



## 解決方案驗證與使用案例 NetApp Solutions

NetApp  
April 12, 2024

# 目錄

解決方案驗證與使用案例：採用NetApp的Red Hat OpenShift .....	1
部署Jenkins CI/CD管道搭配持續儲存設備：Red Hat OpenShift with NetApp .....	1
使用NetApp ONTAP 功能在Red Hat OpenShift上設定多租戶共享 .....	11
Red Hat OpenShift虛擬化搭配NetApp ONTAP 產品 .....	31
採用NetApp的Red Hat OpenShift上的Kubernetes進階叢集管理 .....	56

# 解決方案驗證與使用案例：採用NetApp的Red Hat OpenShift

本頁所提供的範例包括採用NetApp的Red Hat OpenShift解決方案驗證和使用案例。

- ["使用持續儲存設備部署Jenkins CI/CD管道"](#)
- ["使用NetApp在Red Hat OpenShift上設定多租戶共享"](#)
- ["Red Hat OpenShift虛擬化搭配NetApp ONTAP 產品"](#)
- ["採用NetApp的Red Hat OpenShift上的Kubernetes進階叢集管理"](#)

## 部署Jenkins CI/CD管道搭配持續儲存設備：Red Hat OpenShift with NetApp

本節提供與Jenkins部署持續整合/持續交付或部署（CI/CD）管線的步驟、以驗證解決方案的運作。

### 建立Jenkins部署所需的資源

若要建立部署Jenkins應用程式所需的資源、請完成下列步驟：

1. 建立名為Jenkins的新專案。

# Create Project

Name \*

Display Name

Description

Cancel

Create

2. 在此範例中、我們部署Jenkins搭配持續儲存設備。若要支援Jenkins建置、請建立永久虛擬基礎架構。瀏覽至「儲存設備」>「持續磁碟區宣告」、然後按一下「建立持續磁碟區宣告」。選取建立的儲存類別、確定「持續Volume宣告名稱」為Jenkins、選取適當的大小和存取模式、然後按一下「建立」。

## Create Persistent Volume Claim

[Edit YAML](#)

### Storage Class

 basic ▼

Storage class for the new claim.

### Persistent Volume Claim Name \*

jenkins

A unique name for the storage claim within the project.

### Access Mode \*

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

Permissions to the mounted drive.

### Size \*

100 GiB ▼

Desired storage capacity.

☐ Use label selectors to request storage

Use label selectors to define how storage is created.

[Create](#) [Cancel](#)

## 使用持續儲存設備部署Jenkins

若要使用持續儲存設備來部署Jenkins、請完成下列步驟：

1. 在左上角、將角色從「管理員」變更為「開發人員」。按一下「+新增」、然後從「目錄」中選取。在「依關鍵字篩選」列中、搜尋Jenkins。選取Jenkins Service with Persistent Storage。

## Developer Catalog

Add shared apps, services, or source-to-image builders to your project from the Developer Catalog. Cluster admins can install additional apps which will show up here automatically.

All Items

Languages

Databases

Middleware

CI/CD

Other

Type

☒ Operator Backed (0)

☐ Helm Charts (0)

☒ Builder Image (0)

☒ Template (4)

☐ Service Class (0)

All Items

jenkins

Group By: None ▾

Template

Jenkins  
provided by Red Hat, Inc.

Jenkins service, with persistent storage. NOTE: You must have persistent volumes available in...

Template

Jenkins  
provided by Red Hat, Inc.

Jenkins service, with persistent storage. NOTE: You must have persistent volumes available in...

Template

Jenkins (Ephemeral)  
provided by Red Hat, Inc.

Jenkins service, without persistent storage. WARNING: Any data stored will be lost upon...

Template

Jenkins (Ephemeral)  
provided by Red Hat, Inc.

Jenkins service, without persistent storage. WARNING:

2. 按一下「實體化範本」。



## Jenkins

Provided by Red Hat, Inc.



Instantiate Template

### Provider

Red Hat, Inc.

### Support

[Get support](#)

### Created At

 May 26, 3:58 am

### Description

Jenkins service, with persistent storage.

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


### Documentation

[https://docs.okd.io/latest/using\\_images/other\\_images/jenkins.html](https://docs.okd.io/latest/using_images/other_images/jenkins.html)

3. 根據預設、會填入Jenkins應用程式的詳細資料。根據您的需求修改參數、然後按一下「Create（建立）」。此程序可建立所有必要資源、以支援OpenShift上的Jenkins。

## Instantiate Template

**Namespace \***

 jenkins

**Jenkins Service Name**

jenkins

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

**Jenkins JNLP Service Name**

jenkins-jnlp

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

**Enable OAuth in Jenkins**

true

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

**Memory Limit**

1Gi

Maximum amount of memory the container can use.

**Volume Capacity \***

50Gi

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

**Jenkins ImageStream Namespace**

openshift

The OpenShift Namespace where the Jenkins ImageStream resides.

**Disable memory intensive administrative monitors**

false

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

**Jenkins ImageStreamTag**

jenkins:2

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

**Fatal Error Log File**

false

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

**Allows use of Jenkins Update Center repository with invalid SSL certificate**

false

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

[Create](#) [Cancel](#)



### Jenkins

INSTANT-APP JENKINS

[View documentation](#) [Get support](#)

Jenkins service, with persistent storage.

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

The following resources will be created:

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

4. Jenkins Pod約需10至12分鐘才能進入就緒狀態。

## Pods

Create Pod

Filter by name...

1 Running

0 Pending

0 Terminating

0 CrashLoopBackOff





1 Completed

0 Failed

0 Unknown

Select all filters

1 of 2 Items

Name ↑	Namespace ↑	Status ↑	Ready ↑	Owner ↑	Memory ↑	CPU ↑	
 jenkins-l-c77n9	 jenkins	 Running	1/1	 jenkins-1	-	0.004 cores	⋮





5. 建立Pod之後、請瀏覽至「Networking」（網路）>「Routes」（路由）。若要開啟Jenkins網頁、請按一下Jenkins路由提供的URL。

## Routes

Create Route

Filter by name...

