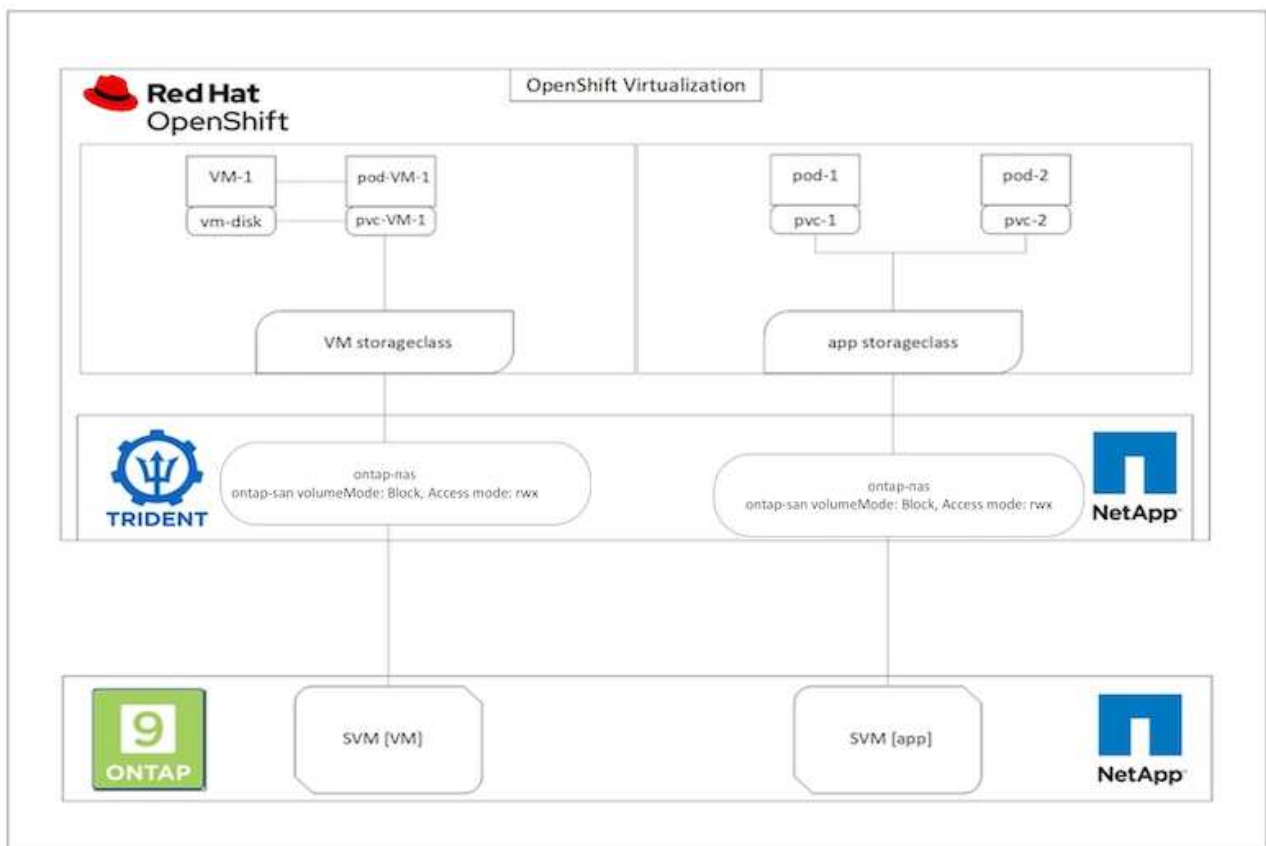
exigences spécifiques. Comme prérequis, vous devez avoir déjà créé le backend Trident, le StorageClass et les objets de classe de snapshot de volume. Vous pouvez consulter la "[Section d'installation du Trident](#)" pour plus de détails.

Créer VM

Les machines virtuelles sont des déploiements avec état qui nécessitent des volumes pour héberger le système d'exploitation et les données. Avec CNV, étant donné que les machines virtuelles sont exécutées en tant que pods, elles sont sauvegardées par des PV hébergés sur NetApp ONTAP via Trident. Ces volumes sont attachés en tant que disques et stockent l'intégralité du système de fichiers, y compris la source de démarrage de la machine virtuelle.



Trident StorageClass doit être défini comme StorageClass par défaut sur le cluster pour garantir que les PVC créés pour les disques des machines virtuelles soient provisionnés à l'aide du backend Trident. Vous pouvez avoir plusieurs StorageClasses sur le cluster, mais le StorageClass par défaut doit être utilisé pour le provisionnement du PVC des disques de démarrage des machines virtuelles. Des disques supplémentaires peuvent utiliser d'autres StorageClasses si nécessaire.



Pour créer rapidement une machine virtuelle sur le cluster OpenShift, procédez comme suit :

1. Accédez à Virtualisation > Machines virtuelles et cliquez sur Créer.
2. Sélectionner à partir du modèle.
3. Sélectionnez le système d'exploitation souhaité pour lequel la source de démarrage est disponible.

4. Cochez la case Démarrer la machine virtuelle après la création.
5. Cliquez sur Créer rapidement une machine virtuelle.

La machine virtuelle est créée, démarrée et passe à l'état **En cours d'exécution**. Elle crée automatiquement un PVC et un PV correspondant pour le disque de démarrage en utilisant le StorageClass par défaut. Pour pouvoir migrer la machine virtuelle à chaud ultérieurement, vous devez vous assurer que le StorageClass utilisé pour les disques peut prendre en charge les volumes RWX. Il s'agit d'une exigence pour la migration à chaud. ontap-nas et ontap-san (volumeMode block pour les protocoles iSCSI et NVMe/TCP) peuvent prendre en charge les modes d'accès RWX pour les volumes créés à l'aide des StorageClasses respectifs.

Pour configurer ontap-san StorageClass sur le cluster, consultez le "[Section pour la configuration de StorageClass](#)".



Cliquer sur Création rapide VirtualMachine utilisera la StorageClass par défaut pour créer le PVC et le PV du disque racine amorçable de la machine virtuelle. Vous pouvez sélectionner une autre StorageClass pour le disque, en sélectionnant Personnaliser VirtualMachine > Personnaliser les paramètres de VirtualMachine > Disques, puis en modifiant le disque pour utiliser la StorageClass requise.

