



儲存和虛擬化入門

NetApp virtualization solutions

NetApp
July 21, 2026

This PDF was generated from <https://docs.netapp.com/zh-tw/netapp-solutions-virtualization/openshift/osv-storage-getting-started.html> on July 21, 2026. Always check docs.netapp.com for the latest.

目錄

儲存和虛擬化入門	1
了解NetApp Trident與 Red Hat OpenShift 的集成	1
NetApp Trident	1
Trident Protect	1
NetApp ONTAP 硬體型號與支援的通訊協定	2
用於 ONTAP 儲存的 Trident 驅動程式	2
在 Red Hat OpenShift 叢集上安裝 Trident 並建立儲存物件	3
步驟 1：安裝 Trident	4
步驟 2：為您的環境準備儲存後端和 StorageClass 設定檔	8
步驟 3：建立 VolumeSnapshotClass 設定檔	28
步驟 4：將組態檔套用至叢集	28
步驟 5：設定預設的 Trident 儲存和快照類別	30
使用ONTAP部署 Red Hat OpenShift Virtualization 的需求	31
先決條件	31
NetApp Trident CSI、儲存設定檔、OpenShift 虛擬化磁碟區存取模式	32
使用 Red Hat OpenShift Virtualization 與 ONTAP 儲存建立虛擬機器	36
建立VM	37
影片示範	40
使用其他方法，透過 ONTAP 儲存以 Red Hat OpenShift Virtualization 建立 VM	40
使用 Red Hat OpenShift Virtualization 從 YAML 檔案建立虛擬機	40
使用 Red Hat OpenShift Virtualization 從快照副本建立 VM	42
使用 Trident 在 OpenShift Virtualization 中複製 VM	46
在 Red Hat OpenShift 叢集中的兩個節點之間遷移虛擬機	50
虛擬機器即時遷移	50
OpenShift 虛擬化中虛擬機器在不同儲存類別之間的儲存遷移	51

儲存和虛擬化入門

了解NetApp Trident與 Red Hat OpenShift 的集成

儲存對於容器化應用程式至關重要，它能確保資料在動態且往往短暫的環境中持續可用並且高效管理。現代容器儲存解決方案旨在與應用程式無縫擴展，可根據應用程式需求靈活動態分配儲存資源，從而支援成長和多樣化的工作負載。由於 OpenShift 虛擬化技術使虛擬機器能夠以容器的形式運行，因此它們與 OpenShift 叢集中的其他應用程式使用相同的儲存空間。

NetApp ONTAP 是一款強大且高度可擴展的資料管理軟體，是 NetApp 企業級儲存解決方案的核心。它提供統一的儲存架構，可與各種儲存環境無縫整合，包括本地部署、雲端和混合部署。ONTAP 支援多種通訊協定，例如 NFS、CIFS/SMB、iSCSI、FC 和基於 TCP 的 NVMe，從而能夠與不同的應用程式和工作負載無縫整合。它支援將 SAN、NAS 和物件儲存整合到單一平台，從而簡化管理並降低成本。NetApp ONTAP 既可在本地部署使用，也可在雲端以託管服務形式提供。

NetApp Trident

Trident 是由 NetApp 開發的開源儲存協調器，旨在簡化 ONTAP 儲存於容器化環境中的使用。它支援為容器化應用程式動態佈建和管理持續性儲存。

Trident 的主要功能

CSI 相容性 確保與最新的 Kubernetes 儲存功能和標準相容。聲明式設定 允許使用者在 YAML 檔案中定義儲存需求，Trident 將自動管理這些需求。驅動程式 Trident 支援 ONTAP 支援的 NFS、CIFS/SMB、iSCSI、FC 和 NVMe/TCP 協定的驅動程式。混合雲和多雲支援 Trident 支援 ONTAP 本地儲存以及超大規模雲端服務商（例如 AWS 中的 FSx for NetApp ONTAP、Azure 中的 Azure NetApp Files 和 Google Cloud 中的 Google Cloud NetApp Volumes）中的託管儲存服務的驅動程式。進階資料管理 利用 ONTAP 的快照和複製功能，實現高效的資料保護並快速佈建測試和開發環境。

有關 Trident 的完整詳細資訊，請參閱 NetApp Trident ["文件在此處"](#)

Trident Protect

Trident Protect 是 Trident 的進階功能，可為容器化環境提供增強的資料保護和安全性。其設計旨在確保關鍵資料免受潛在威脅和資料遺失情境的影響。所有 Trident 使用者均可免費使用此功能，並可在任何已安裝 Trident 的 OpenShift 叢集中輕鬆啟用。

Trident Protect 的主要功能

自動備份 Trident Protect 可對容器應用程式和虛擬機器進行自動、原則驅動的備份，確保定期備份而無需人工介入。

快照管理 它利用 ONTAP 的快照功能，為容器應用程式和虛擬機器建立頻繁的時間點副本，提供可靠的復原機制。

資料加密 確保資料在儲存和傳輸過程中均已加密，滿足嚴格的安全要求並保護敏感資訊。

合規與審計 提供工具和功能，幫助組織滿足合規要求並定期稽核其資料保護實務。

災難復原 與 ONTAP 的複寫功能整合，提供強大的災難復原解決方案，即使面對災難性事件也能確保業務連續性。

有關 Trident Protect 的完整詳細資訊，請參閱 NetApp Trident Protect ["文件在此處"](#)。

NetApp ONTAP 硬體型號與支援的通訊協定

NetApp ONTAP 軟體可在多種硬體型號上運行，每種型號都旨在滿足不同的效能和容量需求。Trident 擴展了其強大的功能，以支援各種 NetApp ONTAP 系統，包括全快閃 FAS (AFF)、全 SAN 陣列 (ASA) 和 ASA R2 系統。這種支援確保企業能夠在容器化環境中充分發揮其高效能儲存解決方案的潛力。

全快閃 **FAS (AFF)** 系統 AFF 系統具有卓越的效能和低延遲，使其成為高需求應用程式的理想選擇。

All SAN Array (ASA) 系統 ASA 系統專為專用 SAN 環境而設計，提供強大的效能和可靠性。

ASA R2 Systems ASA R2 系統為 SAN 環境提供增強的功能和效能。

AFX NetApp AFX 是一種專為企業級 AI 工作負載而設計的解耦式全快閃儲存架構。它提供高效能 NAS，支援運算能力和容量的獨立擴充。NetApp AFX 儲存系統與其他基於 ONTAP 的系統 (ASA、AFF 和 FAS) 的差異在於其儲存層的實作方式。AFX 使用分散式檔案系統，從而實現高效能和可擴充性，而其他基於 ONTAP 的系統則採用傳統的儲存架構。

1. AFF 支援的協定：NFS、CIFS/SMB、iSCSI、FC、NVMe/TCP
2. ASA 支援的協定：iSCSI、FC、NVMe/TCP
3. ASA R2 支援的協定：iSCSI、FC、NVMe/TCP
4. AFX 支援的協定：從 Trident 25.10 開始，僅支援 NFS 協定；不支援 SMB 協定。

用於 ONTAP 儲存的 Trident 驅動程式

Trident 包含以下 CSI 驅動程式，每個驅動程式都能夠在後端儲存中佈建磁碟區。

適用於支援硬體型號上的 **NAS** 通訊協定

1. ontap-nas：一個 PVC 映射到一個 PV，該 PV 是一個專用的 flexvolume
2. ontap-nas-economy：每個配置的 PV 都是一個 qtree，每個 FlexVolume 可配置的 qtree 數量（預設值為 200，可配置範圍為 50 到 300）。一個 PVC 映射到一個 PV，該 PV 是共享 flexvolume 上的一個 qtree。



您可以將所有虛擬機器的磁碟作為 qtree 放置在單一磁碟區中。但是，請注意，Trident 不支援快照、複製和複寫功能。當這些功能由儲存管理員管理時，它們並非針對單一 PV 進行細粒度控制。

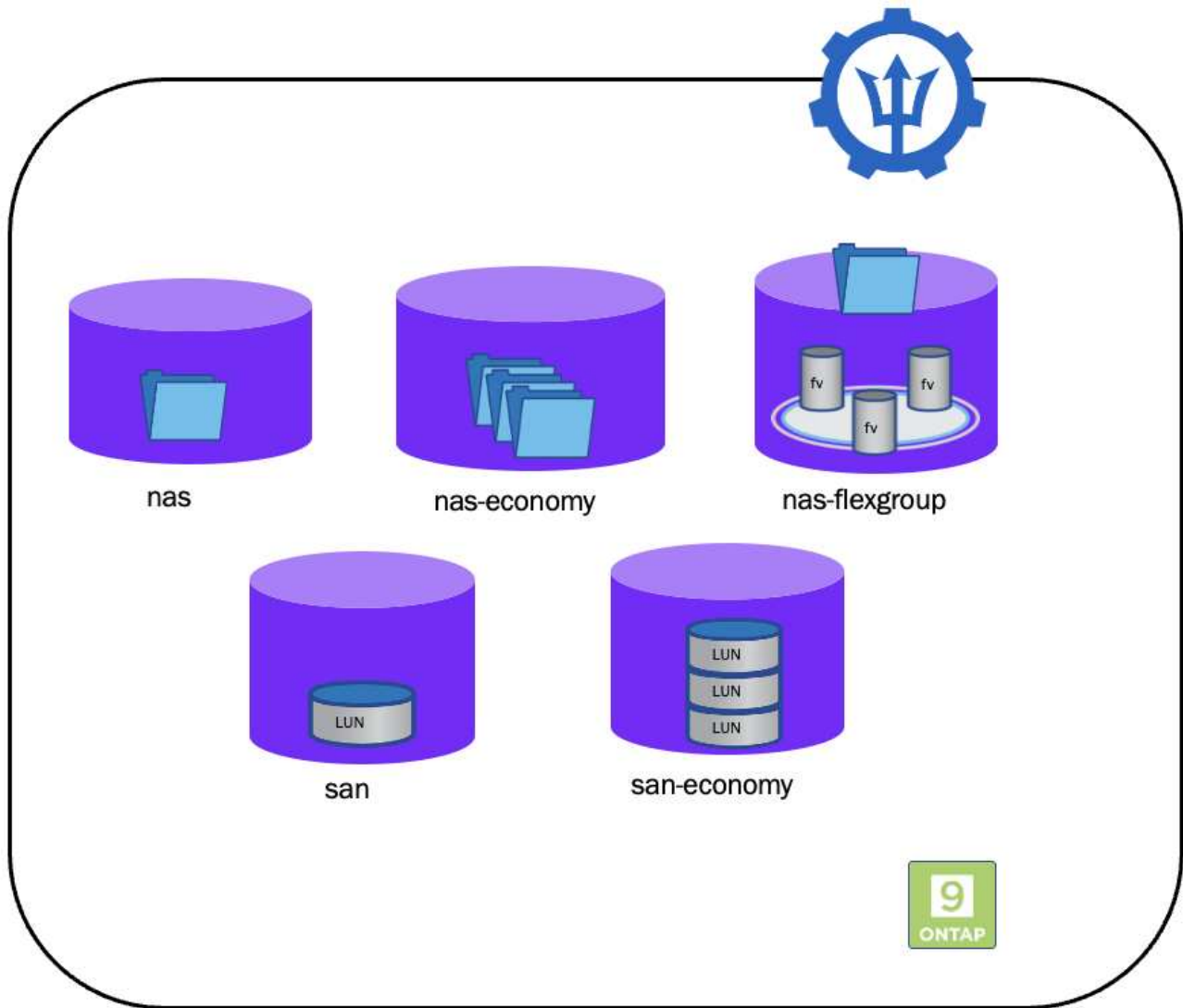
1. ontap-nas-flexgroup：每個配置的 PV 都是一個完整的 ONTAP FlexGroup，並且會使用分配給 SVM 的所有集合體。