1 Accepted	0 Rejected	0 Pending	Select all filters	1 Item
------------	------------	-----------	--------------------	--------

Name ↓	Namespace ↑	Status	Location ↑	Service ↑	
 jenkins	 jenkins	 Accepted	<a href="https://jenkins-jenkins.apps.rhv-ocp-cluster.cie.netapp.com">https://jenkins-jenkins.apps.rhv-ocp-cluster.cie.netapp.com</a>	 jenkins	⋮

6. 由於在建立Jenkins應用程式時使用OpenShift OAuth、請按一下「使用OpenShift登入」。





7. 授權Jenkins服務帳戶存取OpenShift使用者。

## Authorize Access

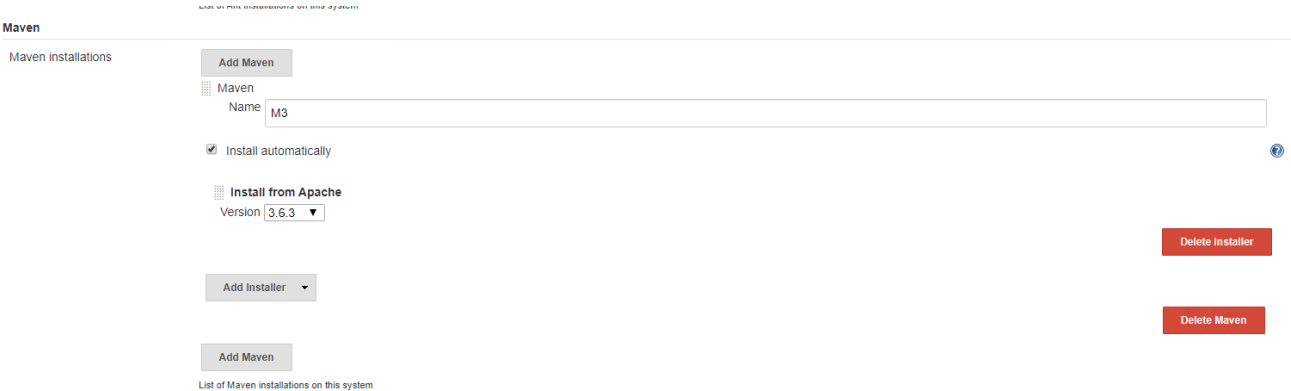
Service account `jenkins` in project `jenkins` is requesting permission to access your account (`kube:admin`)

### Requested permissions

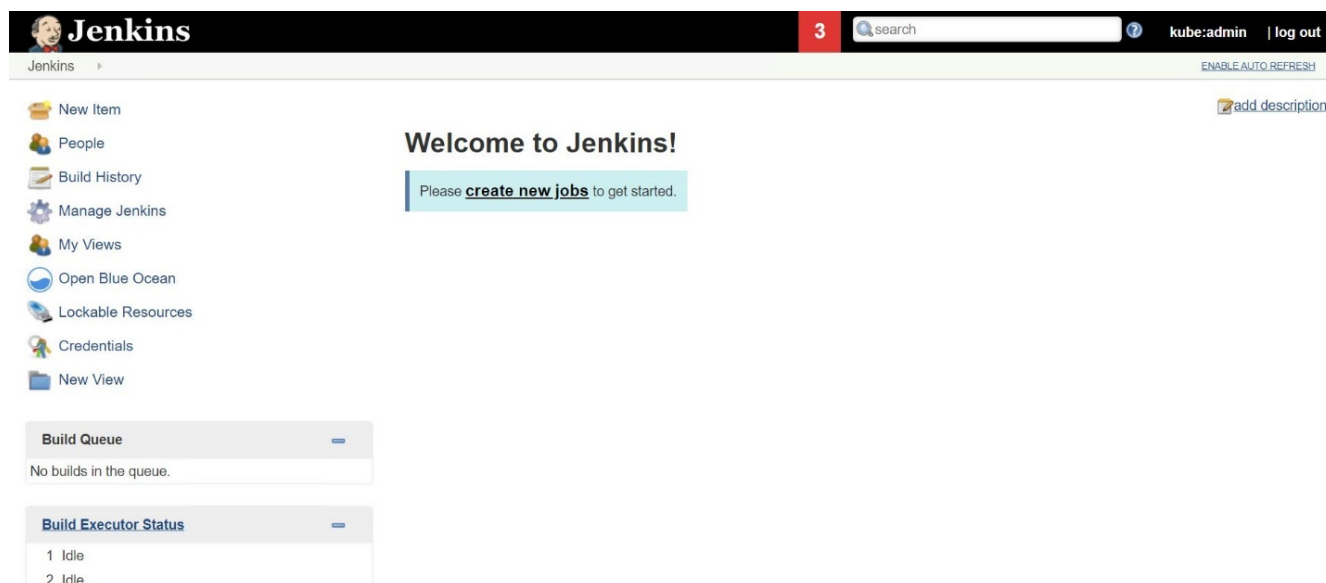
- ☒ **user:info**  
Read-only access to your user information (including username, identities, and group membership)
- ☒ **user:check-access**  
Read-only access to view your privileges (for example, "can I create builds?")

You will be redirected to <https://jenkins-jenkins.apps.rhv-ocp-cluster.cie.netapp.com/securityRealm/finishLogin>