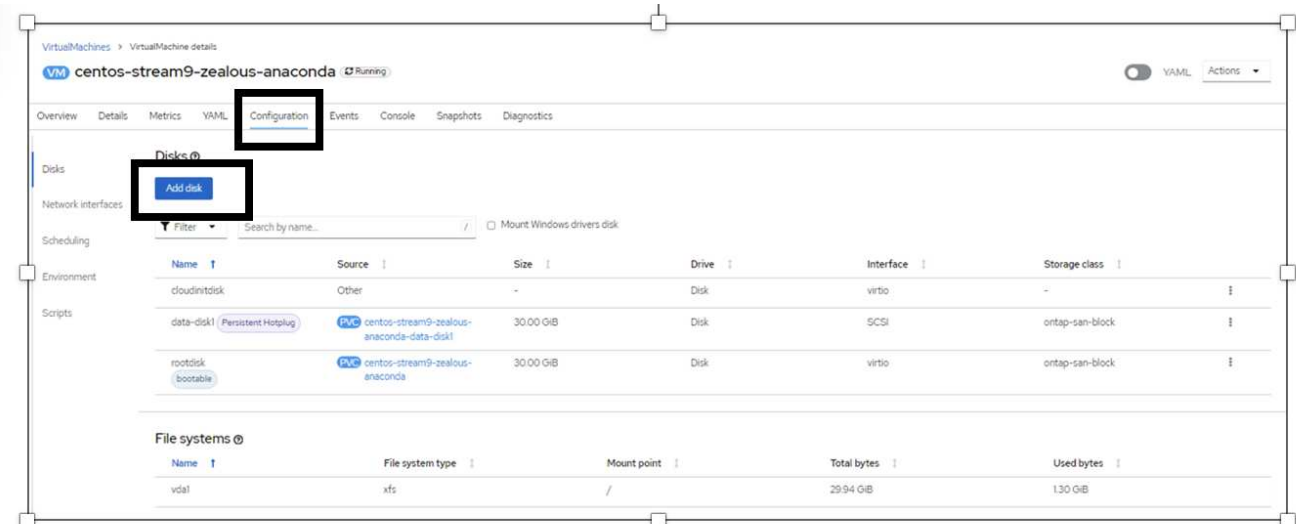
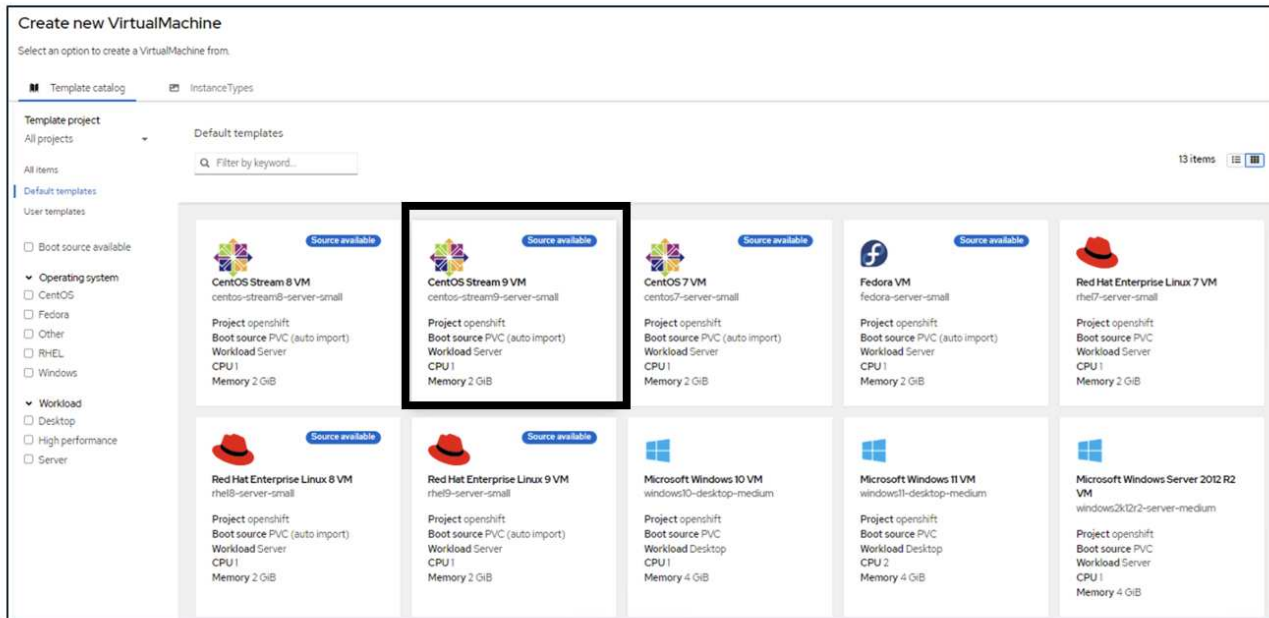
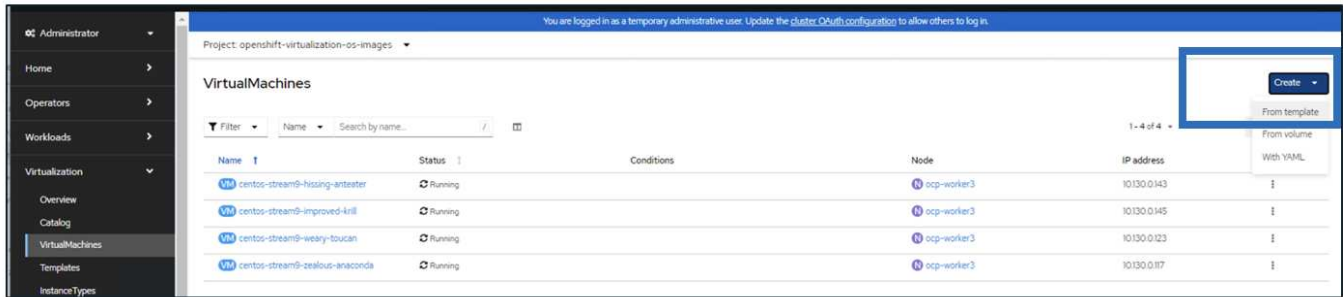
En règle générale, le mode d'accès par bloc est préféré aux systèmes de fichiers lors du provisionnement des disques de machine virtuelle.

Pour personnaliser la création de la machine virtuelle après avoir sélectionné le modèle de système d'exploitation, cliquez sur Personnaliser la machine virtuelle au lieu de Créer rapidement.

1. Si le système d'exploitation sélectionné a une source de démarrage configurée, vous pouvez cliquer sur **Personnaliser les paramètres de la machine virtuelle**.
2. Si le système d'exploitation sélectionné n'a pas de source de démarrage configurée, vous devez la configurer. Vous pouvez voir les détails des procédures présentées dans le "[documentation](#)".
3. Après avoir configuré le disque de démarrage, vous pouvez cliquer sur **Personnaliser les paramètres de la machine virtuelle**.
4. Vous pouvez personnaliser la VM à partir des onglets de cette page. Par exemple, cliquez sur l'onglet **Disques** puis sur **Ajouter un disque** pour ajouter un autre disque à la machine virtuelle.
5. Cliquez sur Créer une machine virtuelle pour créer la machine virtuelle ; cela lance un pod correspondant en arrière-plan.



Lorsqu'une source de démarrage est configurée pour un modèle ou un système d'exploitation à partir d'une URL ou d'un registre, elle crée un PVC dans le openshift-virtualization-os-images projet et télécharge l'image invitée KVM sur le PVC. Vous devez vous assurer que les modèles PVC disposent de suffisamment d'espace provisionné pour accueillir l'image invitée KVM pour le système d'exploitation correspondant. Ces PVC sont ensuite clonés et attachés en tant que disque racine aux machines virtuelles lorsqu'elles sont créées à l'aide des modèles respectifs dans n'importe quel projet.



Project: openshift-virtualization-os-images

Catalog > Customize template parameters > Customize VirtualMachine

Customize and create VirtualMachine

Template: CentOS Stream 9 VM

Overview YAML Scheduling Environment Network interfaces Disks Scripts Metadata

Name

centos-stream9-pleased-hamster

Namespace

openshift-virtualization-os-images

Description

Not available

Operating system

CentOS Stream 9 VM

CPU | Memory

1 CPU | 2 GiB Memory

Machine type

pc-q35-rhel9.2.0

Boot mode

BIOS

Start in pause mode

☐

Workload profile

Server

Network interfaces (1)

Name	Network	Type
default	Pod networking	Masquerade

Disks (2)

Name	Drive	Size
rootdisk	Disk	30 GiB
cloudinitdisk	Disk	-

Hardware devices

GPU devices

Not available

Host devices

Not available

Headless mode

☐

Hostname

centos-stream9-pleased-hamster

☒ Start this VirtualMachine after creation

Create VirtualMachine Cancel

Démonstration vidéo

La vidéo suivante montre une démonstration de la création d'une machine virtuelle dans OpenShift Virtualization à l'aide du stockage iSCSI.

[Créer une machine virtuelle dans OpenShift Virtualization à l'aide du stockage en blocs](#)

Créer une machine virtuelle avec Red Hat OpenShift Virtualization en utilisant le stockage ONTAP par des méthodes alternatives

Outre la méthode de création d'une machine virtuelle à l'aide de l'interface utilisateur OpenShift Virtualization, il existe d'autres méthodes pour créer des machines virtuelles sur OpenShift Virtualization. Cette section présente un aperçu de certaines de ces méthodes.

Créer une machine virtuelle à partir d'un fichier YAML avec Red Hat OpenShift Virtualization

Il est également possible de créer une machine virtuelle en appliquant un fichier YAML avec la `oc apply` commande. Cette méthode est utile pour les utilisateurs qui préfèrent utiliser l'interface de ligne de commande ou qui souhaitent automatiser la création de machines virtuelles à l'aide de scripts. Pour créer une machine virtuelle avec cette méthode, vous pouvez créer un fichier YAML qui définit la machine virtuelle et ses

ressources associées, telles que les PVC pour les disques de la machine virtuelle. Vous pouvez ensuite appliquer le fichier YAML avec la `oc apply -f <yaml-file>` commande pour créer la machine virtuelle sur le cluster OpenShift.