對於受支援硬體型號上的 **SAN** 通訊協定

1. ontap-san：一個 PVC 映射到一個 PV，該 PV 是專用 FlexVol 上的一個 LUN。
2. ontap-san-economy：一個 PVC 映射到一個 PV，PV 是共享 FlexVolume 上的一個 LUN。共享 FlexVolume 中可以有多個 LUN。（預設值為 100，每個 FlexVolume 磁碟區可配置為 50 到 200 個 LUN/PV）



您可以將所有虛擬機器的磁碟作為 LUN 放置在單一磁碟區中。但是，此驅動程式雖然支援 Snapshots 和 Clones，但不支援 Replication。當 Replication 由儲存管理員管理時，它不是以 PV 為單位的細粒度控制。



有關透過 Trident 驅動程式公開的功能以及必須由儲存管理員套用的功能的完整清單，請參閱 Trident 文件中的["ONTAP 後端驅動程式"](#)章節。

在 Red Hat OpenShift 叢集上安裝 Trident 並建立儲存物件

使用經過 Red Hat 認證的 Trident Operator 安裝 Trident，並為 ONTAP 和 Amazon FSx for NetApp ONTAP 建立儲存物件，以啟用容器和 VM 的動態磁碟區資源配置。必要時準備工作節點以進行區塊存取。

開始之前

- 請先完成此頁面上的步驟，然後再安裝 OpenShift Virtualization。OpenShift Virtualization 需要一個預設的 Trident 支援的 StorageClass 和 VolumeSnapshotClass 用於建立虛擬機範本的黃金鏡像。

- 如果您在設定 Trident 之前已安裝 OpenShift Virtualization，請刪除所有使用其他儲存類別建立的黃金鏡像。將 Trident 設為預設儲存空間後，OpenShift Virtualization 將使用 Trident 儲存重新建立黃金鏡像。

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

步驟 1：安裝 Trident

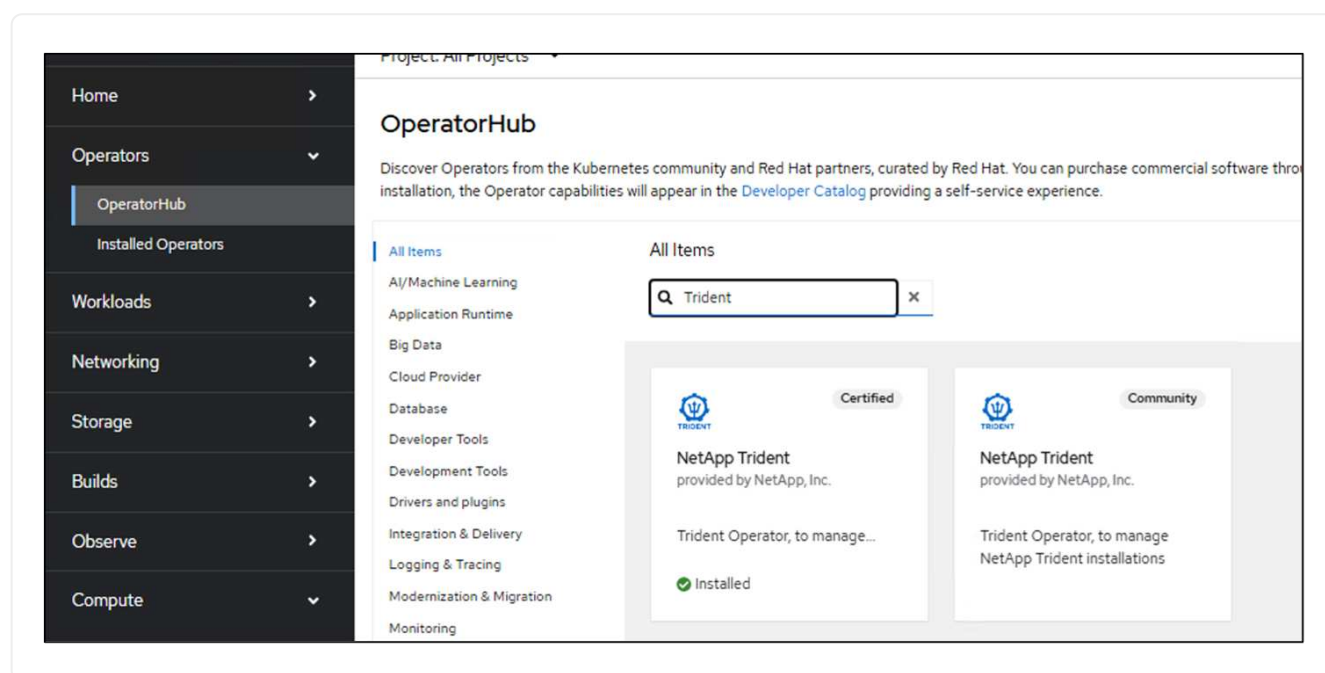
Red Hat 認證的 Trident Operator 由 NetApp 支援，適用於 OpenShift 內部部署、公有雲以及 ROSA 等託管服務。從 Trident 25.02 開始，當您使用 Amazon FSx for NetApp ONTAP 並計劃執行 OpenShift Virtualization VM 工作負載時，該 Operator 也可以為 iSCSI 準備工作節點。

有關其他安裝選項，請參閱 "[Trident 說明文件](#)"。

步驟

1. 在 **OperatorHub** 中、選取 **Certified NetApp Trident**。

顯示範例



2. 在 **Install** 頁面上，保持最新版本並選擇 **Install**。

OperatorHub > Operator Installation

Install Operator

Install your Operator by subscribing to one of the update channels to keep the Operator up to date. The strategy determines either manual or automatic updates.

Update channel *

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. 安裝操作員後，選擇 **View operator** 並建立 Trident Orchestrator 實例。

如果要為 iSCSI 準備工作節點，請切換到 YAML 視圖並新增 `iscsi`至`nodePrep`。`

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. 確認叢集中所有 Trident Pod 都正在運作。

顯示範例

```

[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. 如果您啟用了 iSCSI 節點準備，請登入工作節點並驗證 `iscsid` 和 `multipathd` 是否處於活動狀態，以及 `multipath.conf` 是否具有條目。

顯示範例

```
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#
```

顯示範例

```
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#
```

```
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#
```

影片示範

以下影片示範如何使用紅帽認證的 Trident Operator 安裝 Trident。

[在 OpenShift 中使用經過認證的Trident Operator 安裝Trident 25.02.1](#)

步驟 2：為您的環境準備儲存後端和 **StorageClass** 設定檔

為您的環境建立 TridentBackendConfig 和 StorageClass 定義。您可以在環境中設定多種儲存協定。為要使用的每種協定建立 YAML 檔案，並將佔位符值替換為您的特定組態詳細資料。



根據您的環境，完成下面的內部部署或 ROSA 部分，然後繼續執行步驟 3。

內部部署 OpenShift 叢集

為每個要設定的協定建立 YAML 檔案。您可以設定以下一個或多個協定：NAS（用於基於 NFS 的檔案儲存）、iSCSI（用於 iSCSI 區塊儲存）、NVMe/TCP（用於高效能 NVMe over TCP 區塊儲存）或 FC（用於 Fibre Channel 區塊儲存）。

為每個協定建立一個 TridentBackendConfig 和 StorageClass。後端設定包含儲存在 Kubernetes Secret 中的憑證。

後端配置基本選項的重要指南

1. 後端設定檔參考您的 **ONTAP SVM** 和管理 **LIF**。+
2. 您可以使用使用者名稱和密碼對 ONTAP 進行驗證。可以是叢集管理員或 vservers 管理員的使用者名稱和密碼。+
3. 您也可以使用用戶端憑證進行 ONTAP 身份驗證。有關如何產生用戶端憑證、私密金鑰和 CA 憑證的詳細資訊，請參閱 ["Trident 文件在此"](#)。

NAS 經濟型驅動程式後端組態選項的基本指南

以下參數可用於配置 **NAS economy driver**：

1. **qtreesPerFlexvol** 該值必須在 50-300 範圍內，預設值為 200。+
2. **limitVolumePoolSize** 此值是您在 ontap-nas-economy 後端使用 qtrees 時所要求的最大 FlexVol 大小。未設定預設值，且預設不強制執行此參數。如果設定此參數，Trident 運算子會在目前的 FlexVol 達到指定大小時建立新的 FlexVol。此參數可透過限制用於佈建 PV 的 FlexVols 大小，協助您控制叢集中的容量使用率。+
3. **denyNewVolumePools** 此參數限制 ontap-nas-economy 後端建立新 FlexVol 磁碟區來包含其 qtree。僅使用預先存在的 FlexVol 來佈建新的 PV。

SAN 驅動程式後端組態選項的基本準則

以下參數可用於 **SAN** 驅動程式配置：

1. **limitVolumeSize** 此值是您在 SAN 後端時要求的最大磁碟區大小。未設定預設值，且預設不強制執行此參數。如果您設定此參數，則當請求的磁碟區大小超過此值時，Trident operator 將無法完成佈建。您可以使用此參數來限制其為 LUN 管理的磁碟區的最大大小。+
2. **lunsPerFlexvol** 每個 FlexVol 的最大 LUN 數量，必須在 [50, 200] 範圍內，預設值為 100。如果設定此參數，Trident operator 會在目前 FlexVol 的 LUN 數量達到指定值時建立新的 FlexVol。此參數可透過限制每個 FlexVol 用於佈建 PV 的 LUN 數量，協助您控制叢集中的容量使用率。

以下參數可用於配置 **SAN economy driver**：

1. **limitVolumePoolSize** 此值是您在使用 SAN 經濟型後端時所要求的最大 FlexVol 容量。未設定預設值，且預設不強制執行此參數。如果設定此參數，Trident 操作員將限制可用於在 flexvol 中建立 LUN 的最大 flexvol 容量。此參數可透過限制用於配置 PV 的 FlexVols 容量大小，協助您控制叢集中的容量使用率。+
2. **denyNewVolumePools** 此參數限制 SAN 經濟型後端建立新的 FlexVol 磁碟區來包含其 LUN。僅使用預先存在的 Flexvols 來佈建新的 PV。



在每個協定提供的範例 YAML 檔案中，如果未提供 `storagePrefix`，則使用預設前綴「trident」。當使用 `ontap-nas-economy` 且 `storagePrefix` 為 24 個字元或更多字元時，`qtree` 中不會嵌入儲存前綴，但磁碟區名稱中會包含該前綴。

當您直接從系統管理員或 CLI 在 ONTAP 中建立磁碟區時，若未為所有選用參數提供值，則在磁碟區佈建期間會自動設定一些預設值，如下所示。