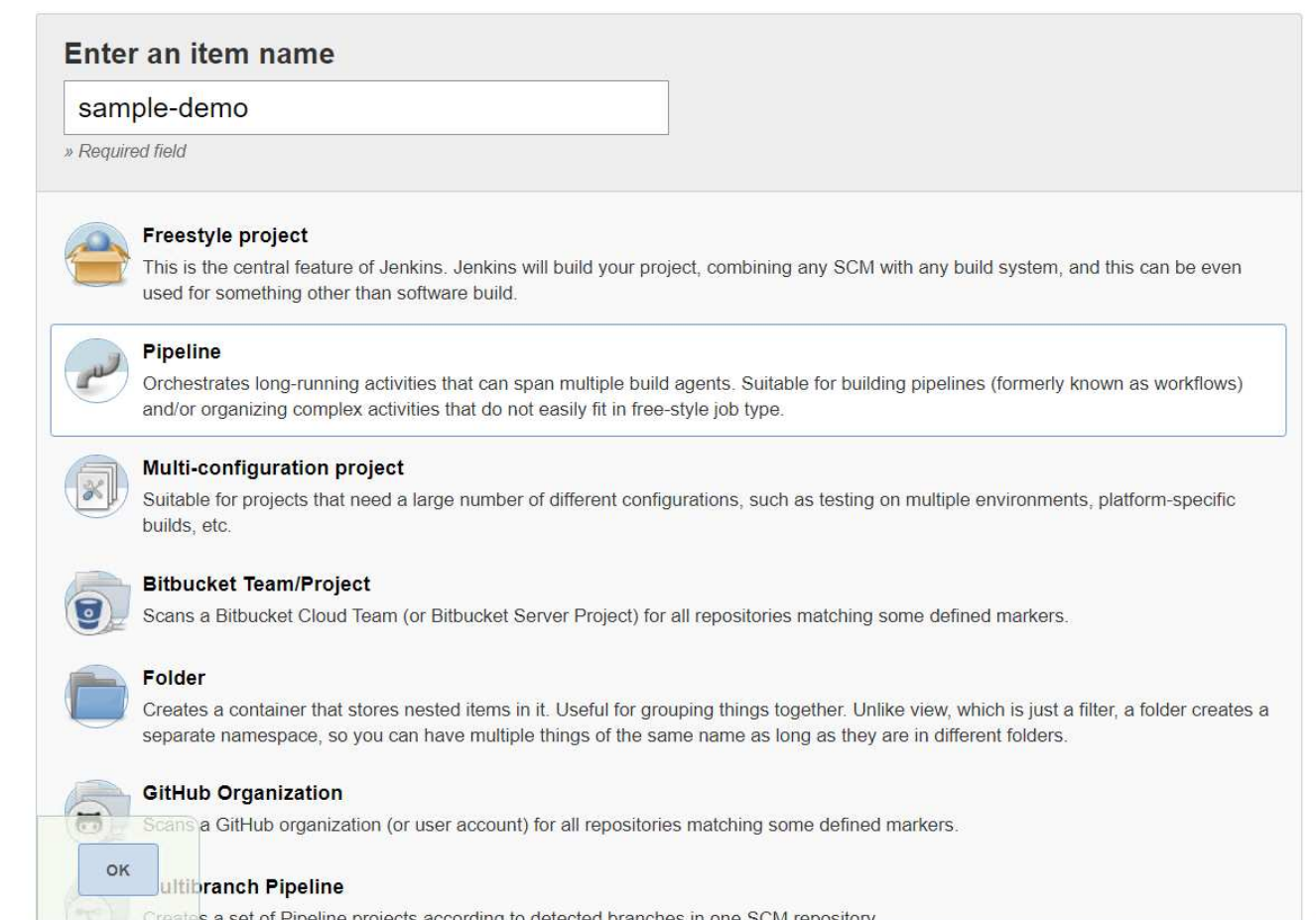
8. 隨即顯示Jenkins歡迎頁面。因為我們使用的是墨文建置、所以請先完成墨文安裝。瀏覽至「Manage Jenkins（管理Jenkins）」>「Global Tool Configuration（全域工具組態）」、然後按一下「Men（新增）」子標題中的「Add Maven（新增輸入您選擇的名稱、並確定已選取「自動安裝」選項。按一下儲存。



9. 您現在可以建立管道來示範CI/CD工作流程。在首頁上、按一下左側功能表中的「Create New Jobs（建立新工作）」或「New item（新項目）」。



10. 在「Create item（建立項目）」頁面上、輸入您選擇的名稱、選取「Pipeline（管道）」、然後按一下「OK（確定）」。



11. 選取Pipeline（管道）索引標籤。從「試用範例管道」下拉式功能表中、選取「Github + Maven」。程式碼會自動填入。按一下儲存。

General Build Triggers Advanced Project Options **Pipeline**

Advanced...

### Pipeline

Definition Pipeline script

Script

```
1 node {
2   def mvnHome
3   stage('Preparation') { // for display purposes
4     // Get some code from a GitHub repository
5     git 'https://github.com/jglick/simple-maven-project-with-tests.git'
6     // Get the Maven tool.
7     // ** NOTE: This 'M3' Maven tool must be configured
8     // ** in the global configuration.
9     mvnHome = tool 'M3'
10  }
11  stage('Build') {
12    // Run the maven build
13    withEnv(["MVN_HOME=$mvnHome"]) {
14      if (isUnix()) {
15        sh "$MVN_HOME/bin/mvn" -Dmaven.test.failure.ignore clean package
16      } else {
17        bat("/%MVN_HOME%\bin\mvn" -Dmaven.test.failure.ignore clean package/)
18      }
19    }
20  }
21 }
```

GitHub + Maven

☒ Use Groovy Sandbox

[Pipeline Syntax](#)

Save Apply

12. 按一下「立即建置」、即可在準備、建置和測試階段觸發開發。完成整個建置程序並顯示建置結果可能需要幾分鐘的時間。

Jenkins

Jenkins

sample-demo

Back to Dashboard

Status

Changes

Build Now

Delete Pipeline

Configure

Full Stage View

Open Blue Ocean

Rename

Pipeline Syntax

Build History

find

X

#1

May 27, 2020 3:53 PM

Atom feed for all

Atom feed for failures

Pipeline sample-demo

Last Successful Artifacts

simple-maven-project-with-tests-1.0-SNAPSHOT.jar

1.71 KB

view

Recent Changes

Stage View

Average stage times:

(Average full run time: ~7s)

#1

May 27

No Changes

08:53

Preparation	Build	Results
2s	4s	69ms
2s	4s	69ms

Latest Test Result (no failures)

Permalinks

- Last build (#1), 1 min 23 sec ago
- Last stable build (#1), 1 min 23 sec ago
- Last successful build (#1), 1 min 23 sec ago
- Last completed build (#1), 1 min 23 sec ago

- 只要有任何程式碼變更、就能重新建置管線、修補新版軟體、實現持續整合與持續交付。按一下「近期變更」以追蹤先前版本的變更。

10

Jenkins

sample-demo

Back to Dashboard

Status

Changes

Build Now

Delete Pipeline

Configure

Full Stage View

Open Blue Ocean

Rename

Pipeline Syntax

Build History

find

X

#2

May 27, 2020 3:56 PM

#1

May 27, 2020 3:53 PM

Atom feed for all

Atom feed for failures

Pipeline sample-demo

Last Successful Artifacts

simple-maven-project-with-tests-1.0-SNAPSHOT.jar

1.71 KB

view

Recent Changes

Stage View

Average stage times:

(Average full run time: ~6s)

#2

May 27 08:56

No Changes

#1

May 27 08:53

No Changes

Preparation	Build	Results
2s	4s	86ms
1s	4s	104ms
2s	4s	69ms

Latest Test Result (no failures)

Permalinks

- Last build (#2). 19 sec ago
- Last stable build (#2). 19 sec ago
- Last successful build (#2). 19 sec ago
- Last completed build (#2). 19 sec ago

# 使用NetApp ONTAP 功能在Red Hat OpenShift上設定多租戶共享

## 使用NetApp在Red Hat OpenShift上設定多租戶共享

許多在容器上執行多個應用程式或工作負載的組織、傾向於針對每個應用程式或工作負載部署一個Red Hat OpenShift叢集。這可讓他們針對應用程式或工作負載實作嚴格的隔離、最佳化效能並減少安全性弱點。不過、為每個應用程式部署個別的Red Hat OpenShift叢集、會產生自己的問題集。它會增加營運成本、必須自行監控及管理每個叢集、因為不同應用程式的專屬資源而增加成本、並阻礙有效的擴充性。

若要克服這些問題、您可以考慮在單一Red Hat OpenShift叢集中執行所有應用程式或工作負載。但是在這樣的架構中、資源隔離和應用程式安全性弱點是其中一項重大挑戰。某个工作負載中的任何安全弱點都可能自然延伸到另一個工作負載、進而增加影響區域。此外、任何應用程式突然無法控制的資源使用率、都可能影響另一個應用程式的效能、因為預設不會有資源配置原則。

因此、企業組織希望能在這兩個領域中找到最佳的解決方案、例如允許他們在單一叢集中執行所有工作負載、同時為每個工作負載提供專屬叢集的優點。

其中一個有效的解決方案是在Red Hat OpenShift上設定多租戶共享。多租戶共享是一種架構、可讓多個租戶在同一個叢集上共存、並適當隔離資源、安全性等。在這種情況下、租戶可視為叢集資源的子集、而這些資源已設

定為供特定使用者群組專用。在Red Hat OpenShift叢集上設定多租戶共享可提供下列優點：

- 允許共用叢集資源、進而降低資本支出和營運成本
- 降低營運與管理成本
- 保護工作負載免於安全漏洞的交叉污染
- 保護工作負載、避免資源爭用造成非預期的效能降級

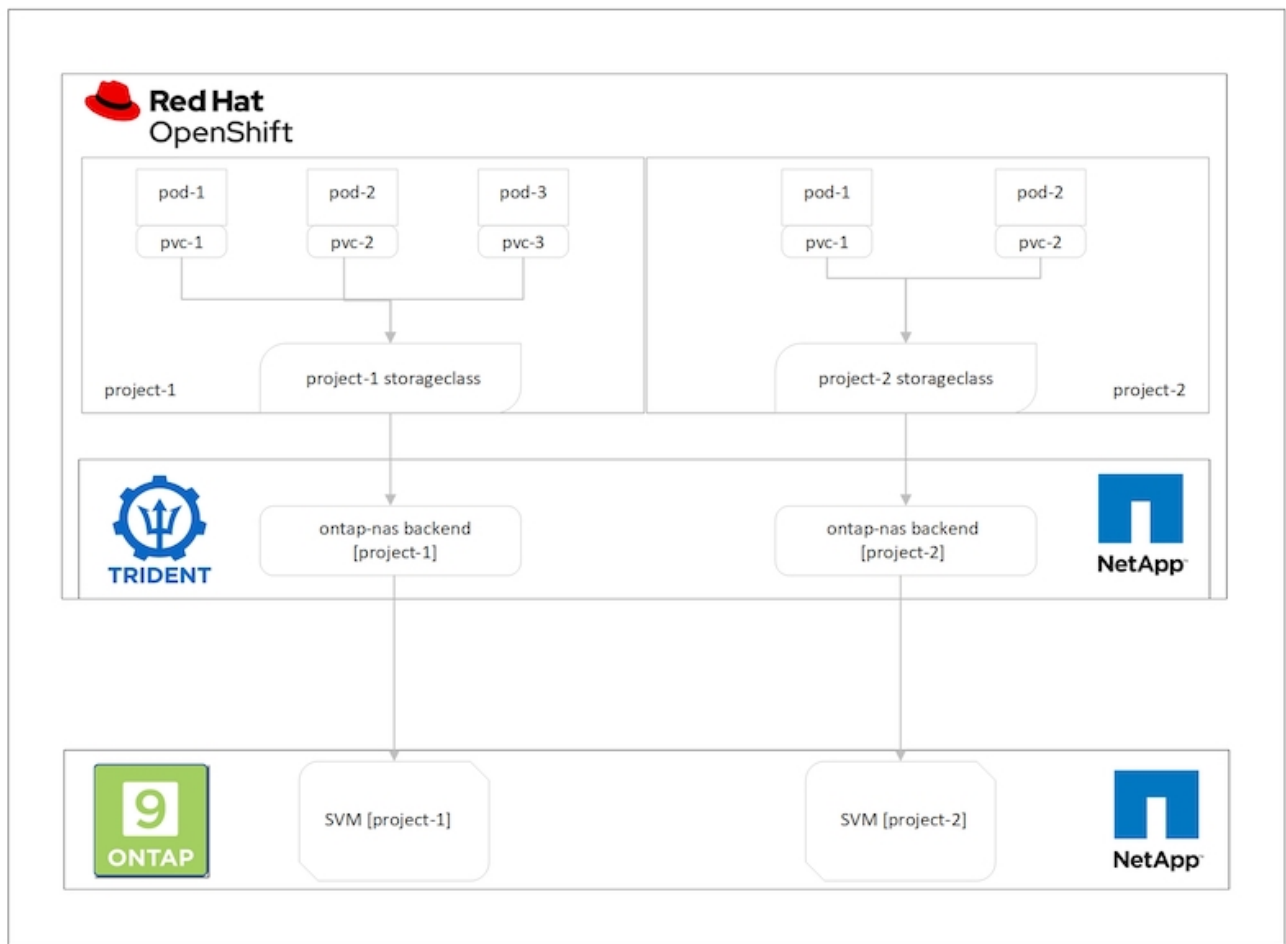
對於完全實現的多租戶OpenShift叢集、必須針對屬於不同資源桶的叢集資源設定配額和限制：運算、儲存、網路、安全性等。雖然我們涵蓋本解決方案中所有資源桶的某些層面、我們將重點放在隔離和保護同一個Red Hat OpenShift叢集上多個工作負載所提供或使用的資料的最佳實務做法上、方法是在由Astra Trident以NetApp ONTAP S動地 配置的儲存資源上設定多租戶。