Afficher un exemple

```
apiVersion: kubevirt.io/v1
kind: VirtualMachine
metadata:
  name: test-vms-vm
  namespace: test-5vms
  labels:
    app: test-vms-vm
    vm.kubevirt.io/template: centos-stream9-server-small
spec:
  dataVolumeTemplates:
    - apiVersion: cdi.kubevirt.io/v1beta1
      kind: DataVolume
      metadata:
        name: test-vms-vm-rootdisk
      spec:
        sourceRef:
          kind: DataSource
          name: centos-stream9
          namespace: openshift-virtualization-os-images
        storage:
          resources:
            requests:
              storage: 30Gi
  runStrategy: Always
  template:
    metadata:
      labels:
        kubevirt.io/domain: test-vms-vm
    spec:
      domain:
        cpu:
          cores: 1
          sockets: 1
          threads: 1
        devices:
          disks:
            - bootOrder: 1
              disk:
                bus: virtio
                name: rootdisk
            - disk:
                bus: virtio
                name: cloudinitdisk
          interfaces:
```

```

- masquerade: {}
  model: virtio
  name: default
resources:
  requests:
    memory: 2Gi
networks:
- name: default
  pod: {}
terminationGracePeriodSeconds: 180
volumes:
- dataVolume:
    name: test-vms-vm-rootdisk
    name: rootdisk
- cloudInitNoCloud:
    userData: |-
      #cloud-config
      user: admin
      password: <replace with your password>
      chpasswd: { expire: False }
    name: cloudinitdisk

```

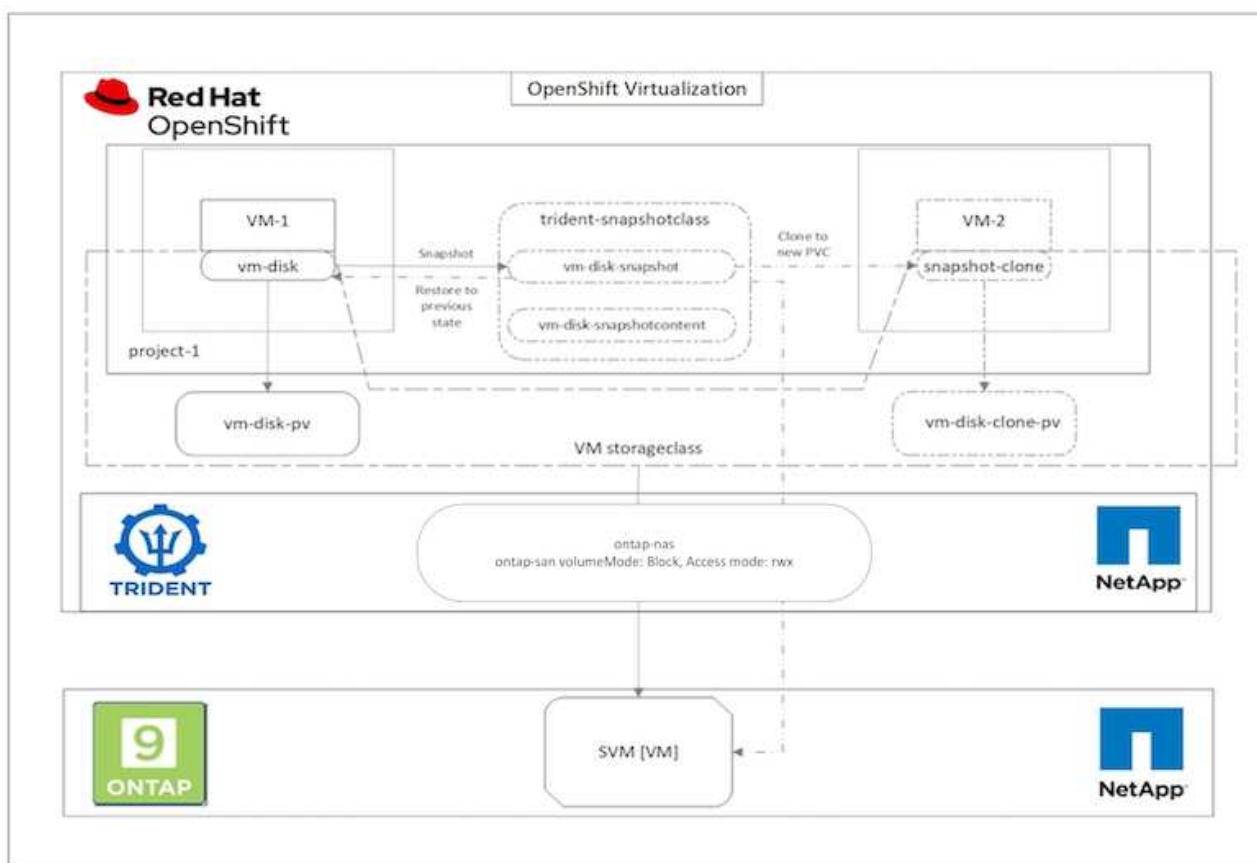
Créer une machine virtuelle à partir d'une copie instantanée avec Red Hat OpenShift Virtualization

Vous pouvez créer une machine virtuelle à partir d'un instantané avec OpenShift Virtualization. Cette procédure inclut la création d'une VolumeSnapshotClass, la prise d'un instantané de la demande de volume persistant (PVC) de la machine virtuelle, la restauration de l'instantané sur un nouveau PVC, et le déploiement d'une nouvelle machine virtuelle utilisant le PVC restauré comme disque racine.

Avec Trident et Red Hat OpenShift, les utilisateurs peuvent prendre un instantané d'un volume persistant sur les classes de stockage provisionnées par celui-ci. Grâce à cette fonctionnalité, les utilisateurs peuvent prendre une copie ponctuelle d'un volume et l'utiliser pour créer un nouveau volume ou restaurer le même volume à un état antérieur. Cela permet ou prend en charge une variété de 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 VolumeSnapshotContent doivent être définies.

- Un VolumeSnapshotContent est l'instantané réel pris à partir d'un volume du cluster. Il s'agit d'une ressource à l'échelle du cluster analogue à PersistentVolume pour le stockage.
- Un VolumeSnapshot est une demande de création d'un instantané d'un volume. Il est analogue à un PersistentVolumeClaim.
- VolumeSnapshotClass permet à l'administrateur de spécifier différents attributs pour un VolumeSnapshot. Il vous permet d'avoir des attributs différents pour différents instantanés pris à partir du même volume.



Afficher un exemple

Condition préalable

Vous devez disposer d'une machine virtuelle (VM) existante avec un PVC attaché, provisionné sur une classe de stockage Trident, et la VM doit être en cours d'exécution avant que vous puissiez prendre un instantané du PVC. Vous devez avoir une VolumeSnapshotClass créée, qui pourra ensuite être utilisée pour créer un VolumeSnapshot. Celle-ci aura été créée dans le cadre des prérequis pour l'installation de OpenShift Virtualization. Vous pouvez consulter la ["Étape 3 pour créer une configuration VolumeSnapshotClass"](#) pour plus de détails.

Pour créer un instantané d'une machine virtuelle, suivez les étapes suivantes :

1. Identifiez le PVC attaché à la machine virtuelle source, puis créez un instantané de ce PVC. Accédez à `Storage > VolumeSnapshots` et cliquez sur `Créer des VolumeSnapshots`.
2. Sélectionnez le PVC pour lequel vous souhaitez créer le snapshot, entrez le nom du snapshot ou acceptez la valeur par défaut, puis sélectionnez le VolumeSnapshotClass approprié. Cliquez ensuite sur `Créer`.
3. Cela crée l'instantané du PVC à ce moment précis.

Create VolumeSnapshot

[Edit YAML](#)

PersistentVolumeClaim *

PVC rhel8-short-frog-rootdisk-28dvv

Name *

rhel8-short-frog-rootdisk-28dvv-snapshot

Snapshot Class *

VSC trident-snapshot-class

Create

Cancel

Créer une nouvelle machine virtuelle à partir de l'instantané

1. Tout d'abord, restaurez l'instantané dans un nouveau PVC. Accédez à `Stockage > Instantanés de volume`, cliquez sur les points de suspension en regard de l'instantané que vous souhaitez restaurer, puis cliquez sur `Restaurer` en tant que nouveau PVC.
2. Saisissez les détails du nouveau PVC et cliquez sur `Restaurer`. Cela crée un nouveau PVC.

Restore as new PVC

When restore action for snapshot **rhel8-short-frog-rootdisk-28dvb-snapshot** is finished a new crash-consistent PVC copy will be created.