```
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
```

以下是一個使用 Trident 後端設定建立的磁碟區範例。除了名稱模板之外，其預設值部分沒有明確設定任何選項。

```
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
```

使用此後端組態建立磁碟區時，後端組態檔案中未設定的選項將設定為如下所示的預設值。

```
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          n
one
```

但是，當您在 Trident 後端配置的預設部分設定值時，您可以建立具有特定參數值的磁碟區。在下列範例中，`snapshotPolicy` 設定為預設原則（如果您未明確設定任何內容，則為 `none`），並且 `snapshotDir` 在後端配置 YAML 的預設部分中設定為 `true`。透過這些設定，您可以看到後端磁碟區獲得了這些值。（`snapdir-access` 為 `true`，`snapshot-policy` 為 `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
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      snapshot-policy  snapdir-access
oc-test-automation testautomation_default_test_trident_set_defaults_7567a true      default
```

其他所有附加參數，請參閱 Trident 文件：

1. "Trident NAS 後端設定選項" +
2. "Trident SAN 後端設定選項"

NAS

為 ONTAP NAS 建立 TridentBackendConfig 和 StorageClass，以啟用基於 NFS 的持久性儲存配置。後端設定包含儲存在 Kubernetes Secret 中的憑證，並引用您的 ONTAP SVM 和管理 LIF。

使用使用者名稱和密碼認證的後端金鑰和後端設定檔案範例（另存為 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
  }}_{{ .volume.RequestName }}"
  credentials:
    name: tbc-nas-secret
```

使用用戶端憑證認證的後端金鑰和後端設定檔案範例（另存為 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 定義（另存為 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 經濟

建立 TridentBackendConfig 和 StorageClass，以供 ONTAP NAS 經濟型驅動程式使用，並透過 qtree 啟用以 NFS 為基礎的持續性儲存佈建。後端組態包含儲存在 Kubernetes Secret 中的認證，並參照您的

ONTAP SVM 和管理 LIF。

使用使用者名稱和密碼認證的後端金鑰和後端設定檔案範例（另存為 `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 定義（另存為 `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
```



有關 ontap-nas-economy 驅動程式後端組態檔案中可包含的其他參數詳細資訊，請參閱上方的「NAS 經濟型驅動程式後端組態選項的基本準則」一節。

iSCSI SAN

為 ONTAP SAN 建立 TridentBackendConfig 和 StorageClass，並搭配 iSCSI 以啟用基於 iSCSI 的區塊儲存佈建。後端組態使用 ontap-san 驅動程式，並包含儲存在 Kubernetes Secret 中的認證。若省略 sanType 參數，則預設使用 iSCSI 通訊協定。

使用使用者名稱密碼認證的後端金鑰和後端設定檔案（另存為 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

```

使用用戶端憑證認證的後端金鑰和後端設定檔案（另存為 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 定義 (另存為 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
```

NVMe/TCP SAN

建立 TridentBackendConfig 和 StorageClass 用於 ONTAP SAN 的 NVMe over TCP 連接，以實現高效能區塊儲存配置。後端配置使用針對 NVMe/TCP 傳輸最佳化的 ontap-san 驅動程序，並包含儲存在 Kubernetes Secret 中的憑證。

使用使用者名稱密碼認證的後端金鑰和後端設定檔案（另存為 `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
```

使用用戶端憑證認證的後端金鑰和後端設定檔案（另存為 `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 定義 (另存為 `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

```



如果您打算將 ReadWriteMany (RWX) 磁碟區與 NVMe 協定搭配使用，則單一 RWX NVMe 磁碟區在 Kubernetes 叢集中最多只能由 64 個節點掛載。如需有關此限制的詳細資訊，以及有關 Trident 26.02 中引入的 Super-subsystem 模型和 Trident 26.02 之前的 Per-volume subsystem 模型的詳細資訊，請參閱 Trident 文件：<https://docs.netapp.com/us-en/trident/trident-use/ontap-san-prep.html#nvme-tcp-considerations>。

FC SAN

建立 TridentBackendConfig 和 StorageClass 以用於 ONTAP SAN 與光纖通道 (Fibre Channel)，以啟用基於 FC 的區塊儲存配置。後端配置使用 ontap-san 驅動程式，並指定了 FCP 協定，同時包含儲存在 Kubernetes Secret 中的憑證。

使用使用者名稱/密碼驗證方式或用戶端憑證驗證方式對 ONTAP 進行驗證。有關如何產生用戶端憑證、私密金鑰和 CA 憑證的詳細資訊，請參閱 "[Trident 文件在此](#)"。以下提供兩種方法的範例。

顯示範例

使用 ONTAP 使用者名稱和密碼進行身份驗證的後端金鑰和後端設定檔案（另存為 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
  }}"
    credentials:
      name: tbc-fc-secret
```

使用用戶端憑證認證的後端金鑰和後端設定檔案（另存為 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
```

建立 TridentBackendConfig 和 StorageClass，以搭配 iSCSI 使用 ONTAP SAN economy，進而啟用基於 iSCSI 的區塊儲存佈建。後端組態使用 ontap-san-economy 驅動程式搭配 iSCSI 通訊協定，並包含儲存於 Kubernetes Secret 中的認證。

使用用戶端憑證認證的後端金鑰和後端設定檔案（另存為 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"
```

查看 Trident 文件，以了解更多 yaml 檔案範例，以及 SAN 驅動程式和 NAS 驅動程式支援的存取模式和磁碟區模式資訊。

1. ["使用 NAS 驅動程式的範例"](#)
2. ["使用 SAN 驅動程式的範例"](#)

ROSA 叢集與 Amazon FSx for NetApp ONTAP

為每個要設定的協定建立 YAML 檔案。您可以設定以下一種或兩種協定：用於基於 NFS 的檔案儲存的 NAS，或用於區塊儲存的 iSCSI。

NAS

建立 TridentBackendConfig 和 StorageClass 適用於 Amazon FSx for NetApp ONTAP 和 ONTAP NAS 的配置，以在 ROSA 叢集上啟用基於 NFS 的持久性儲存配置。後端設定使用 Amazon FSx for NetApp ONTAP DNS 名稱作為管理和資料 LIF，並包含儲存在 trident 命名空間中的 Kubernetes Secret 中的憑證。

後端密鑰和後端組態檔（另存為 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 定義（另存為 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

建立 TridentBackendConfig 和 StorageClass 適用於 Amazon FSx for NetApp ONTAP 和 ONTAP SAN 的配置，以便在 ROSA 叢集上啟用基於 iSCSI 的區塊儲存配置。後端配置使用 ontap-san 驅動程式，並包含儲存在 Kubernetes Secret 中的憑證。確保工作節點已準備好進行 iSCSI 存取。

後端密鑰和後端組態檔（另存為 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 定義（另存為 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
```

步驟 3：建立 VolumeSnapshotClass 設定檔

為本地部署和 ROSA 部署建立 VolumeSnapshotClass 定義。此配置啟用持久卷的基於快照的操作。

VolumeSnapshotClass 定義（另存為 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
```

步驟 4：將組態檔套用至叢集

將您在前面步驟中建立的組態檔套用到您的 OpenShift 叢集。

步驟

1. 為您配置的每個協定套用 TridentBackendConfig 和 StorageClass 檔案。

對於內部部署叢集：

```
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
```

對於 ROSA 叢集：

```
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. 應用 VolumeSnapshotClass 配置。

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

3. 確認資源已成功建立。

檢查 TridentBackendConfig 物件：

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

檢查 StorageClass 物件：

```
oc get storageclass
```

檢查 VolumeSnapshotClass：

```
oc get volumesnapshotclass
```

步驟 5：設定預設的 Trident 儲存和快照類別

將 Trident StorageClass 和 VolumeSnapshotClass 設定為 OpenShift 叢集中的預設值。這是 OpenShift Virtualization 為 VM 範本建立黃金映像來源所必需的。

步驟

1. 設定預設的 Trident StorageClass。

將 Trident 支援的 StorageClass 設定為叢集預設值，以便在未指定儲存類別時，PersistentVolumeClaims 自動使用它。您需要設定兩個註解：一個用於叢集範圍的預設值，另一個專門用於 OpenShift Virtualization。

a. 設定叢集範圍的預設 StorageClass 註解。

確保只將其中一個 StorageClass 設定為預設值。如果另一個 StorageClass 已設定為預設值，請將其註解設為 false。

在主控台中，編輯註釋：

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

從 CLI：

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

b. 設定 OpenShift Virtualization 特有的預設註解。

OpenShift Virtualization 使用特定的註解，其優先順序高於叢集的一般 `is-default-class` 註解。如果另一個 StorageClass 已設定為預設值，請將其註解設為 false。