## 架構

雖然Red Hat OpenShift和Astra Trident以NetApp ONTAP 支援、但在預設情況下並未隔離不同的工作負載、但它們提供多種功能、可用來設定多租戶。為了更深入瞭ONTAP 解如何在採用NetApp®技術的Astra Trident 的Red Hat OpenShift叢集上設計多租戶解決方案、請讓我們參考一組需求範例、並概述相關的組態。

假設某組織在Red Hat OpenShift叢集上執行兩項工作負載、這是兩個不同團隊正在進行的兩個專案的一部分。這些工作負載的資料位於PVCS上、由Astra Trident在NetApp ONTAP 的NAS後端動態配置。組織必須針對這兩項工作負載設計多租戶解決方案、並隔離用於這些專案的資源、以確保安全性和效能得以維持、主要著重於為這些應用程式提供服務的資料。

下圖說明Red Hat OpenShift叢集上的多租戶解決方案、其中Astra Trident以NetApp ONTAP 效益為後盾。



## 技術需求

1. NetApp ONTAP 解決方案儲存叢集
2. Red Hat OpenShift叢集
3. Astra Trident

## Red Hat OpenShift–叢集資源

從Red Hat OpenShift叢集觀點來看、最重要的資源是專案。OpenShift專案可視為叢集資源、將整個OpenShift叢集分成多個虛擬叢集。因此、專案層級的隔離功能可提供設定多租戶的基礎。

接下來是在叢集中設定RBAC。最佳實務做法是讓所有開發人員在身分識別供應商（IDP）的單一使用者群組中、處理單一專案或工作負載。Red Hat OpenShift允許IDP整合和使用者群組同步、因此可將IDP中的使用者和群組匯入叢集。這有助於叢集管理員將專案專屬的叢集資源存取權、隔離給該專案的使用者群組、進而限制未獲授權存取任何叢集資源。若要深入瞭解IDP與Red Hat OpenShift的整合、請參閱文件 ["請按這裡"](#)。

## NetApp ONTAP

將共享儲存設備隔離為Red Hat OpenShift叢集的持續儲存提供者非常重要、因為如此可確保在儲存設備上為每個專案建立的磁碟區、在主機看來就像是在不同的儲存設備上建立的磁碟區一樣。為達成此目標、請在NetApp ONTAP 支援上建立數量不限的SVM（儲存虛擬機器）、如同專案或工作負載一樣多、並將每個SVM專用於工作負載。



## Astra Trident

在NetApp ONTAP 支援上建立不同專案的不同SVM之後、您必須將每個SVM對應到不同的Trident後端。Trident上的後端組態會將持續儲存設備分配給OpenShift叢集資源、而且需要將SVM的詳細資料對應至該資源。這至少應該是後端的傳輸協定驅動程式。您也可以選擇定義如何在儲存設備上配置磁碟區、以及設定磁碟區大小或集合體使用量等限制。如需有關Trident後端定義的詳細資料、請參閱 ["請按這裡"](#)。

## Red Hat OpenShift-儲存資源

設定Trident後端之後、下一步是設定StorageClass。設定多個後端儲存類別、讓每個儲存類別都能存取、只在一個後端上增加磁碟區。我們可以在定義儲存類別時、使用storagePools參數、將StorageClass對應至特定的Trident後端。您可以找到定義儲存類別的詳細資料 ["請按這裡"](#)。因此、StorageClass與Trident後端之間有一對一對應關係、可指向一個SVM。如此可確保透過指派給該專案的StorageClass進行的所有儲存設備宣告、均由專屬該專案的SVM提供服務。

由於儲存類別並非命名資源、我們如何確保另一個命名空間或專案中的Pod對某個專案的儲存類別提出的要求遭到拒絕？答案是使用資源配額。資源配額是控制每個專案資源總使用量的物件。它可以限制專案中物件可耗用的資源數量和總容量。使用資源配額幾乎可以限制專案的所有資源、而且有效率地使用資源、有助於組織降低因資源過度配置或過度使用而造成的成本與中斷。請參閱文件 ["請按這裡"](#) 以取得更多資訊。

在此使用案例中、我們需要限制特定專案中的Pod、使其無法從非專屬專案的儲存類別中申請儲存設備。為達成此目標、我們必須將「<storage-class-name>.storageclass.storage.k8s.io/永久性 磁碟區」設為0、以限制其他儲存類別的持續磁碟區宣告。此外、叢集管理員必須確保專案中的開發人員不應擁有修改資源配額的存取權。

## 組態

對於任何多租戶解決方案、任何使用者都無法存取超過所需的叢集資源。因此、要設定為多租戶組態一部分的一整組資源、會分為叢集管理、儲存管理員和開發人員、分別負責每個專案。

下表概述不同使用者要執行的不同工作：

角色	工作
叢集管理	為不同的應用程式或工作負載建立專案
	為儲存管理員建立Cluster角色 和角色繫結
	建立角色與角色繫結、讓開發人員指派特定專案的存取權
	[選用]設定專案以排程特定節點上的Pod
儲存設備管理	在NetApp ONTAP 上建立SVM
	建立Trident後端
	建立StorageClass
	建立儲存資源配額
開發人員	驗證存取權限、以便在指派的專案中建立或修補PVCS或Pod
	驗證存取權限、以在另一個專案中建立或修補PVCS或Pod
	驗證存取權限、以檢視或編輯專案、資源配額和儲存類別