Name *

rhel8-short-frog-rootdisk-28dvb-snapshot-restore

Storage Class *



basic



Access Mode *

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

Size *

20

GiB



VolumeSnapshot details

Created at

 May 21, 12:46 am

Namespace

 default

Status

 Ready

API version

snapshot.storage.k8s.io/v1

Size

20 GiB

1. Ensuite, créez une nouvelle VM à partir de ce PVC. Accédez à Virtualisation > Machines virtuelles et cliquez sur Créer > Avec YAML.
2. Dans la section spec > template > spec > volumes, spécifiez le nouveau PVC créé à partir de Snapshot plutôt qu'à partir du disque conteneur. Fournissez tous les autres détails de la nouvelle machine virtuelle en fonction de vos besoins.

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

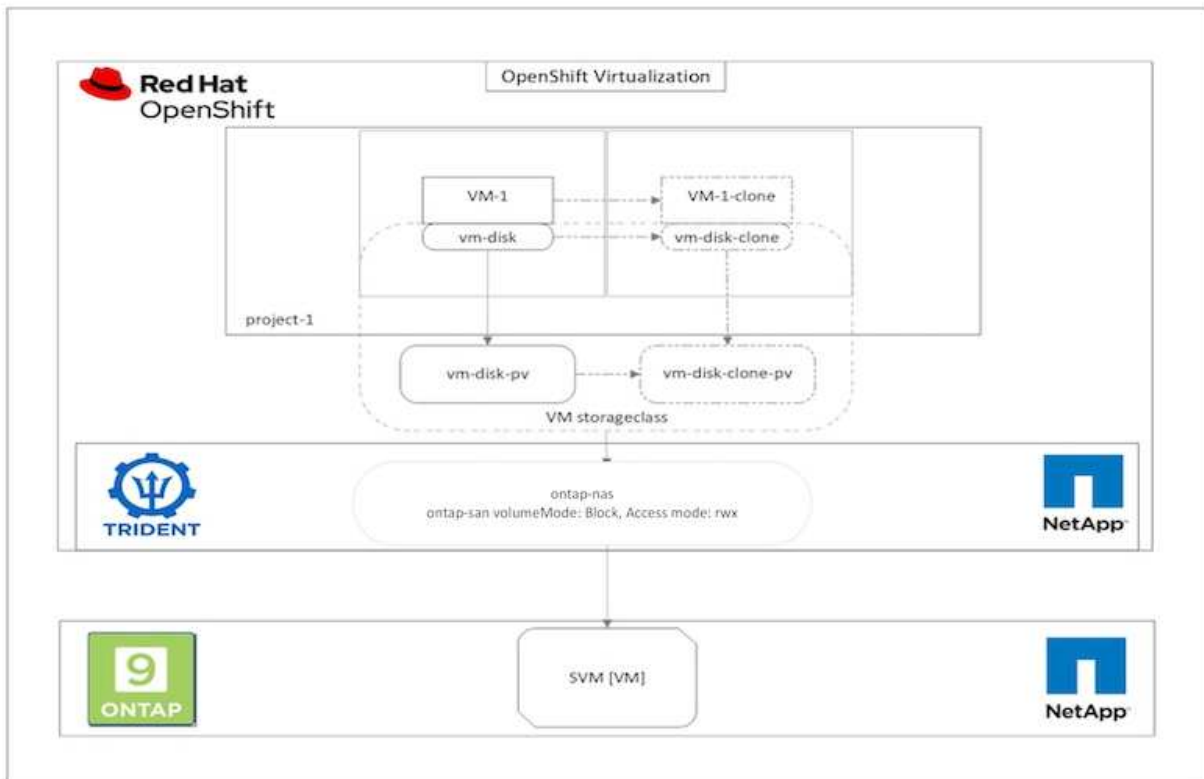
1. Cliquez sur Créer pour créer la nouvelle VM.
2. Une fois la machine virtuelle créée avec succès, accédez à la nouvelle machine virtuelle et vérifiez qu'elle a le même état que la machine virtuelle dont le PVC a été utilisé pour créer l'instantané au moment de la création de l'instantané.

Cloner une machine virtuelle dans OpenShift Virtualization à l'aide de Trident

Cette procédure inclut l'utilisation du clonage de volumes Trident CSI, vous permettant de créer une nouvelle VM en arrêtant la VM source ou en la maintenant en fonctionnement.

Clonage de machines virtuelles

Le clonage d'une machine virtuelle existante dans OpenShift est réalisé avec la prise en charge de la fonction de clonage Volume CSI de Trident. Le clonage de volume CSI permet de créer un nouveau PVC en utilisant un PVC existant comme source de données en dupliquant son PV. Une fois le nouveau PVC créé, il fonctionne comme une entité distincte et sans aucun lien ni dépendance avec le PVC source.



Il existe certaines restrictions à prendre en compte lors du clonage de volume CSI :

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 ne peut être effectué 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 machines virtuelles d'un cluster OpenShift peuvent être clonées de deux manières :

1. En arrêtant la VM source
2. En gardant la VM source en direct

En arrêtant la machine virtuelle source

Le clonage d'une machine virtuelle existante en arrêtant la machine virtuelle est une fonctionnalité native d'OpenShift qui est implémentée avec le support de Trident. Suivez les étapes suivantes pour cloner une machine virtuelle.

1. Accédez à Charges de travail > Virtualisation > Machines virtuelles et cliquez sur les points de suspension en regard de la machine virtuelle que vous souhaitez cloner.
2. Cliquez sur Cloner la machine virtuelle et fournissez les détails de la nouvelle machine virtuelle.

Clone Virtual Machine

Name *

rhel8-short-frog-clone

Description

Namespace *

default

☒ Start virtual machine on clone

Configuration

Operating System

Red Hat Enterprise Linux 8.0 or higher

Flavor

Small: 1 CPU | 2 GiB Memory

Workload Profile

server

NICs

default - virtio

Disks

cloudinitdisk - cloud-init disk

rootdisk - 20Gi - basic

 The VM rhel8-short-frog is still running. It will be powered off while cloning.

Cancel

Clone Virtual Machine

3. Cliquez sur Cloner la machine virtuelle ; cela arrête la machine virtuelle source et lance la création de la machine virtuelle clonée.
4. Une fois cette étape terminée, vous pouvez accéder au contenu de la machine virtuelle clonée et le

vérifier.

En maintenant la machine virtuelle source en activité

Une machine virtuelle existante peut également être clonée en clonant le PVC existant de la machine virtuelle source, puis en créant une nouvelle machine virtuelle à l'aide du PVC cloné. Cette méthode ne nécessite pas d'arrêter la machine virtuelle source. Suivez les étapes suivantes pour cloner une machine virtuelle sans l'arrêter.

1. Accédez à Stockage > PersistentVolumeClaims et cliquez sur les points de suspension en regard du PVC attaché à la machine virtuelle source.
2. Cliquez sur Cloner le PVC et fournissez les détails du nouveau PVC.

Clone

Name *

rhel8-short-frog-rootdisk-28dvv-clone

Access Mode *

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

Size *

20

GiB

PVC details

Namespace

 default

Requested capacity

20 GiB

Access mode

Shared Access (RWX)

Storage Class

 basic

Used capacity

2.2 GiB

Volume mode

Filesystem

Cancel

Clone

3. Cliquez ensuite sur Cloner. Cela crée un PVC pour la nouvelle VM.
4. Accédez à Charges de travail > Virtualisation > Machines virtuelles et cliquez sur Créer > Avec YAML.
5. Dans la section spec > template > spec > volumes, attachez le PVC cloné au lieu du disque conteneur. Fournissez tous les autres détails de la nouvelle machine virtuelle en fonction de vos besoins.

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

1. Cliquez sur Créer pour créer la nouvelle VM.
2. Une fois la machine virtuelle créée avec succès, accédez à la nouvelle machine virtuelle et vérifiez qu'elle est un clone de la machine virtuelle source.