在主控台中，編輯註釋：

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

從 CLI：

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

2. 設定預設的 Trident VolumeSnapshotClass。

將 Trident 支援的 VolumeSnapshotClass 設定為叢集預設設備，以啟用持久性磁碟區的快照操作。這樣可確保 VolumeSnapshots 在未指定快照類別時自動使用 Trident CSI 驅動程式，並允許 OpenShift Virtualization

建立黃金映像的快照。

確保只將其中一個 VolumeSnapshotClass 設定為預設值。如果另一個 VolumeSnapshotClass 已設定為預設值，請將其註解設為 false。

在主控台中，編輯註釋：

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

從 CLI：

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

使用ONTAP部署 Red Hat OpenShift Virtualization 的需求

查看使用 ONTAP 儲存系統部署 OpenShift 虛擬化的先決條件和其他詳細資訊。

先決條件

- 安裝在裸機基礎架構上並帶有 RHCOS 工作節點的 Red Hat OpenShift 叢集（高於 4.6 版本）
- 部署機器健康檢查以維護虛擬機器的高可用性
- NetApp ONTAP 叢群，其中 SVM 配置了正確的協定。
- 在 OpenShift 叢集上安裝 Trident
- 建立 Trident 後端配置
- 在 OpenShift 叢集上設定 StorageClass，並使用 Trident 作為配置器
- 在以 Trident 為佈建程式的 OpenShift 叢集上設定的 Trident VolumeSnapshotClass
- Trident StorageClass 配置為 OpenShift 叢集或 OpenShift Virtualization 的預設設定。

有關上述 Trident 先決條件，請參閱["Trident 安裝部分"](#)了解詳情。

- 叢集管理員存取 Red Hat OpenShift 叢集
- NetApp ONTAP 叢集的管理員存取權限
- 安裝了 tridentctl 和 oc 工具並新增至 \$PATH 的管理員工作站

由於 OpenShift 虛擬化是由安裝在 OpenShift 叢集上的運算子管理，因此會對記憶體、CPU 和儲存產生額外的負荷，在規劃叢集的硬體需求時必須將這些因素納入考量。如需詳細資訊，請參閱文件["這裡"](#)。

您也可以透過設定節點放置規則，指定 OpenShift 叢集節點的子集來託管 OpenShift Virtualization 的操作員、控制器和虛擬機器。若要設定 OpenShift Virtualization 的節點放置規則，請參閱相關文件["這裡"](#)。

對於支援 OpenShift Virtualization 的儲存，NetApp 建議使用專用的 StorageClass，該 StorageClass 從特定

的Trident後端請求存儲，而該後端又由專用的 SVM 支援。這對於為 OpenShift 叢集上基於 VM 的工作負載提供的資料保持了一定程度的多租用戶等級。

NetApp Trident CSI、儲存設定檔、OpenShift 虛擬化磁碟區存取模式

在 OpenShift 虛擬化中，啟動來源是用於從範本佈建虛擬機器（VM）的磁碟映像。它們通常以 DataVolumes 或持久性磁碟區宣告（PVC）的形式儲存在 openshift-virtualization-os-images 命名空間中。為了使 OpenShift 虛擬化正常運作，在使用 Operator 安裝 OpenShift 虛擬化之前，必須完成兩項儲存配置任務：

1. 您必須為叢集設定預設的 StorageClass。否則，OpenShift Virtualization 將無法自動匯入開機來源映像。
2. 如果您的儲存設備提供者未被 CDI 識別，您必須設定儲存設定檔。儲存設定檔會根據相關聯的 StorageClass 提供建議的儲存設定。StorageProfile 的關鍵功能之一是為 StorageClass 定義複製策略。這決定了磁碟區的複製方式（例如：使用快照、CSI 複製或主機協助的複製），並根據底層儲存設備廠商的功能來最佳化效能與資源使用率。+ Trident CSI 驅動程式可被 CDI 識別，且系統會自動為叢集中建立的任何 Trident StorageClass 建立預設儲存設定檔。此預設儲存設定檔的複製策略設定為快照，這是 Trident CSI 驅動程式建議的複製策略。在安裝 OpenShift Virtualization 之前，您必須為 Trident 建立 VolumeSnapshotClass 和 StorageClass，並將 StorageClass 設定為預設值。這可確保開機來源映像在 openshift-virtualization-os-images 命名空間中以 VolumeSnapshots 的形式提供，以便能夠有效率且無誤地佈建 VM。

對於某些支援的作業系統（例如 RHEL、Fedora 或 CentOS），OpenShift 虛擬化會自動提供並更新啟動來源。這些啟動來源會在安裝 OpenShift 虛擬化時下載，並自動管理和更新至最新的作業系統版本。如果某個作業系統未自動提供啟動來源，則管理員必須手動準備並附加自訂映像。使用 NetApp Trident 作為 CSI 時，您還可以獲得以下額外優勢：

1. 啟動來源映像將以快照的形式儲存在 openshift-virtualization-os-images 命名空間中。佈建 VM 時，啟動映像將從快照複製，從而大幅加快 VM 啟動速度。
2. 系統會為每個建立的 Trident StorageClass 自動建立儲存設定檔，並自動將複製策略設定為快照。存取模式已設定為 ReadWriteMany。無需手動設定。

在您的叢集中安裝 OpenShift Virtualization 之前，建立 Trident StorageClass 和 Trident VolumeSnapshotClass 並將其設定為預設值至關重要。這將確保開機來源映像檔在 openshift-virtualization-os-images 命名空間中可作為 VolumeSnapshots 使用，以便能夠高效且無誤地佈建 VM。

使用 Trident CSI 的另一個優勢：。在 OpenShift Virtualization 中，VM 的開機磁碟需要具有 RWX 存取模式，才能進行 VM 的 Live migration。Trident 支援針對使用 NAS 協定以及使用 SAN 協定（當 volumeMode 設定為 Block 而非 FileSystem 時）所佈建的磁碟區，提供 **ReadWriteMany** 存取模式。您可以使用任一協定作為開機磁碟。Storage profile 物件將根據 StorageClass 設定自動進行設定，且在佈建 VM 時，將會適當地設定開機磁碟的存取模式。

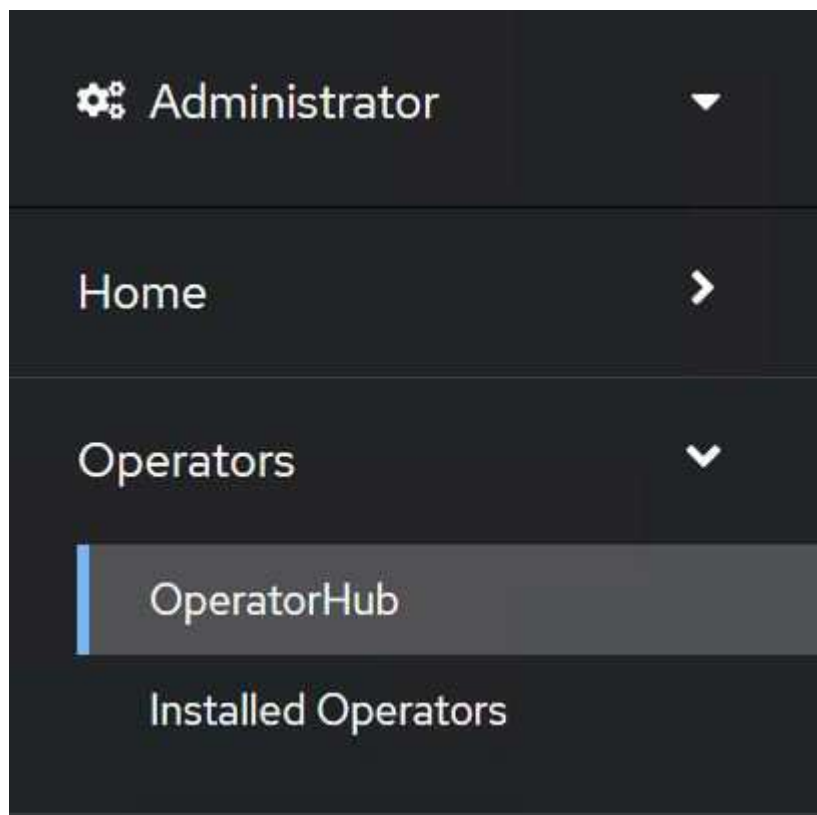
將 Trident StorageClass 設定為叢集的預設值後，您可以為 Trident 建立具有不同參數的其他 StorageClasses，OpenShift 虛擬化系統將自動識別這些 StorageClasses，並為其建立相應的儲存設定檔。然後，您可以在 VM 定義中指定要用於 VM 開機磁碟和資料磁碟的 StorageClass。

在 OpenShift 叢集上安裝 OpenShift Virtualization 是一個簡單的程序，使用 OpenShift OperatorHub 中提供的 OpenShift Virtualization operator 即可。如需如何安裝 OpenShift Virtualization 的詳細步驟，請參閱說明文件 "[這裡](#)"。

在 Red Hat OpenShift 裸機叢集上安裝 OpenShift Virtualization。此過程包括使用叢集管理員存取權限登入、導覽至 OperatorHub 以及安裝 OpenShift Virtualization 操作員。

1. 使用叢集管理員存取權限登入 Red Hat OpenShift 裸機叢集。

2. 從“視角”下拉式選單中選擇“管理員”。
3. 導覽至 Operators > OperatorHub 並蒐索 OpenShift Virtualization。



4. 選擇 OpenShift 虛擬化磁貼並點選安裝。

**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. 在「安裝操作員」畫面上，保留所有預設參數並按一下「安裝」。

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

 OpenShift Virtualization
provided by Red Hat

Provided APIs

HC OpenShift Virtualization Deployment **Required**

Represents the deployment of OpenShift Virtualization

6. 等待操作員安裝完成。

 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. 操作員安裝完成後，按一下建立超融合。



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. 在“建立超融合”畫面上，按一下“建立”，接受所有預設參數。此步驟開始安裝 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. 當 openshift-cnv 命名空間中的所有 pod 都轉為 Running 狀態，並且 OpenShift Virtualization Operator 處於 Succeeded 狀態後，該 Operator 即可使用。現在可以在 OpenShift 叢集上建立虛擬機器。