## 組態

### 先決條件

- NetApp ONTAP 產品叢集
- Red Hat OpenShift叢集
- 叢集上安裝的Trident
- 安裝了tridentctl和occ工具並新增至\$path的管理工作站
- 管理員存取ONTAP 功能
- 叢集管理存取OpenShift叢集
- 叢集已與Identity Provider整合
- 身分識別供應商的設定可有效區分不同團隊中的使用者

### 組態：叢集管理工作

Red Hat OpenShift叢集管理會執行下列工作：

1. 以叢集管理的身分登入Red Hat OpenShift叢集。
2. 建立兩個對應於不同專案的專案。

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

3. 建立專案1的開發人員角色。

```
cat << EOF | oc create -f -
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  namespace: project-1
  name: developer-project-1
rules:
  - verbs:
    - '*'
    apiGroups:
    - apps
    - batch
    - autoscaling
    - extensions
    - networking.k8s.io
    - policy
    - apps.openshift.io
    - build.openshift.io
```

```

- image.openshift.io
- ingress.operator.openshift.io
- route.openshift.io
- snapshot.storage.k8s.io
- template.openshift.io
resources:
  - '*'
- verbs:
  - '*'
apiGroups:
  - ''
resources:
  - bindings
  - configmaps
  - endpoints
  - events
  - persistentvolumeclaims
  - pods
  - pods/log
  - pods/attach
  - podtemplates
  - replicationcontrollers
  - services
  - limitranges
  - namespaces
  - componentstatuses
  - nodes
- verbs:
  - '*'
apiGroups:
  - trident.netapp.io
resources:
  - trident.snapshots
EOF

```



本節提供的角色定義只是一個範例。開發人員角色必須根據終端使用者需求加以定義。

1. 同樣地、請為專案2建立開發人員角色。
2. 所有OpenShift和NetApp儲存資源通常由儲存管理員管理。儲存管理員的存取權由安裝Trident時所建立的Trident操作員角色控制。此外、儲存管理員也需要存取資源配額、才能控制儲存設備的使用方式。
3. 在叢集中的所有專案中建立管理資源配額的角色、以將其附加至儲存設備管理員。

```

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

```

4. 請確定叢集已與組織的身分識別提供者整合、而且使用者群組已與叢集群組同步。下列範例顯示身分識別提供者已與叢集整合、並與使用者群組同步。

```

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

```

1. 為儲存管理員設定Cluster勞力 綁定。

```

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

```



對於儲存管理員、必須綁定兩個角色：Trident運算子和資源配額。

1. 為開發人員建立角色連結、將開發人員專案1角色繫結至專案1中對應的群組（OCP專案-1）。

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

2. 同樣地、請為開發人員建立角色連結至專案2中對應的使用者群組的角色連結。

組態：儲存設備管理工作

儲存管理員必須設定下列資源：

1. 以admin身分登入NetApp ONTAP 解決方案叢集。
2. 瀏覽至Storage（儲存設備）> Storage VM（儲存設備VM）、然後按一下Add提供所需的詳細資料、建立兩個SVM、一個用於專案1、另一個用於專案2。也可建立vsadmin帳戶來管理SVM及其資源。

# Add Storage VM



STORAGE VM NAME

project-1-svm

## Access Protocol

☒ SMB/CIFS, NFS

iSCSI

☐ Enable SMB/CIFS

☒ Enable NFS

☒ Allow NFS client access

Add at least one rule to allow NFS clients to access volumes in this storage VM. [?](#)

EXPORT POLICY

Default

RULES

Rule Index	Clients	Access Protocols	Read-Only R...	Read/Wr
	10.61.181.0/24	Any	Any	Any

[+ Add](#)

DEFAULT LANGUAGE [?](#)

c.utf\_8

NETWORK INTERFACE

Use multiple network interfaces when client traffic is high.

K8s-Ontap-01

IP ADDRESS

10.61.181.224

SUBNET MASK

24

GATEWAY

[Add optional gateway](#)

BROADCAST DOMAIN

Default-4

1. 以儲存管理員身分登入Red Hat OpenShift叢集。
2. 建立專案1的後端、並將其對應至專案專用的SVM。NetApp建議使用SVM的vsadmin帳戶、將後端連線至SVM、而非ONTAP 使用該叢集管理員。

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



本例使用的是ONTAP-NAS驅動程式。根據使用案例建立後端時、請使用適當的驅動程式。



我們假設Trident安裝在Trident專案中。

1. 同樣地、請為專案2建立Trident後端、並將其對應至專案2專用的SVM。
2. 接下來、建立儲存類別。建立專案1的儲存類別、並設定storagePools參數、以使用從專屬後端到專案1的儲存資源池。

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

3. 同樣地、請為專案2建立儲存類別、並將其設定為使用從專屬後端到專案2的儲存資源池。
4. 建立資源配額、以限制專案1中的資源、要求儲存資源來自其他專案專用的儲存設備。

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

5. 同樣地、也可以建立資源配額、以限制專案2中的資源、要求儲存資源來自其他專案專用的儲存設備。

## 驗證

若要驗證先前步驟中設定的多租戶架構、請完成下列步驟：

驗證存取權、以在指派的專案中建立**PVCS**或**Pod**

1. 以專案1的開發人員OCP專案1使用者身分登入。
2. 檢查存取權限以建立新專案。

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

3. 在專案1中使用指派給專案1的storageclass建立一個PVC.