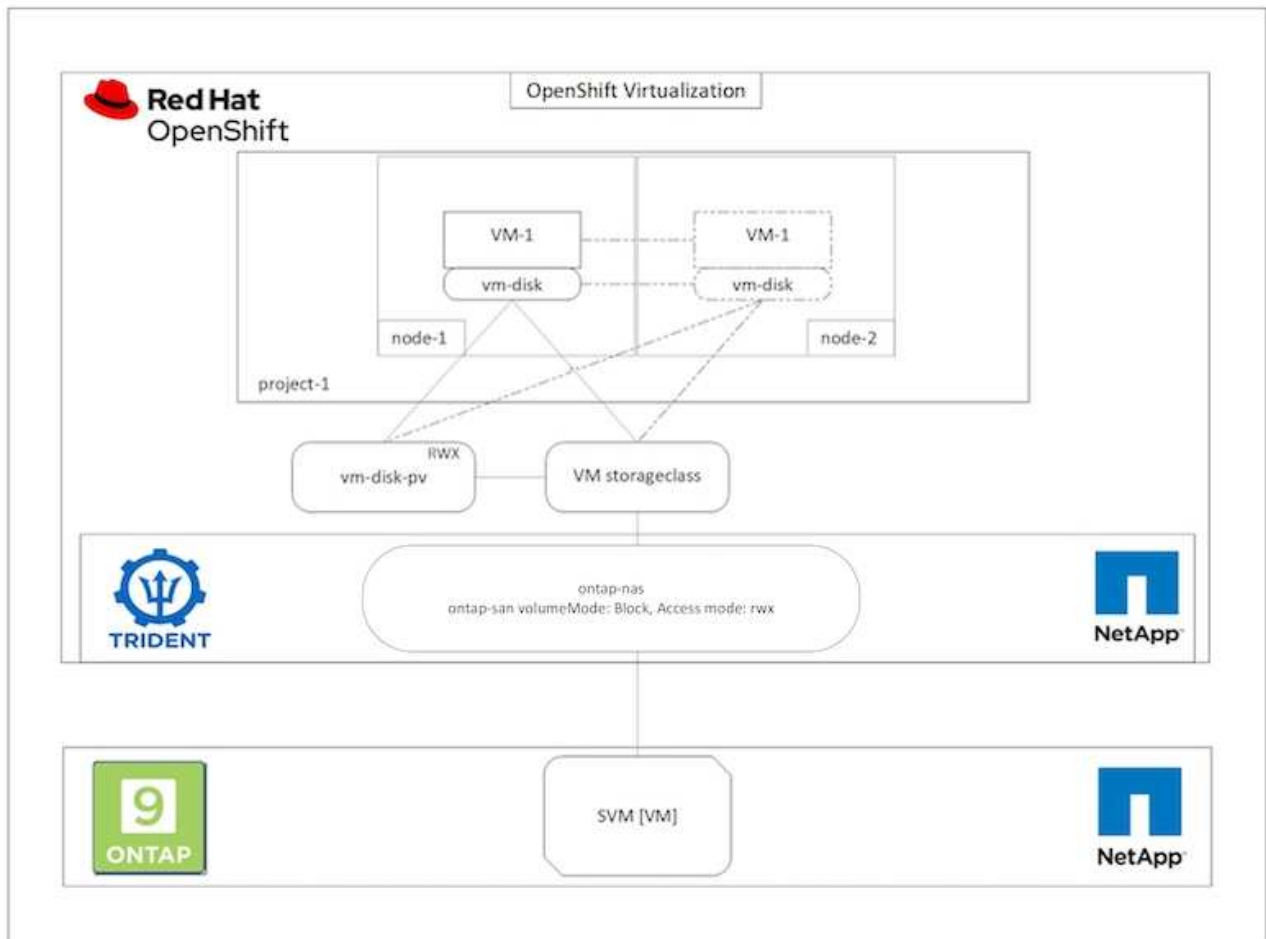
Migrer une machine virtuelle entre deux nœuds dans un cluster Red Hat OpenShift

Migrez une machine virtuelle dans OpenShift Virtualization entre deux nœuds du cluster sans temps d'arrêt. Cette procédure comprend la confirmation que les disques utilisent des classes de stockage compatibles RWX, le lancement de la migration et la surveillance de la progression.

Migration en direct de machines virtuelles

La migration en direct est un processus de migration d'une instance de machine virtuelle d'un nœud à un autre dans un cluster OpenShift sans temps d'arrêt. Pour que la migration en direct fonctionne dans un cluster OpenShift, les machines virtuelles doivent être liées à des PVC avec un mode d'accès ReadWriteMany partagé. Les backends Trident configurés à l'aide des pilotes ontap-nas prennent en charge le mode d'accès RWX pour les protocoles FileSystem nfs et smb. Se référer à la documentation ["ici"](#) . Les backends Trident configurés à l'aide des pilotes ontap-san prennent en charge le mode d'accès RWX pour le mode volume de bloc pour les protocoles iSCSI et NVMe/TCP. Se référer à la documentation ["ici"](#) .

Par conséquent, pour que la migration en direct réussisse, les machines virtuelles doivent être provisionnées avec des disques (disques de démarrage et disques enfichables à chaud supplémentaires) avec des PVC utilisant les classes de stockage ontap-nas ou ontap-san (volumeMode : Block). Lorsque les PVC sont créés, Trident crée des volumes ONTAP dans un SVM compatible NFS ou iSCSI.

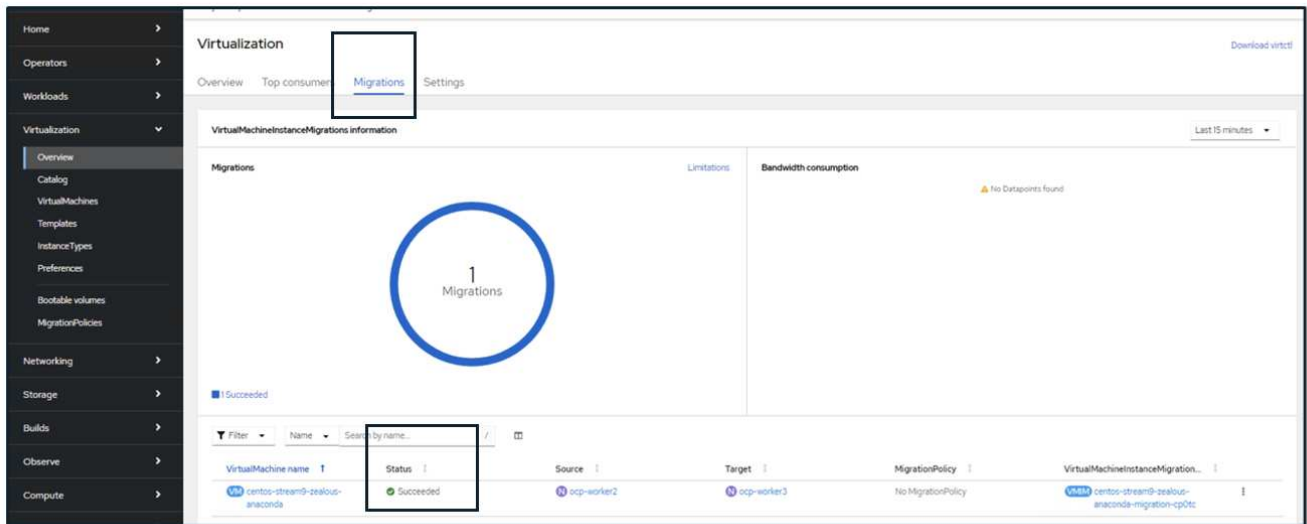
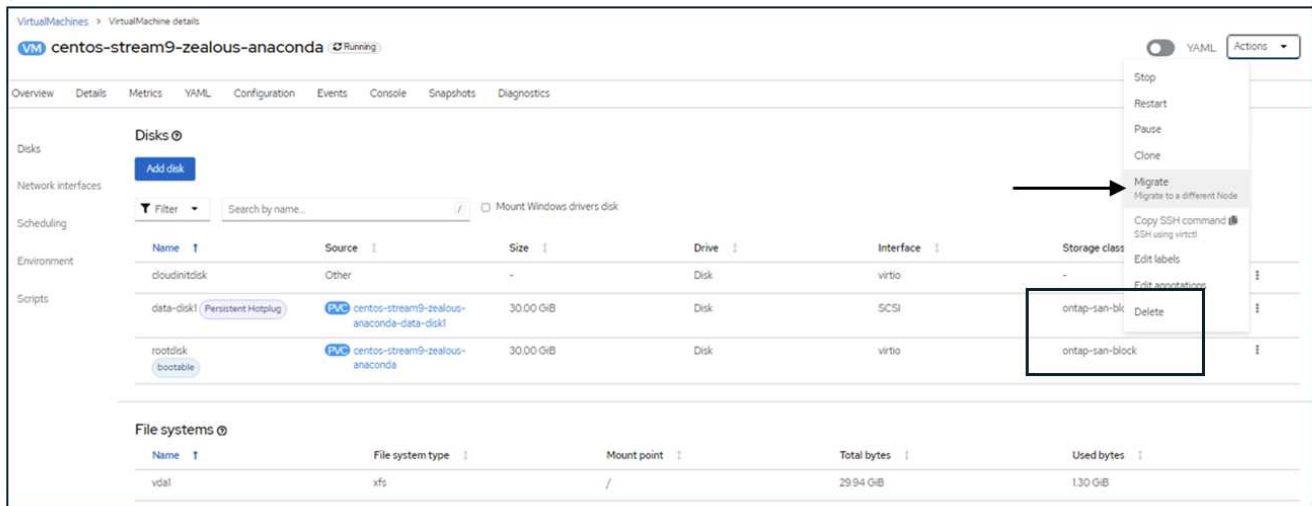