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

使用 Red Hat OpenShift Virtualization 與 ONTAP 儲存建立虛擬機器

使用 OpenShift 虛擬化建立虛擬機器。此程序包括選擇作業系統範本、設定 StorageClasses，以及自訂虛擬機器參數以符合特定需求。前提條件是，您應已建立 Trident 後端、StorageClass 及磁碟區快照類別物件。如需詳細資訊，請參閱 ["Trident安裝"](#)

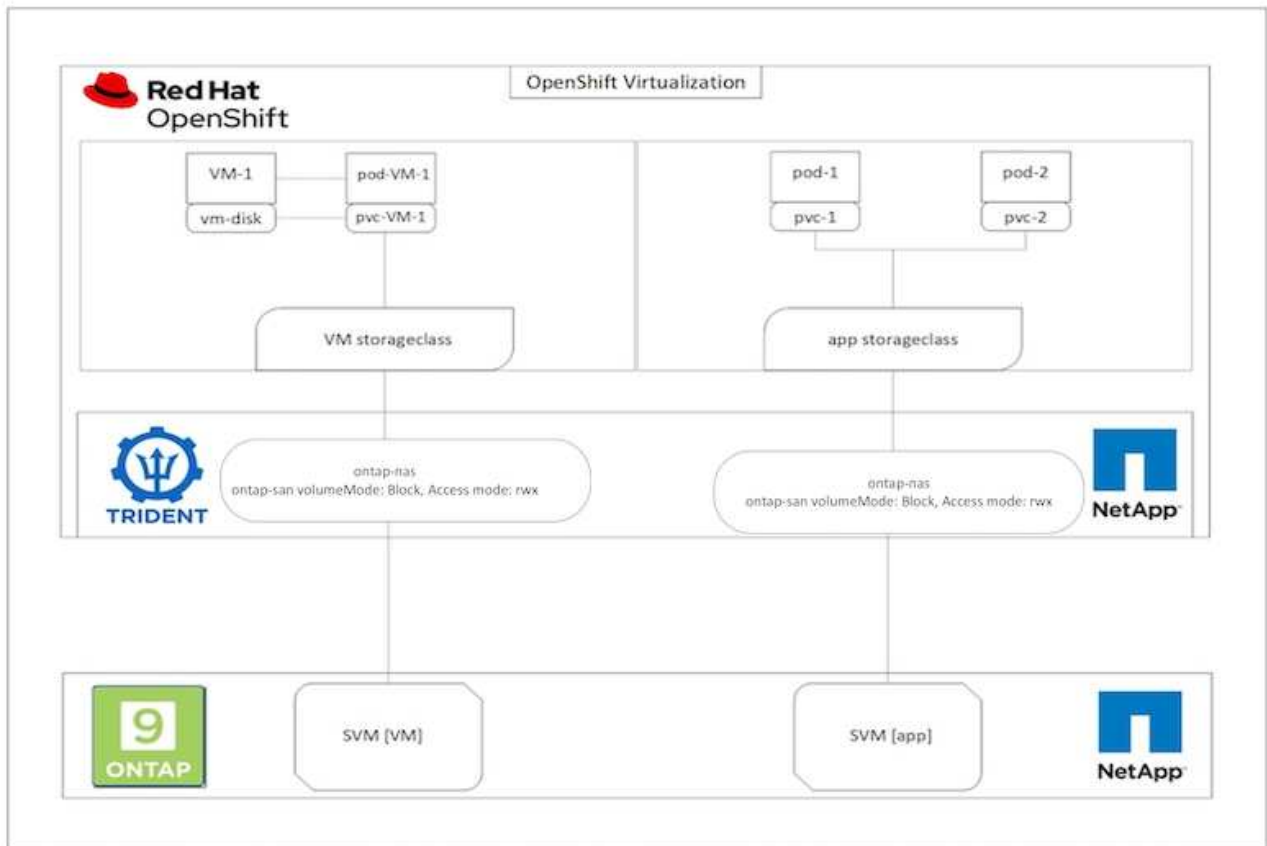
部分"。

建立VM

虛擬機器是有狀態的部署，需要磁碟區來託管作業系統和資料。使用 CNV，由於虛擬機器作為 pod 運行，因此虛擬機器由透過Trident託管在NetApp ONTAP上的 PV 支援。這些磁碟區作為磁碟連接並儲存整個檔案系統，包括虛擬機器的啟動來源。



必須將 Trident StorageClass 設定為叢集上的預設 StorageClass，以確保為虛擬機器磁碟建立的 PVC 使用 Trident 後端進行佈建。叢集上可以有多個 StorageClasses，但必須使用預設 StorageClass 來佈建虛擬機器開機磁碟的 PVC。如有需要，其他磁碟可以使用其他 StorageClasses。



若要在 OpenShift 叢集上快速建立虛擬機，請完成下列步驟：

1. 導航至“虛擬化”>“虛擬機器”，然後按一下“建立”。
2. 選擇來自模板。
3. 選擇所需的可用啟動來源的作業系統。
4. 勾選方塊「建立後啟動虛擬機器」。
5. 按一下快速建立虛擬機器。

虛擬機器已建立並啟動，處於「Running」狀態。它會自動使用預設 StorageClass 為開機磁碟建立 PVC 及對應的 PV。為了將來能夠對虛擬機器進行即時移轉，您必須確保用於磁碟的 StorageClass 能夠支援 RWX 磁碟區。

這是即時移轉的必要條件。ontap-nas 和 ontap-san (iSCSI 和 NVMe/TCP 通訊協定的 volumeMode 區塊) 可支援使用各自 StorageClasses 所建立磁碟區的 RWX 存取模式。

若要在叢集上設定 ontap-san StorageClass，請參閱["配置 StorageClass 的區段"](#)。



點擊「快速建立 VirtualMachine」將使用預設 StorageClass 建立虛擬機器可啟動根磁碟的 PVC 和 PV。您可以選擇不同的 StorageClass 作為磁碟，方法是選取「自訂 VirtualMachine」>「自訂 VirtualMachine 參數」>「磁碟」，然後編輯磁碟以使用所需的 StorageClass。

在配置虛擬機器磁碟時，通常與檔案系統相比，區塊存取模式更受歡迎。

若要在選擇作業系統範本後自訂虛擬機器創建，請按一下「自訂虛擬機器」而不是「快速建立」。

1. 如果選擇的作業系統已經配置了啟動來源，可以點選*自訂虛擬機器參數*。
2. 如果所選作業系統沒有配置啟動來源，則必須配置它。您可以查看["文件"](#)。
3. 配置啟動磁碟後，可以點選*自訂虛擬機器參數*。
4. 您可以從此頁面上的選項卡自訂虛擬機器。例如，按一下「磁碟」選項卡，然後按一下「新增磁碟」為虛擬機器新增另一個磁碟。
5. 按一下「建立虛擬機器」以建立虛擬機器；這會在背景啟動對應的 pod。



當從 URL 或註冊表為模板或作業系統配置啟動來源時，它會在 `openshift-virtualization-os-images` 專案並將 KVM 客戶映像下載到 PVC。您必須確保範本 PVC 具有足夠的配置空間來容納相應作業系統的 KVM 用戶機映像。然後，當使用任何專案中的相應範本建立虛擬機器時，這些 PVC 將被複製並作為根磁碟附加到虛擬機器。

The screenshot shows the OpenShift Virtualization console interface. On the left is a sidebar with navigation links: Administrator, Home, Operators, Workloads, Virtualization (selected), Overview, Catalog, VirtualMachines (active), Templates, and InstanceTypes. The main panel is titled 'Project: openshift-virtualization-os-images' and 'VirtualMachines'. It features a search bar and a table with columns: Name, Status, Conditions, Node, and IP address. There are 4 items in the table, all with a status of 'Running'. A 'Create' button is visible in the top right corner of the table area.

Name	Status	Conditions	Node	IP address
VM centos-stream9-hissing-antelope	Running		ocp-worker3	10.130.0.143
VM centos-stream9-improved-kill	Running		ocp-worker3	10.130.0.145
VM centos-stream9-weary-toucan	Running		ocp-worker3	10.130.0.123
VM centos-stream9-zealous-anaconda	Running		ocp-worker3	10.130.0.127

Create new VirtualMachine

Select an option to create a VirtualMachine from.

Template catalog InstanceTypes

Template project All projects

Default templates

Filter by keyword...

13 items

Operating system

- ☐ CentOS
- ☐ Fedora
- ☐ Other
- ☐ RHEL
- ☐ Windows

Workload

- ☐ Desktop
- ☐ High performance
- ☐ Server

Source available

CentOS Stream 8 VM
centos-stream8-server-small

Project openshift
Boot source PVC (auto import)
Workload Server
CPU 1
Memory 2 GiB

Source available

CentOS Stream 9 VM
centos-stream9-server-small

Project openshift
Boot source PVC (auto import)
Workload Server
CPU 1
Memory 2 GiB

Source available

CentOS 7 VM
centos7-server-small

Project openshift
Boot source PVC (auto import)
Workload Server
CPU 1
Memory 2 GiB

Source available

Fedora VM
fedora-server-small

Project openshift
Boot source PVC (auto import)
Workload Server
CPU 1
Memory 2 GiB

Source available

Red Hat Enterprise Linux 7 VM
rhel7-server-small

Project openshift
Boot source PVC (auto import)
Workload Server
CPU 1
Memory 2 GiB

Source available

Red Hat Enterprise Linux 8 VM
rhel8-server-small

Project openshift
Boot source PVC (auto import)
Workload Server
CPU 1
Memory 2 GiB

Source available

Red Hat Enterprise Linux 9 VM
rhel9-server-small

Project openshift
Boot source PVC (auto import)
Workload Server
CPU 1
Memory 2 GiB