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



4. 檢查與室早相關的PV。

```
oc get pv
```

5. 驗證PV及其Volume是否是在專門用於NetApp ONTAP 上專案1的SVM中建立。

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

6. 在專案1中建立一個Pod、然後掛載上一步建立的永久虛擬儲存設備。

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

7. 檢查Pod是否正在執行、以及是否已掛載磁碟區。

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

驗證存取權限、以在另一個專案中建立**PVCS**或**Pod**、或使用其他專案專用的資源

1. 以專案1的開發人員OCP專案1使用者身分登入。
2. 使用指派給專案2的儲存裝置在專案1中建立一個PVC。

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

### 3. 在專案2中建立一個PVC.

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

### 4. 確保未建立PVCS「test-PVC-project - 1-sc-2」和「test-PVC-project - 2-sc-1」。

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

### 5. 在專案2中建立Pod。

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

驗證存取權限、以檢視及編輯專案、資源配額和儲存類別

1. 以專案1的開發人員OCP專案1使用者身分登入。
2. 檢查存取權限以建立新專案。

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

3. 驗證存取權限以檢視專案。

```
oc get ns
```

4. 檢查使用者是否可以在專案1中檢視或編輯資源配額。

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

5. 驗證使用者是否有權檢視儲存空間。

```
oc get sc
```

6. 檢查存取以描述儲存空間。
7. 驗證使用者的存取權、以編輯儲存空間。

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

## 擴充：新增更多專案

在多租戶組態中、新增含有儲存資源的專案需要額外的組態、以確保不違反多租戶共享。若要在多租戶叢集中新增更多專案、請完成下列步驟：

1. 以儲存管理員身分登入NetApp ONTAP 解決方案叢集。
2. 瀏覽至「儲存虛擬機器」、然後按一下「Add（新增）」。建立專案3專用的新SVM。也可建立vsadmin帳戶來管理SVM及其資源。

# Add Storage VM



STORAGE VM NAME

project-3-svm

## Access Protocol

☒ SMB/CIFS, NFS

iSCSI

☐ Enable SMB/CIFS

☒ Enable NFS

☒ Allow NFS client access

Add at least one rule to allow NFS clients to access volumes in this storage VM. [?](#)

EXPORT POLICY

Default

RULES

Rule Index	Clients	Access Protocols	Read-Only R...	Read/Wr
	10.61.181.0/24	Any	Any	Any

[+ Add](#)

DEFAULT LANGUAGE [?](#)

c.utf\_8

NETWORK INTERFACE

Use multiple network interfaces when client traffic is high.

K8s-Ontap-01

IP ADDRESS

10.61.181.228

SUBNET MASK

24

GATEWAY

[Add optional gateway](#)

BROADCAST DOMAIN

Default-4

1. 以叢集管理身分登入Red Hat OpenShift叢集。
2. 建立新專案。

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

3. 確認專案3的使用者群組是在IDP上建立、並與OpenShift叢集同步。

```
oc get groups
```

4. 建立專案3的開發人員角色。

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

```

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

```



本節提供的角色定義只是一個範例。開發人員角色必須根據終端使用者需求加以定義。

1. 在Project 3中為開發人員建立角色繫結、將開發人員專案3角色繫結至專案3中對應的群組（OCP專案3）。


```


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

```

2. 以儲存管理員身分登入Red Hat OpenShift叢集
3. 建立Trident後端、並將其對應至專案3專用的SVM。NetApp建議使用SVM的vsadmin帳戶、將後端連線至SVM、而非ONTAP 使用叢集管理員。