Pour effectuer une migration en direct d'une machine virtuelle qui a été créée précédemment et qui est en état d'exécution, procédez comme suit :

1. Sélectionnez la machine virtuelle que vous souhaitez migrer en direct.
2. Cliquez sur l'onglet **Configuration**.
3. Assurez-vous que tous les disques de la machine virtuelle sont créés à l'aide de classes de stockage pouvant prendre en charge le mode d'accès RWX.
4. Cliquez sur **Actions** dans le coin droit, puis sélectionnez **Migrer**.
5. Pour suivre la progression de la migration, accédez à Virtualisation > Présentation dans le menu de gauche, puis cliquez sur l'onglet **Migrations**. La migration de la machine virtuelle passera de **En attente à Planification** puis à **Réussie**



Une instance de machine virtuelle dans un cluster OpenShift migre automatiquement vers un autre nœud lorsque le nœud d'origine est placé en mode maintenance si evictionStrategy est défini sur LiveMigrate.



Migration du stockage des machines virtuelles entre classes de stockage dans OpenShift Virtualization

Dans OpenShift Virtualization, vous pouvez migrer les disques d'une machine virtuelle (VM) vers une autre classe de stockage directement via la console web, sans supprimer la VM. Ce processus vous permet de déplacer la totalité du stockage de la VM ou des volumes spécifiques.

Les étapes suivantes décrivent comment migrer la classe de stockage d'une machine virtuelle :

1. Assurez-vous que la classe de stockage cible est correctement configurée et disponible dans votre OpenShift cluster. Cela inclut la mise en place et l'accessibilité du système de stockage principal nécessaire (par exemple, NetApp ONTAP).

Afficher un exemple

```
[root@00-50-56-b5-2b-9b ~]# oc get tbe -n trident
NAME          BACKEND          BACKEND UUID
tbe-k2vqg     tbc-san          3e9dc8ae-3b29-44c8-99a9-affdeaa6b4cc
tbe-mtp5n     tbc-nas-eco      a59e8053-bb00-4410-9b12-1dc4d283928b
tbe-nwqvq     tbc-nas-50       8b49f02f-3bc3-4720-b60f-bf90c3600008
tbe-wtp9h     tbc_san_eco      61745526-04cd-4acb-8f1b-daa9aa79f05a
tbe-zfp46     tbc-nas          259a1c2a-0a84-488d-ab62-ccea45499e09
```

```
[root@00-50-56-b5-2b-9b ~]# oc get sc
NAME          PROVISIONER          RECLAIMPOLICY  VOLUMEBINDINGMODE  ALLOWVOLUMEEXPANSION  AGE
sc-nas (default)  csi.trident.netapp.io  Delete         Immediate          true                  35d
sc-nas-50        csi.trident.netapp.io  Delete         Immediate          true                  22d
sc-nas-economy   csi.trident.netapp.io  Delete         Immediate          true                  20d
sc-san           csi.trident.netapp.io  Delete         Immediate          false                 21d
```

2. Dans la console web OpenShift Container Platform, accédez à la perspective Virtualisation et sélectionnez la machine virtuelle que vous souhaitez migrer.

VirtualMachines

VirtualMachine details

VM

rhel9-sc-migration

Running

■

↺

⏸

▶

Actions

Overview

Metrics

YAML

Configuration

Events

Console

Snapshots

Diagnostics

Details

Name

rhel9-sc-migration

Status

Running

Created

May 14, 2020, 1:31 PM (17 minutes ago)

Operating system

Red Hat Enterprise Linux 9.7 (Plow)

CPU | Memory

1 CPU | 2 GiB Memory

Time zone

EDT

Template

rhel9-server-small

Hostname

rhel9-sc-migration

VNC console

Open web console

Alerts (0)

General

Namespace

test-sc-migration

Node

00-50-56-b5-8d-fa

VirtualMachineInstance

rhel9-sc-migration

Pod

virt-launcher-rhel9-sc-migration-Sjkdw

Owner

No owner

Snapshots (0)

Take snapshot

VirtualMachines

VirtualMachine details

VM

rhel9-sc-migration

Running

YAML

■

↺

Overview

Metrics

YAML

Configuration

Events

Console

Snapshots

Diagnostics

Details

Storage

Network

Scheduling

SSH

Initial run

Metadata

Disks

Add disk

Filter

Search by name...

/

Mount Windows drivers disk

Name	Source	Size	Drive	Interface	Storage class
cloudinitdisk	Other	-	Disk	virtio	-
disk-rhel9-nas	PVC cv-rhel9-migration-disk-emerald-wren-75-d88usc	53 GiB	Disk	virtio	sc-nas
rootdisk	PVC rhel9-migration	31.8 GiB	Disk	virtio	sc-nas

```
[root@00-50-56-b5-2b-9b ~]# oc get vm,pvc -n test-sc-migration
NAME                                AGE    STATUS    READY
virtualmachine.kubevirt.io/rhel9-sc-migration  4m19s  Running  True

NAME                                STATUS    VOLUME                                CAPACITY    ACCESS MODES    STORAGECLASS    VOLUMEATTRIBUTESCLASS    AGE
persistentvolumeclaim/dv-rhel9-migration-disk-emerald-wren-75-d88usc  Bound    pvc-bb54342f-c01f-453e-864d-e8f72e710a2f  34144990004    RWO            sc-nas          <unset>                4m19s
persistentvolumeclaim/rhel9-migration
[root@00-50-56-b5-2b-9b ~]#
```

- 57

Afficher un exemple

VirtualMachines > VirtualMachine details

VM **rhel9-sc-migration** Running

Overview

Metrics

YAML

Configuration

Events

Console

Snapshots

Diagnostics

Details

Name	rhel9-sc-migration
Status	Running
Created	May 14, 2026, 1:31 PM (19 minutes ago)
Operating system	Red Hat Enterprise Linux 9.7 (Plow)
CPU Memory	1 CPU 2 GiB Memory
Time zone	EDT
Template	rhel9-server-small
Hostname	rhel9-sc-migration

VNC console
[Open web console](#)

Alerts (0)

General

Compute

Namespace

Node

VirtualMachineInstance

Pod

Owner

Migrate VirtualMachine to a different Node

Migrate VirtualMachine storage to a different StorageClass

rhel9-sc-migration

virt-launcher-rhel9-sc-migration-5jkdw

No owner

Snapshots (0)

[Take snapshot](#)

Migrate VirtualMachine storage

Migrate VirtualMachine storage to a different StorageClass.

1 Migration details

2 Destination StorageClass

3 Review

Migration details

Select the storage to migrate for **rhel9-sc-migration**

☒ The entire VirtualMachine

☐ Selected volumes

Back

Next

Cancel

Migrate VirtualMachine storage

Migrate VirtualMachine storage to a different StorageClass.

1 Migration details

2 Destination StorageClass

3 Review

Destination StorageClass

Select the destination storage for the VirtualMachine storage migration.

sc-nas-economy

Back

Next

Cancel

Migrate VirtualMachine storage

Migrate VirtualMachine storage to a different StorageClass.

1 Migration details

2 Destination StorageClass

3 Review

Review

Verify the details and click **Migrate VirtualMachine storage** to start the migration

Migration type	VirtualMachine storage
Destination StorageClass	sc-nas-economy
Migrating 1 out of 3	disk-rhel9-nas

Back

Migrate VirtualMachine storage

Cancel

Migrate VirtualMachine storage

Migrate VirtualMachine storage to a different StorageClass.

In progress

Started on

🕒 May 14, 2026, 1:54 PM

Status

Scheduling

Stop

Cancel

[View report](#)

4. Une fois la migration terminée, vous pouvez vérifier que les disques de la machine virtuelle utilisent désormais la nouvelle classe de stockage en consultant les détails de la machine virtuelle dans la console Web ou en utilisant l'interface de ligne de commande OpenShift.