Source available

Microsoft Windows 10 VM
windows10-desktop-medium

Project openshift
Boot source PVC
Workload Desktop
CPU 1
Memory 4 GiB

Source available

Microsoft Windows 11 VM
windows11-desktop-medium

Project openshift
Boot source PVC
Workload Desktop
CPU 2
Memory 4 GiB

Source available

Microsoft Windows Server 2012 R2 VM
windows2k12r2-server-medium

Project openshift
Boot source PVC
Workload Server
CPU 1
Memory 4 GiB

VirtualMachine details

VM centos-stream9-zealous-anaconda Running

VAML Actions

Overview Details Metrics VAML Configuration Events Console Snapshots Diagnostics

Disks

Add disk

Filter Search by name... Mount Windows drivers disk

Name	Source	Size	Drive	Interface	Storage class
cloudinitdisk	Other	-	Disk	virtio	-
data-disk1 Persistent Hotplug	PVC centos-stream9-zealous-anaconda-data-disk1	30.00 GiB	Disk	SCSI	ontap-san-block
rootdisk bootable	PVC centos-stream9-zealous-anaconda	30.00 GiB	Disk	virtio	ontap-san-block

File systems

Name	File system type	Mount point	Total bytes	Used bytes
vdal	xfs	/	29.94 GiB	1.30 GiB

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

影片示範

以下影片示範如何使用 iSCSI 儲存在 OpenShift Virtualization 中建立 VM。

使用區塊儲存在 OpenShift 虛擬化中建立虛擬機

使用其他方法，透過 ONTAP 儲存以 Red Hat OpenShift Virtualization 建立 VM

除了使用 OpenShift 虛擬化使用者介面建立虛擬機器的方法之外，還有其他方法可以在 OpenShift 虛擬化平台上建立虛擬機器。本節概述其中的一些方法。

使用 Red Hat OpenShift Virtualization 從 YAML 檔案建立虛擬機

也可以透過套用 YAML 檔案並使用 `oc apply` 指令來建立虛擬機器。這種方法適用於喜歡使用命令列介面（CLI）或希望使用腳本自動建立虛擬機器的使用者。若要使用此方法建立虛擬機器，您可以建立一個 YAML 檔案來定義虛擬機器及其關聯資源，例如虛擬機器磁碟的 PVC。然後，您可以使用 `oc apply -f <yaml-file>` 指令套用該 YAML 檔案，在 OpenShift 叢集上建立虛擬機器。

```
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

```

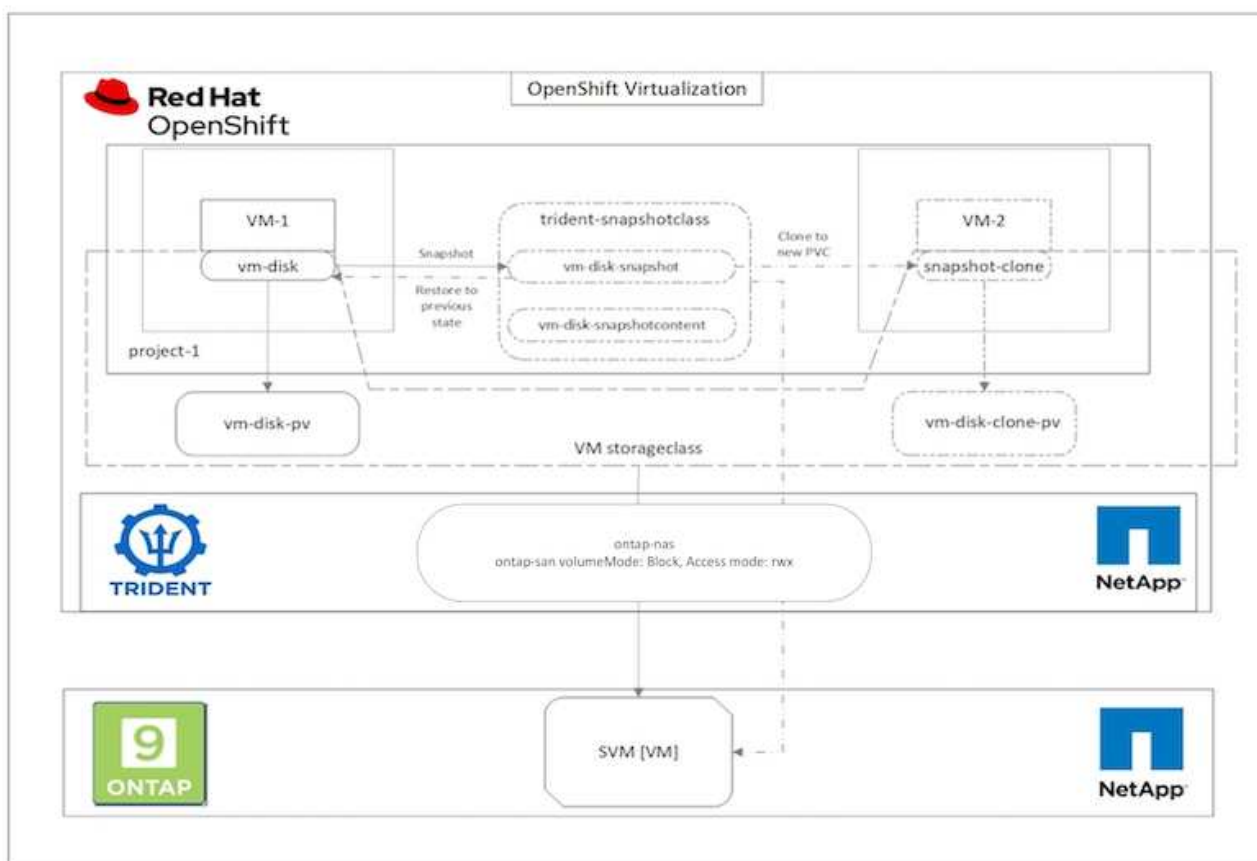
使用 Red Hat OpenShift Virtualization 從快照副本建立 VM

您可以使用 OpenShift 虛擬化技術從快照建立虛擬機器。此程序包括建立 VolumeSnapshotClass、對虛擬機器的持久性磁碟區聲明（PVC）進行快照、將快照還原到新的 PVC，以及部署使用還原後的 PVC 作為根磁碟的新虛擬機器。

透過 Trident 和 Red Hat OpenShift，使用者可以對其所配置的儲存類別上的持久磁碟區進行快照。利用此功能，使用者可以取得磁碟區的某個時間點的副本並使用它來建立新磁碟區或將同一磁碟區還原到先前的狀態。這啟用或支援各種用例，從回滾到克隆到資料恢復。

對於 OpenShift 中的 Snapshot 操作，必須定義資源 VolumeSnapshotClass、VolumeSnapshot 和 VolumeSnapshotContent。

- VolumeSnapshotContent 是從叢集中的磁碟區取得的實際快照。它是類似於用於儲存的 PersistentVolume 的叢集範圍的資源。
- VolumeSnapshot 是建立磁碟區快照的請求。它類似於 PersistentVolumeClaim。
- VolumeSnapshotClass 允許管理員為 VolumeSnapshot 指定不同的屬性。它允許您對從同一磁碟區取得的不同快照設定不同的屬性。



先決條件

您必須擁有一台已連接 PVC 的現有虛擬機，該虛擬機已配置在 Trident 儲存類別上，且該虛擬機必須處於運作狀態，才能對 PVC 進行快照。您必須已建立 VolumeSnapshotClass，才能用於建立 VolumeSnapshot。此項目應已作為安裝 OpenShift 虛擬化的先決條件之一建立。如需詳細資訊，請參閱 "[步驟 3：建立 VolumeSnapshotClass 配置](#)"。

要建立虛擬機器快照，請完成下列步驟：

1. 識別連接到來源虛擬機器的 PVC，然後建立該 PVC 的快照。導航至 `Storage > VolumeSnapshots` 並點擊建立磁碟區快照。
2. 選擇要為其建立 Snapshot 的 PVC，輸入 Snapshot 的名稱或接受預設值，並選擇適當的 VolumeSnapshotClass。然後點擊“創建”。
3. 這將建立該時間點的 PVC 快照。

Create VolumeSnapshot

[Edit YAML](#)**PersistentVolumeClaim ***

PVC rhel8-short-frog-rootdisk-28dvb ▼

Name *

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

Snapshot Class *

VSC trident-snapshot-class ▼

Create

Cancel

從快照建立新虛擬機器

1. 首先，將快照還原到新的 PVC 中。導覽至儲存 > VolumeSnapshots，按一下要復原的快照旁邊的省略號，然後按一下還原為新 PVC。
2. 輸入新 PVC 的詳細信息，然後按一下「恢復」。這將建立一個新的 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. 接下來，從此 PVC 建立一個新的 VM。導航至“虛擬化”>“虛擬機器”，然後按一下“建立”>“使用 YAML”。
2. 在 spec > template > spec > volumes 部分中，指定從快照而不是從容器磁碟建立的新 PVC。根據您的要求提供新虛擬機器的所有其他詳細資訊。

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

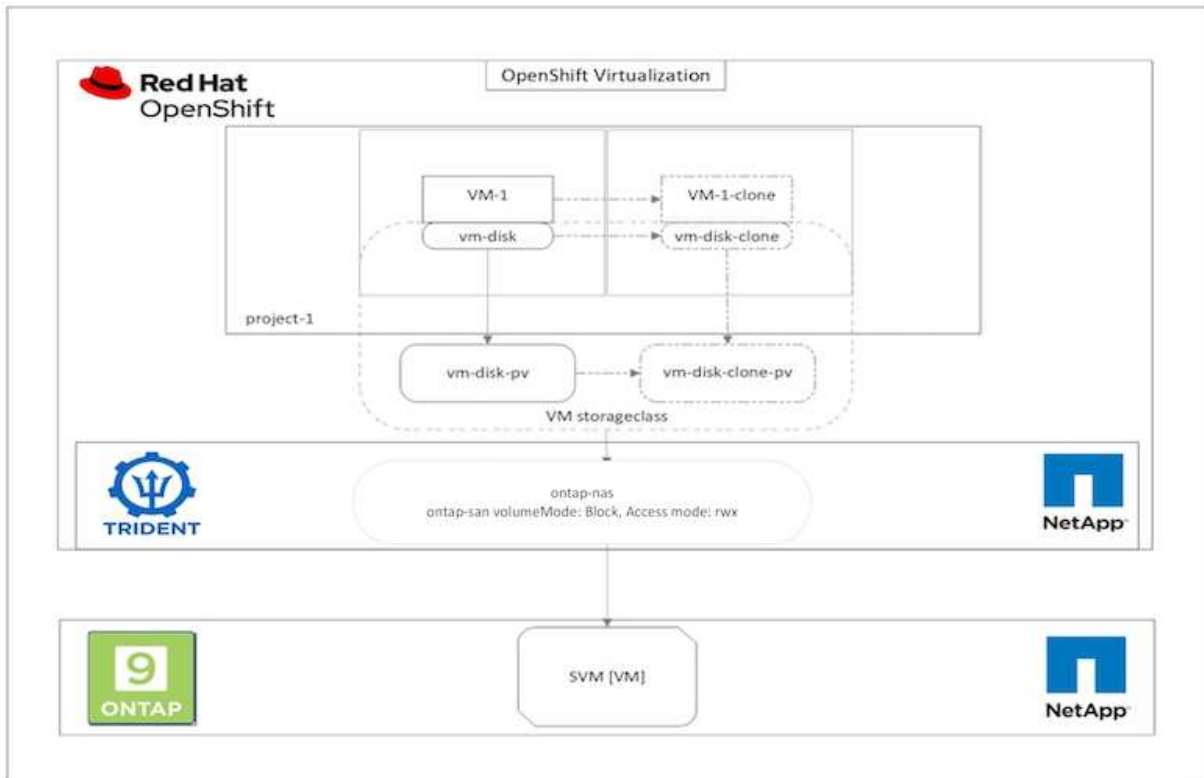
1. 按一下「建立」以建立新的虛擬機器。
2. 虛擬機器建立成功後，存取並驗證新虛擬機器與建立快照時使用 PVC 建立的虛擬機器的狀態是否相同。

使用 Trident 在 OpenShift Virtualization 中複製 VM

此程序包括利用 Trident CSI 磁碟區複製，可讓您透過關閉來源虛擬機器或保持來源虛擬機器運作來建立新的虛擬機器。

VM 複製

在 Trident 的 Volume CSI 克隆功能的支援下，可以克隆 OpenShift 中現有的 VM。CSI 卷克隆允許透過複製 PV 來使用現有 PVC 作為資料來源建立新的 PVC。新的 PVC 建立新之後，它將作為一個獨立的實體運行，並且與來源 PVC 沒有任何連結或依賴關係。



對於 CSI 卷克隆，需要考慮以下某些限制：

1. 來源 PVC 和目標 PVC 必須在同一個專案中。
2. 同一儲存類別內支援克隆。
3. 僅當來源磁碟區和目標磁碟區使用相同的 VolumeMode 設定時才可以執行複製；例如，一個區塊磁碟區只能複製到另一個區塊磁碟區。

OpenShift 叢集中的虛擬機器可以透過兩種方式複製：

1. 透過關閉來源虛擬機
2. 透過保持來源虛擬機器處於活動狀態

透過關閉來源虛擬機器

透過關閉 VM 來複製現有 VM 是 OpenShift 的原生功能，是在 Trident 的支援下實現的。完成以下步驟來複製虛擬機器。

1. 導覽至工作負載>虛擬化>虛擬機，然後按一下要複製的虛擬機旁的省略號。

2. 點擊複製虛擬機器並提供新虛擬機器的詳細資訊。

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.**

3. 按一下複製虛擬機器；這將關閉來源虛擬機器並啟動複製虛擬機器的建立。
4. 此步驟完成後，您可以存取並驗證複製的VM的內容。

透過保持來源虛擬機器運作

也可以透過克隆來源 VM 的現有 PVC，然後使用克隆的 PVC 建立新 VM 來複製現有 VM。此方法不需要您關閉來源虛擬機器。完成以下步驟即可在不關閉虛擬機器的情況下複製虛擬機器。

1. 導覽至儲存 > 持久性磁碟區聲明，然後按一下連接到來源虛擬機器的 PVC 旁的省略號。
2. 點擊克隆 PVC 並提供新 PVC 的詳細資訊。

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. 然後點擊克隆。這會為新的 VM 建立 PVC。
4. 導覽至“工作負載”>“虛擬化”>“虛擬機器”，然後按一下“建立”>“使用 YAML”。
5. 在 spec > template > spec > volumes 部分中，附加克隆的 PVC 而不是容器磁碟。根據您的要求提供新虛擬機器的所有其他詳細資訊。

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

1. 按一下「建立」以建立新的虛擬機器。
2. VM建立成功後，存取並驗證新VM是否為來源VM的複製。

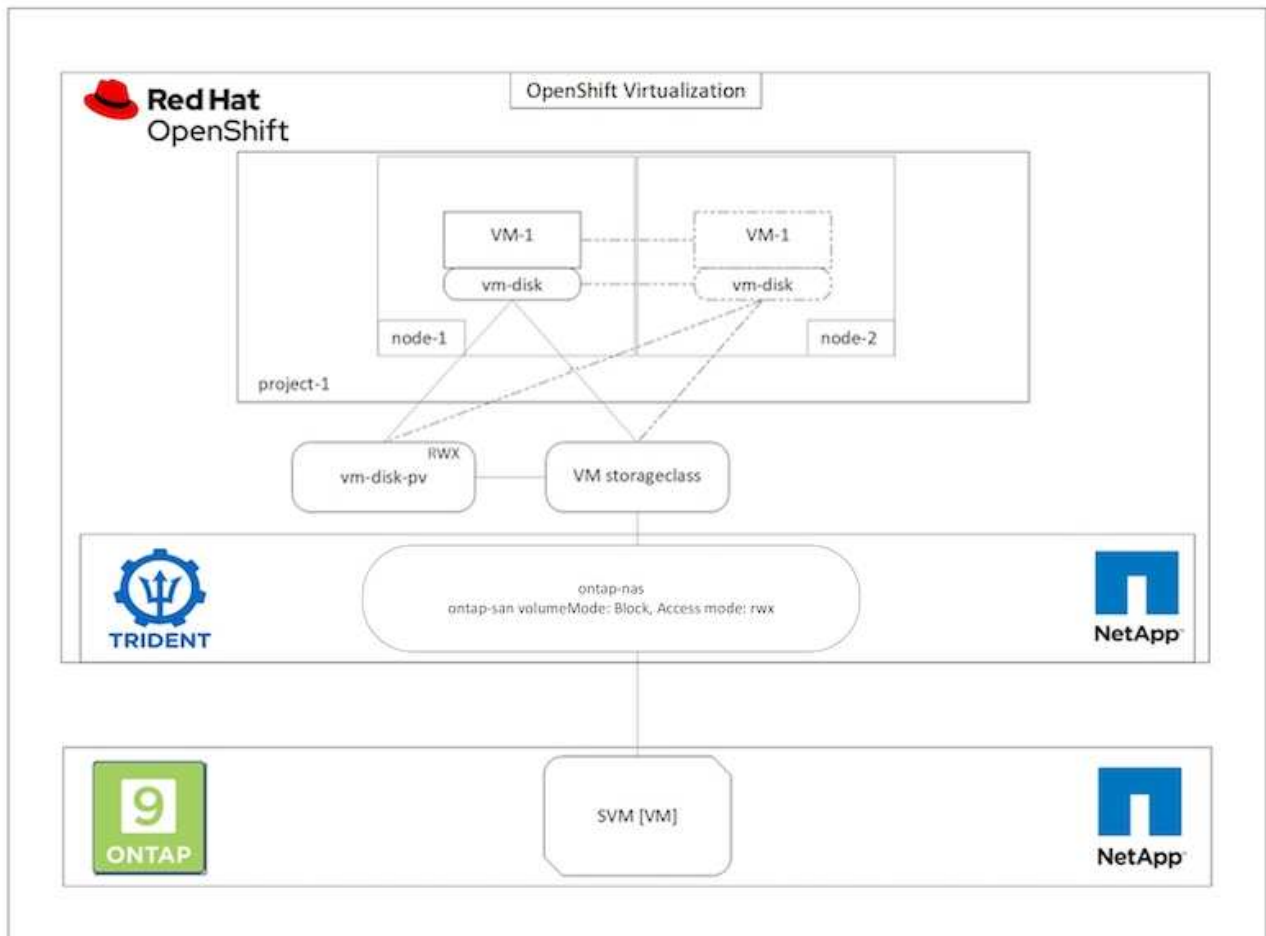
在 Red Hat OpenShift 叢集中的兩個節點之間遷移虛擬機

在叢集中的兩個節點之間遷移 OpenShift Virtualization 中的 VM，無需停機。此過程包括確認磁碟使用與 RWX 相容的儲存類別、啟動遷移以及監控進度。

虛擬機器即時遷移

即時遷移是將虛擬機器執行個體從 OpenShift 叢集中的一個節點遷移到另一個節點的過程，無需停機。為了在 OpenShift 叢集中進行即時遷移，虛擬機器必須綁定到具有共用 ReadWriteMany 存取模式的 PVC。使用 ontap-nas 驅動程式配置的Trident後端支援檔案系統協定 nfs 和 smb 的 RWX 存取模式。請參閱文檔[這裡](#)。使用 ontap-san 驅動程式配置的Trident後端支援 iSCSI 和 NVMe/TCP 協定的區塊磁碟區模式的 RWX 存取模式。請參閱文檔[這裡](#)。

因此，為了成功實現即時遷移，必須使用 ontap-nas 或 ontap-san (volumeMode: Block) 儲存類別為虛擬機器配置具有 PVC 的磁碟（啟動磁碟和附加熱插拔磁碟）。建立 PVC 時，Trident會在啟用 NFS 或 iSCSI 的 SVM 中建立ONTAP區。