Afficher un exemple

```
[root@00-50-56-b5-2b-9b ~]# oc get vm,pvc -n test-sc-migration
NAME                                AGE    STATUS    READY
virtualmachine.kubevirt.io/rhel9-sc-migration    24m    Migrating True

NAME                                STATUS    VOLUME                                CAPACITY    ACCESS MODES    STORAGECLASS    VOLUMEATTRIBUTESCLASS    AGE
persistentvolumeclaim/dv-rhel9-migration-disk-emerald-wren-75-d88usc    Bound    pvc-bb54342f-c01f-453e-864d-e8f72e710a2f    53Gi    RWX    sc-nas    <unset>    24m
persistentvolumeclaim/dv-rhel9-migration-disk-emerald-wren-75-d88usc-mig-839zfb    Bound    pvc-75a885f0-824e-407c-ab29-8cc546dbdb3b    60322815673    RWX    sc-nas-economy    <unset>    2m17s
persistentvolumeclaim/rhel9-migration    Bound    pvc-46a34903-942f-4ddc-8e6d-974008f05d09    34144990004    RWX    sc-nas    <unset>    24m
[root@00-50-56-b5-2b-9b ~]#
```

```
[root@00-50-56-b5-2b-9b ~]# oc get all -n test-sc-migration
Warning: apps.openshift.io/v1 DeploymentConfig is deprecated in v4.14+, unavailable in v4.10000+
Warning: kubevirt.io/v1 VirtualMachineInstancePresets is now deprecated and will be removed in v2.
NAME                                READY    STATUS    RESTARTS    AGE
pod/virt-launcher-rhel9-sc-migration-4hd7    1/1    Running    0    2m50s
pod/virt-launcher-rhel9-sc-migration-5jkdw    0/1    Completed    0    25m

NAME                                TYPE    CLUSTER-IP    EXTERNAL-IP    PORT(S)    AGE
service/headless                    ClusterIP    None    <none>    5434/TCP    25m

NAME                                PHASE    PROGRESS    RESTARTS    AGE
datavolume.cdi.kubevirt.io/dv-rhel9-migration-disk-emerald-wren-75-d88usc    Succeeded    100.0%    0    25m
datavolume.cdi.kubevirt.io/dv-rhel9-migration-disk-emerald-wren-75-d88usc-mig-839zfb    Succeeded    100.0%    0    3m7s
datavolume.cdi.kubevirt.io/rhel9-migration    Succeeded    100.0%    0    25m

NAME                                PHASE    VMI
virtualmachineinstancemigration.kubevirt.io/kubevirt-workload-update-lbfj9    Succeeded    rhel9-sc-migration

NAME                                AGE    PHASE    IP    NODENAME    READY
virtualmachineinstance.kubevirt.io/rhel9-sc-migration    25m    Running    10.131.0.22    00-50-56-b5-84-74    True

NAME                                AGE    STATUS    READY
virtualmachine.kubevirt.io/rhel9-sc-migration    25m    Running    True
[root@00-50-56-b5-2b-9b ~]#
[root@00-50-56-b5-2b-9b ~]#
[root@00-50-56-b5-2b-9b ~]# oc get vm,pvc -n test-sc-migration
NAME                                AGE    STATUS    READY
virtualmachine.kubevirt.io/rhel9-sc-migration    26m    Running    True

NAME                                STATUS    VOLUME                                CAPACITY    ACCESS MODES    STORAGECLASS    VOLUMEATTRIBUTESCLASS    AGE
persistentvolumeclaim/dv-rhel9-migration-disk-emerald-wren-75-d88usc    Bound    pvc-bb54342f-c01f-453e-864d-e8f72e710a2f    53Gi    RWX    sc-nas    <unset>    26m
persistentvolumeclaim/dv-rhel9-migration-disk-emerald-wren-75-d88usc-mig-839zfb    Bound    pvc-75a885f0-824e-407c-ab29-8cc546dbdb3b    60322815673    RWX    sc-nas-economy    <unset>    3m28s
persistentvolumeclaim/rhel9-migration    Bound    pvc-46a34903-942f-4ddc-8e6d-974008f05d09    34144990004    RWX    sc-nas    <unset>    26m
[root@00-50-56-b5-2b-9b ~]#
```

```
[root@00-50-56-b5-2b-9b ~]# oc describe pv pvc-46a34903-942f-4ddc-8e6d-974008f05d09 -n test-sc-migration
Name:          pvc-46a34903-942f-4ddc-8e6d-974008f05d09
Labels:        <none>
Annotations:   pv.kubernetes.io/provisioned-by: csi.trident.netapp.io
               volume.kubernetes.io/provisioner-deletion-secret-name:
               volume.kubernetes.io/provisioner-deletion-secret-namespace:
Finalizers:    [external-provisioner.volume.kubernetes.io/finalizer kubernetes.io/pv-protection external-attacher/csi-trident-netapp-io]
StorageClass:  sc-nas
Status:        Bound
Claim:         test-sc-migration/rhel9-migration
Reclaim Policy: Delete
Access Modes:  RWX
VolumeMode:    Filesystem
Capacity:      34144990004
Node Affinity: <none>
Message:
Source:
  Type:        CSI (a Container Storage Interface (CSI) volume source)
  Driver:      csi.trident.netapp.io
  FSType:
  VolumeHandle: pvc-46a34903-942f-4ddc-8e6d-974008f05d09
  ReadOnly:    false
  VolumeAttributes:
    backendUUID=259a1c2a-0a84-488d-ab62-ccea45499e09
    internalName=oc_automation_pvc_46a34903_942f_4ddc_8e6d_974008f05d09
    name=pvc-46a34903-942f-4ddc-8e6d-974008f05d09
    protocol=file
    storage.kubernetes.io/csiProvisionerIdentity=1778505459781-8296-csi.trident.netapp.io
Events:        <none>
[root@00-50-56-b5-2b-9b ~]#
```

```

NSOL-NetApp-A70-T19U01:> volume show -vserver oc-automation-svm -volume oc_automation_pvc_46a34903_942f_4ddc_8e6d_974008f05d09

Vserver Name: oc-automation-svm
Volume Name: oc_automation_pvc_46a34903_942f_4ddc_8e6d_974008f05d09
Aggregate Name: NSOL_NetApp_A70_T19U01a_NVME_SSD_1
List of Aggregates for FlexGroup Constituents: NSOL_NetApp_A70_T19U01a_
NVME_SSD_1
Encryption Type: none
List of Nodes Hosting the Volume: NSOL-NetApp-A70-T19U01a
Volume Size: 31.80GB
Volume Data Set ID: 1716
Volume Master Data Set ID: 2152408880
Volume State: online
Volume Style: flex
Extended Volume Style: flexvol
FlexCache Endpoint Type: none
Is Cluster-Mode Volume: true
Is Constituent Volume: false
Number of Constituent Volumes: -
Export Policy: default
User ID: 0
Group ID: 0
Security Style: unix
UNIX Permissions: ---rwxrwxrwx
Junction Path: /oc_automation_pvc_46a34903_942f_4ddc_8e6d_974008f05d09
Junction Path Source: RW_volume
Junction Active: true
Junction Parent Volume: oc_automation-svm_root
Comment:
Available Size: 30.03GB
Filesystem Size: 31.80GB
Total User-Visible Size: 31.80GB
Used Size: 1.76GB
Used Percentage: 5%
Volume Nearly Full Threshold Percent: 95%
Volume Full Threshold Percent: 98%
Maximum Autosize: 38.16GB
Minimum Autosize: 31.80GB