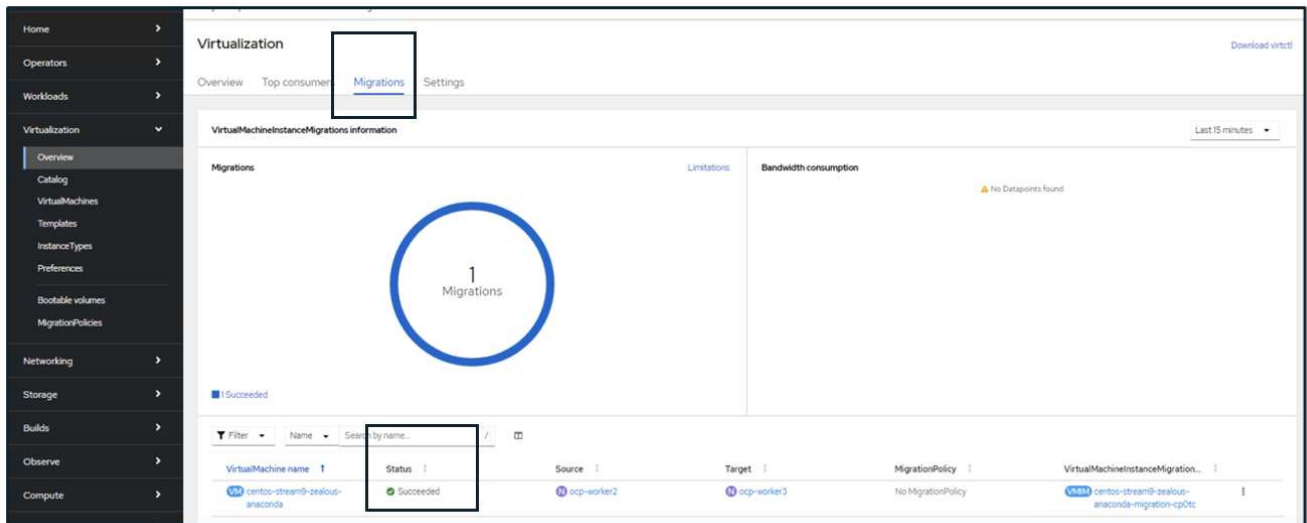
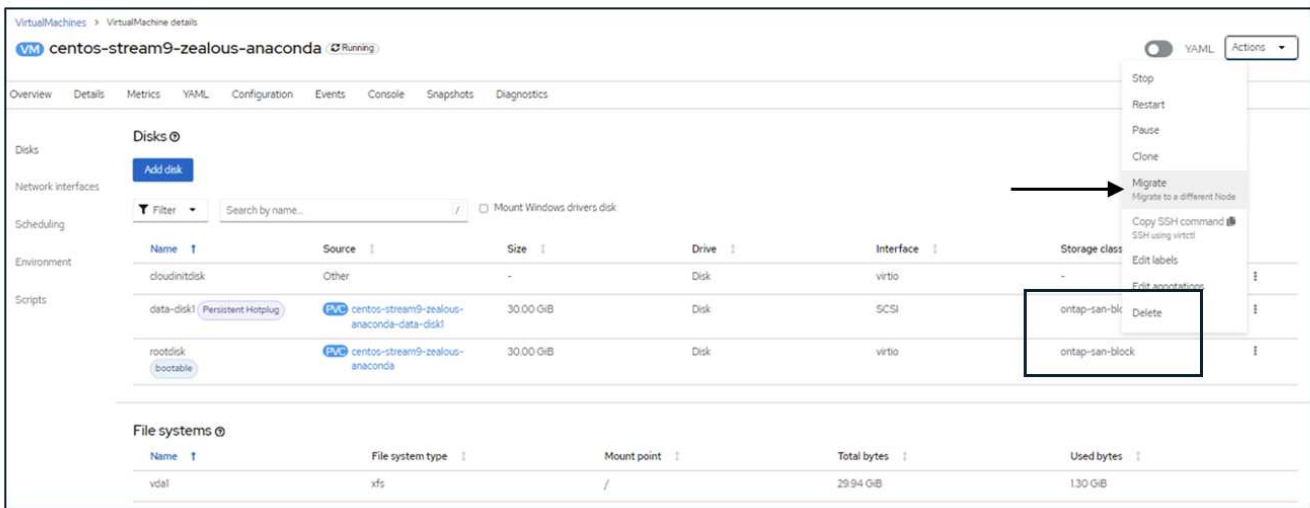


若要對先前已建立且處於運作狀態的 VM 執行即時遷移，請執行下列步驟：

1. 選擇您想要即時遷移的虛擬機器。
2. 按一下“配置”標籤。
3. 確保虛擬機器的所有磁碟都是使用可以支援 RWX 存取模式的儲存類別建立的。
4. 點選右上角的*操作*，然後選擇*遷移*。
5. 若要查看遷移的進度，請前往左側選單上的“虛擬化”>“概述”，然後按一下“遷移”標籤。VM 的遷移將從 待定 過渡到 計劃 再到 成功



如果將 evictionStrategy 設定為 LiveMigrate，則當原始節點處於維護模式時，OpenShift 叢集中的 VM 實例會自動遷移到另一個節點。



OpenShift 虛擬化中虛擬機器在不同儲存類別之間的儲存遷移

在 OpenShift 虛擬化環境中，您可以直接透過 Web 控制台將虛擬機器（VM）的磁碟遷移

到不同的儲存類別，而無需刪除虛擬機器。此程序可讓您遷移整個虛擬機器儲存或特定磁碟區。

以下步驟概述如何遷移虛擬機器的儲存類別：

1. 請確保目標儲存類別已正確配置並在 OpenShift 叢集中可用。這包括已設定並可存取必要的儲存後端（例如 NetApp ONTAP）。

顯示範例

```
[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. 在 OpenShift Container Platform Web 控制台中，導覽至虛擬化透視圖，然後選擇要移轉的 VM。

顯示範例

The image displays two screenshots of the OpenShift Virtualization console interface for a virtual machine named 'rhel9-sc-migration'.

Top Screenshot (Overview): Shows the 'Overview' tab. The VM is in a 'Running' state. Key details include:

- 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

 A VNC console window is visible on the right, showing a terminal output. On the far right, a sidebar shows 'Alerts (0)', 'General' information (Namespace: test-sc-migration, Node: 00-50-56-b5-bd-fa, VirtualMachineInstance: rhel9-sc-migration, Pod: virt-launcher-rhel9-sc-migration-5jkdw, Owner: No owner), and 'Snapshots (0)' with a 'Take snapshot' button.

Bottom Screenshot (Configuration): Shows the 'Configuration' tab, specifically the 'Disks' section. It lists the disks attached to the VM:

- cloudinitdisk:** Source: Other, Size: -, Drive: Disk, Interface: virtio, Storage class: -
- disk-rhel9-nas:** Source: PVC (dv-rhel9-migration-disk-emerald-wren-75-d88usc), Size: 53 GiB, Drive: Disk, Interface: virtio, Storage class: sc-nas
- rootdisk:** Source: PVC (rhel9-migration), Size: 31.8 GiB, Drive: Disk, Interface: virtio, Storage class: sc-nas

 A 'bootable' label is shown next to the rootdisk. Below the disk list, a terminal window shows the output of the command 'oc get vm,pvc -n test-sc-migration', displaying details for the VM and its associated PVCs.

- 按一下「操作」→「遷移」→「遷移儲存」。選擇要遷移的磁碟區和目標儲存類別，然後按一下「遷移」。遷移過程將開始，您可以在虛擬機器的「事件」標籤中監控進度。遷移完成後，虛擬機器的磁碟將使用新的儲存類別。

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

Storage

VirtualMachineInstance VM rhel9-sc-migration

Pod virt-launcher-rhel9-sc-migration-Sjkdw

Owner 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 VM 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 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. 遷移完成後，您可以透過在 Web 控制台中檢查虛擬機器的詳細資訊或使用 OpenShift CLI 來驗證虛擬機器的磁碟現在是否正在使用新的儲存類別。

55

顯示範例

```
[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 ~]# oc get all -n test-sc-migration
Warning: apps.openshift.io/v1 DeploymentConfig is deprecated in v1.24*, unavailable in v1.10000+
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 ~]# 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-ccca45499e09
                    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-5jdkw    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-864d-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_OGBGMIOLG/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_8cct			
▼	oc_automation_pvc_75a885f0_824e_407c_ab29_8cc546dbdb3b	oc-automation-svm	trident_qtree_pool_oc_automation_OGBGMIAOLG	/trident_qtree_pool_oc_automation_OGBGMIAOLG/oc_...

版權資訊

Copyright © 2026 NetApp, Inc. 版權所有。台灣印製。非經版權所有人事先書面同意，不得將本受版權保護文件的任何部分以任何形式或任何方法（圖形、電子或機械）重製，包括影印、錄影、錄音或儲存至電子檢索系統中。

由 NetApp 版權資料衍伸之軟體必須遵守下列授權和免責聲明：

此軟體以 NETAPP「原樣」提供，不含任何明示或暗示的擔保，包括但不限於有關適售性或特定目的適用性之擔保，特此聲明。於任何情況下，就任何已造成或基於任何理論上責任之直接性、間接性、附隨性、特殊性、懲罰性或衍生性損害（包括但不限於替代商品或服務之採購；使用、資料或利潤上的損失；或企業營運中斷），無論是在使用此軟體時以任何方式所產生的契約、嚴格責任或侵權行為（包括疏忽或其他）等方面，NetApp 概不負責，即使已被告知有前述損害存在之可能性亦然。

NetApp 保留隨時變本文所述之任何產品的權利，恕不另行通知。NetApp 不承擔因使用本文所述之產品而產生的責任或義務，除非明確經過 NetApp 書面同意。使用或購買此產品並不會在依據任何專利權、商標權或任何其他 NetApp 智慧財產權的情況下轉讓授權。

本手冊所述之產品受到一項（含）以上的美國專利、國外專利或申請中專利所保障。

有限權利說明：政府機關的使用、複製或公開揭露須受 DFARS 252.227-7013（2014 年 2 月）和 FAR 52.227-19（2007 年 12 月）中的「技術資料權利 - 非商業項目」條款 (b)(3) 小段所述之限制。

此處所含屬於商業產品和 / 或商業服務（如 FAR 2.101 所定義）的資料均為 NetApp, Inc. 所有。根據本協議提供的所有 NetApp 技術資料和電腦軟體皆屬於商業性質，並且完全由私人出資開發。美國政府對於該資料具有非專屬、非轉讓、非轉授權、全球性、有限且不可撤銷的使用權限，僅限於美國政府為傳輸此資料所訂合約所允許之範圍，並基於履行該合約之目的方可使用。除非本文另有規定，否則未經 NetApp Inc. 事前書面許可，不得逕行使用、揭露、重製、修改、履行或展示該資料。美國政府授予國防部之許可權利，僅適用於 DFARS 條款 252.227-7015(b)（2014 年 2 月）所述權利。

商標資訊

NETAPP、NETAPP 標誌及 <http://www.netapp.com/TM> 所列之標章均為 NetApp, Inc. 的商標。文中所涉及的所有其他公司或產品名稱，均為其各自所有者的商標，不得侵犯。