```

```

[root@00-50-56-b5-2b-9b ~]# oc get all -n test-sc-migration
Warning: apps.openshift.io/v1 DeploymentConfig is deprecated in v4.14+, unavailable in v4.10000+
Warning: kubevirt.io/v1 VirtualMachineInstancePresets is now deprecated and will be removed in v2.
NAME                                READY    STATUS    RESTARTS   AGE
pod/virt-launcher-rhel9-sc-migration-44dc7    1/1      Running    0           2m50s
pod/virt-launcher-rhel9-sc-migration-5jkdw    0/1      Completed  0           25m

NAME                                TYPE          CLUSTER-IP    EXTERNAL-IP    PORT(S)    AGE
service/headless                    ClusterIP     None          <none>         5434/TCP   25m

NAME                                PHASE    PROGRESS    RESTARTS   AGE
datavolume.cdi.kubevirt.io/dv-rhel9-migration-disk-emerald-wren-75-d88usc    Succeeded  100.0%      0           25m
datavolume.cdi.kubevirt.io/dv-rhel9-migration-disk-emerald-wren-75-d88usc-mig-839zfb    Succeeded  100.0%      0           3m7s
datavolume.cdi.kubevirt.io/rhel9-migration                                Succeeded  100.0%      0           25m

NAME                                PHASE    VMI
virtualmachineinstancemigration.kubevirt.io/kubevirt-workload-update-lbfj9    Succeeded  rhel9-sc-migration

NAME                                AGE    PHASE    IP            NODENAME    READY
virtualmachineinstance.kubevirt.io/rhel9-sc-migration    25m    Running  10.131.0.22   00-50-56-b5-84-74    True

NAME                                AGE    STATUS    READY
virtualmachine.kubevirt.io/rhel9-sc-migration    25m    Running    True
[root@00-50-56-b5-2b-9b ~]#
[root@00-50-56-b5-2b-9b ~]#
[root@00-50-56-b5-2b-9b ~]# oc get vm,pvc -n test-sc-migration
NAME                                AGE    STATUS    READY
virtualmachine.kubevirt.io/rhel9-sc-migration    26m    Running    True

NAME                                STATUS    VOLUME    CAPACITY    ACCESS MODES    STORAGECLASS    VOLUMEATTRIBUTESCLASS    AGE
persistentvolumeclaim/dv-rhel9-migration-disk-emerald-wren-75-d88usc    Bound    pvc-bb5d342f-c01f-453e-86ad-abf72e710a2f    536i    Rwx    sc-nas    <unset>    26m
persistentvolumeclaim/dv-rhel9-migration-disk-emerald-wren-75-d88usc-mig-839zfb    Bound    pvc-75a885f0-824e-407c-ab29-8cc546dbdb3b    60322815673    Rwx    sc-nas-economy    <unset>    3m28s
persistentvolumeclaim/rhel9-migration                                Bound    pvc-46a34903-942f-4ddc-8e6d-974008f05d09    34144990004    Rwx    sc-nas    <unset>    26m
[root@00-50-56-b5-2b-9b ~]#

```

```

[root@00-50-56-b5-2b-9b ~]# oc describe pv -n test-sc-migration pvc-75a885f0-824e-407c-ab29-8cc546dbdb3b
Name:
Labels:
Annotations:
pv.kubernetes.io/provisioned-by: csi.trident.netapp.io
volume.kubernetes.io/provisioner-deletion-secret-name:
volume.kubernetes.io/provisioner-deletion-namespace:
Finalizers: [external-provisioner.volume.kubernetes.io/finalizer kubernetes.io/pv-protection external-attacher/csi-trident-netapp-io]
StorageClass: sc-nas-economy
Status: Bound
Claim: test-sc-migration/dv-rhel9-migration-disk-emerald-wren-75-d88usc-mig-839zfb
Reclaim Policy: Delete
Access Modes: Rwx
VolumeMode: Filesystem
Capacity: 60322815673
Node Affinity: <none>
Message:
Source:
Type: CSI (a Container Storage Interface (CSI) volume source)
Driver: csi.trident.netapp.io
FSType:
VolumeHandle: pvc-75a885f0-824e-407c-ab29-8cc546dbdb3b
ReadOnly: false
VolumeAttributes:
  backendUUID: a59e8053-bb00-4410-9b12-1dc4d283928b
  internalID: svm/oc-automation-svm/flexvol/trident_qtree_pool_oc_automation_OGBGMIAOLG/qtree/oc_automation_pvc_75a885f0_824e_407c_ab29_8cc546...
  internalName: oc_automation_pvc_75a885f0_824e_407c_ab29_8cc546dbdb3b
  name: pvc-75a885f0-824e-407c-ab29-8cc546dbdb3b
  protocol: file
  storage.kubernetes.io/csiProvisionerIdentity: 1778505459781-8296-csi.trident.netapp.io
Events:
<none>

```

Qtrees

+ Add

🔍 ⬇️ 🗃️ ☰

☐	Name	Storage VM	Volume name	Mount path
	oc_automation_pvc_75a885f0_824e_407c_ab29_8ccf			
▼	oc_automation_pvc_75a885f0_824e_407c_ab29_8cc546dbdb3b	oc-automation-svm	trident_qtree_pool_oc_automation_OGBGMIAOLG	/trident_qtree_pool_oc_automation_OGBGMIAOLG/oc_...

Informations sur le copyright

Copyright © 2026 NetApp, Inc. Tous droits réservés. Imprimé aux États-Unis. Aucune partie de ce document protégé par copyright ne peut être reproduite sous quelque forme que ce soit ou selon quelque méthode que ce soit (graphique, électronique ou mécanique, notamment par photocopie, enregistrement ou stockage dans un système de récupération électronique) sans l'autorisation écrite préalable du détenteur du droit de copyright.

Les logiciels dérivés des éléments NetApp protégés par copyright sont soumis à la licence et à l'avis de non-responsabilité suivants :

CE LOGICIEL EST FOURNI PAR NETAPP « EN L'ÉTAT » ET SANS GARANTIES EXPRESSES OU TACITES, Y COMPRIS LES GARANTIES TACITES DE QUALITÉ MARCHANDE ET D'ADÉQUATION À UN USAGE PARTICULIER, QUI SONT EXCLUES PAR LES PRÉSENTES. EN AUCUN CAS NETAPP NE SERA TENU POUR RESPONSABLE DE DOMMAGES DIRECTS, INDIRECTS, ACCESSOIRES, PARTICULIERS OU EXEMPLAIRES (Y COMPRIS L'ACHAT DE BIENS ET DE SERVICES DE SUBSTITUTION, LA PERTE DE JOUISSANCE, DE DONNÉES OU DE PROFITS, OU L'INTERRUPTION D'ACTIVITÉ), QUELLES QU'EN SOIENT LA CAUSE ET LA DOCTRINE DE RESPONSABILITÉ, QU'IL S'AGISSE DE RESPONSABILITÉ CONTRACTUELLE, STRICTE OU DÉLICTELLE (Y COMPRIS LA NÉGLIGENCE OU AUTRE) DÉCOULANT DE L'UTILISATION DE CE LOGICIEL, MÊME SI LA SOCIÉTÉ A ÉTÉ INFORMÉE DE LA POSSIBILITÉ DE TELS DOMMAGES.

NetApp se réserve le droit de modifier les produits décrits dans le présent document à tout moment et sans préavis. NetApp décline toute responsabilité découlant de l'utilisation des produits décrits dans le présent document, sauf accord explicite écrit de NetApp. L'utilisation ou l'achat de ce produit ne concède pas de licence dans le cadre de droits de brevet, de droits de marque commerciale ou de tout autre droit de propriété intellectuelle de NetApp.

Le produit décrit dans ce manuel peut être protégé par un ou plusieurs brevets américains, étrangers ou par une demande en attente.

LÉGENDE DE RESTRICTION DES DROITS : L'utilisation, la duplication ou la divulgation par le gouvernement sont sujettes aux restrictions énoncées dans le sous-paragraphe (b)(3) de la clause Rights in Technical Data-Noncommercial Items du DFARS 252.227-7013 (février 2014) et du FAR 52.227-19 (décembre 2007).

Les données contenues dans les présentes se rapportent à un produit et/ou service commercial (tel que défini par la clause FAR 2.101). Il s'agit de données propriétaires de NetApp, Inc. Toutes les données techniques et tous les logiciels fournis par NetApp en vertu du présent Accord sont à caractère commercial et ont été exclusivement développés à l'aide de fonds privés. Le gouvernement des États-Unis dispose d'une licence limitée irrévocable, non exclusive, non cessible, non transférable et mondiale. Cette licence lui permet d'utiliser uniquement les données relatives au contrat du gouvernement des États-Unis d'après lequel les données lui ont été fournies ou celles qui sont nécessaires à son exécution. Sauf dispositions contraires énoncées dans les présentes, l'utilisation, la divulgation, la reproduction, la modification, l'exécution, l'affichage des données sont interdits sans avoir obtenu le consentement écrit préalable de NetApp, Inc. Les droits de licences du Département de la Défense du gouvernement des États-Unis se limitent aux droits identifiés par la clause 252.227-7015(b) du DFARS (février 2014).

Informations sur les marques commerciales

NETAPP, le logo NETAPP et les marques citées sur le site <http://www.netapp.com/TM> sont des marques déposées ou des marques commerciales de NetApp, Inc. Les autres noms de marques et de produits sont des marques commerciales de leurs propriétaires respectifs.