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

 本例使用的是ONTAP-NAS驅動程式。根據使用案例、使用適當的驅動程式來建立後端。

 我們假設Trident安裝在Trident專案中。

1. 建立專案3的儲存類別、並將其設定為使用從專案3專用後端的儲存資源池。

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

2. 建立資源配額、以限制專案3中的資源、要求儲存資源來自其他專案專用的儲存設備。



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

3. 修補其他專案中的資源配額、限制這些專案中的資源無法從專案3專用的儲存設備存取儲存設備。

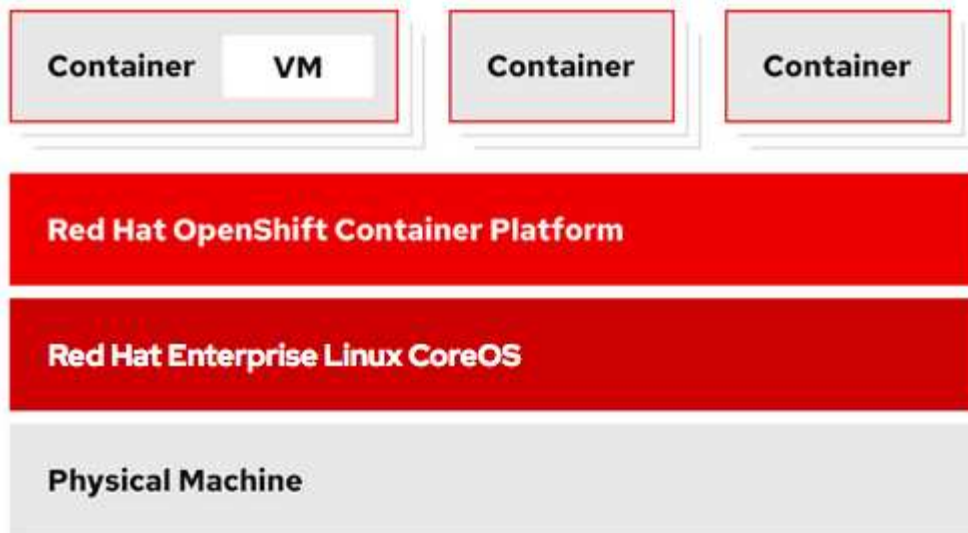
```
oc patch resourcequotas project-1-sc-rq -n project-1 --patch
'{"spec":{"hard":{"project-3-sc.storageclass.storage.k8s.io/persistentvolumeclaims": 0}}}'
oc patch resourcequotas project-2-sc-rq -n project-2 --patch
'{"spec":{"hard":{"project-3-sc.storageclass.storage.k8s.io/persistentvolumeclaims":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

## Red Hat OpenShift 虛擬化搭配 NetApp ONTAP 產品

### Red Hat OpenShift 虛擬化搭配 NetApp ONTAP 產品

根據特定的使用案例、容器和虛擬機器（VM）都能做為不同應用程式類型的最佳平台。因此、許多組織在容器上執行部分工作負載、而在VM上執行部分工作負載。這通常會讓組織面臨更多挑戰、因為必須管理不同的平台：VM的Hypervisor和應用程式的Container Orchestrator。

為了因應這項挑戰、Red Hat從OpenShift版本4.6開始推出OpenShift虛擬化（先前稱為Container Native Virtualization）。OpenShift虛擬化功能可讓您在相同的OpenShift Container Platform安裝上、同時執行及管理虛擬機器與容器、提供混合式管理功能、以便透過操作員自動化VM的部署與管理。除了在OpenShift中建立VM之外、Red Hat還支援從VMware vSphere、Red Hat虛擬化及Red Hat OpenStack平台部署中匯入VM。



OpenShift虛擬化也支援即時VM移轉、VM磁碟複製、VM快照等特定功能、並在由NetApp ONTAP 支援的情況下、由Astra Trident提供協助。本文件稍後將在各自的章節中討論每個工作流程的範例。

若要深入瞭解Red Hat OpenShift虛擬化、請參閱文件 ["請按這裡"](#)。

## 部署 OpenShift 虛擬化

部署Red Hat OpenShift虛擬化技術搭配NetApp ONTAP 功能

先決條件

- Red Hat OpenShift叢集（高於版本4.6）安裝在裸機基礎架構上、並具有RHCOOS工作節點
- OpenShift叢集必須透過安裝程式提供的基礎架構（IPI）進行安裝
- 部署機器健全狀況檢查以維護VM的HA
- NetApp ONTAP 的叢集
- 安裝在OpenShift叢集上的Astra Trident
- Trident後端在ONTAP 叢集上設定SVM
- OpenShift叢集上設定的StorageClass、其中Astra Trident為資源配置程式
- 叢集管理存取Red Hat OpenShift叢集
- 管理員存取NetApp ONTAP 解決方案叢集
- 安裝了tridentctl和occ工具並新增至\$path的管理工作站

由於OpenShift虛擬化是由安裝在OpenShift叢集上的操作員所管理、因此會對記憶體、CPU和儲存設備產生額外的負荷、因此在規劃叢集的硬體需求時、必須將這些負荷列入考量。請參閱文件 ["請按這裡"](#) 以取得更多詳細資料。

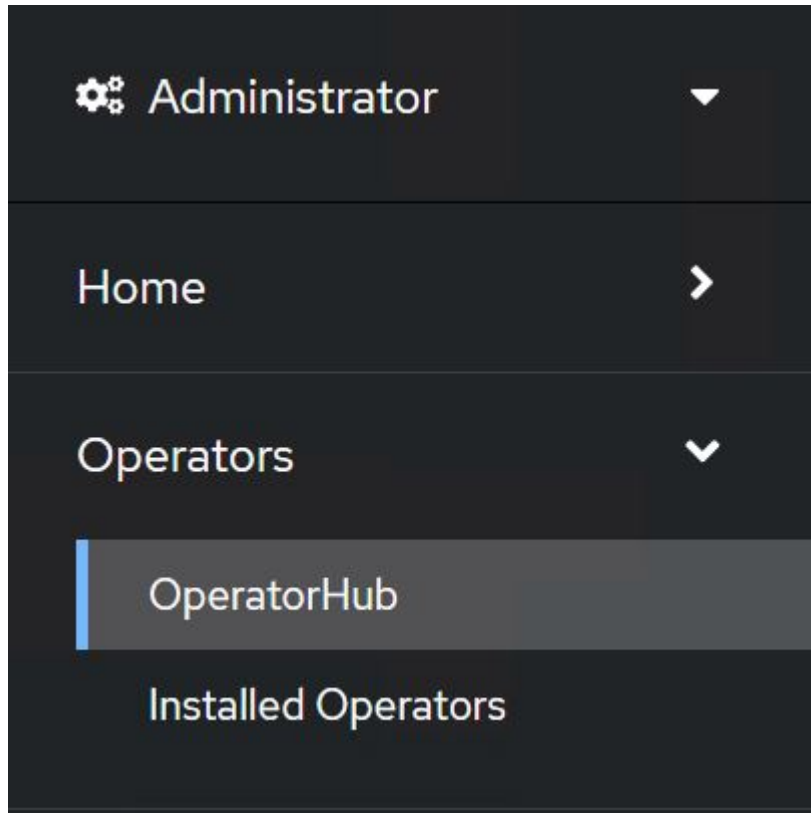
或者、您也可以設定節點放置規則、以指定OpenShift叢集節點的子集來裝載OpenShift虛擬化操作員、控制器和VM。若要設定OpenShift虛擬化的節點放置規則、請遵循文件 ["請按這裡"](#)。

對於支援OpenShift虛擬化的儲存設備、NetApp建議使用專屬StorageClass、從特定Trident後端要求儲存設備、然後再由專屬SVM提供支援。這可維持多租戶層級、以處理OpenShift叢集上VM型工作負載所需的資料。

## 部署Red Hat OpenShift虛擬化技術搭配NetApp ONTAP 功能

若要安裝OpenShift虛擬化、請完成下列步驟：

1. 以叢集管理存取權登入Red Hat OpenShift裸機叢集。
2. 從Perspective（透視）下拉列表中選擇Administrator（管理員
3. 瀏覽至「運算子」>「運算子中樞」、然後搜尋OpenShift虛擬化。



4. 選取OpenShift Virtualization動態磚、然後按一下Install（安裝）。



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Install Operator（安裝操作員）畫面上、保留所有預設參數、然後按一下Install（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



#### 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




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. 安裝完操作員之後、按一下「Create hyperconverged（建立超融合式）」





**OpenShift Virtualization**  
2.6.2 provided by Red Hat



## Installed operator – operand required

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

 **HyperConverged**  **Required**

Creates and maintains an OpenShift Virtualization Deployment

[Create HyperConverged](#)

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

8. 在Create hyperconverged（建立超融合式）畫面上、按一下Create（建立）、接受所有預設參數。此步驟會開始安裝OpenShift虛擬化。

35

**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.

- 在 openshift-cnv 命名空間中的所有 Pod 移至執行狀態、且 OpenShift 虛擬化運算子處於「成功」狀態之後、即可開始使用運算子。現在可以在 OpenShift 叢集上建立 VM。

Project: openshift-cnv ▾





## Installed Operators

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

Name ▾

Search by name...

/

Name ↑	Managed Namespaces ⌵	Status	Last updated	Provided APIs
<div><div></div><div><div>OpenShift Virtualization</div><div>2.6.2 provided by Red Hat</div></div></div>	<div><div></div><div>openshift-cnv</div></div>	<div><div></div><div>Succeeded</div><div>Up to date</div></div>	<div><div></div><div>May 18, 8:02 pm</div></div>	<div><div><a href="#">OpenShift Virtualization Deployment</a></div><div><a href="#">HostPathProvisioner deployment</a></div></div> <div>⋮</div>

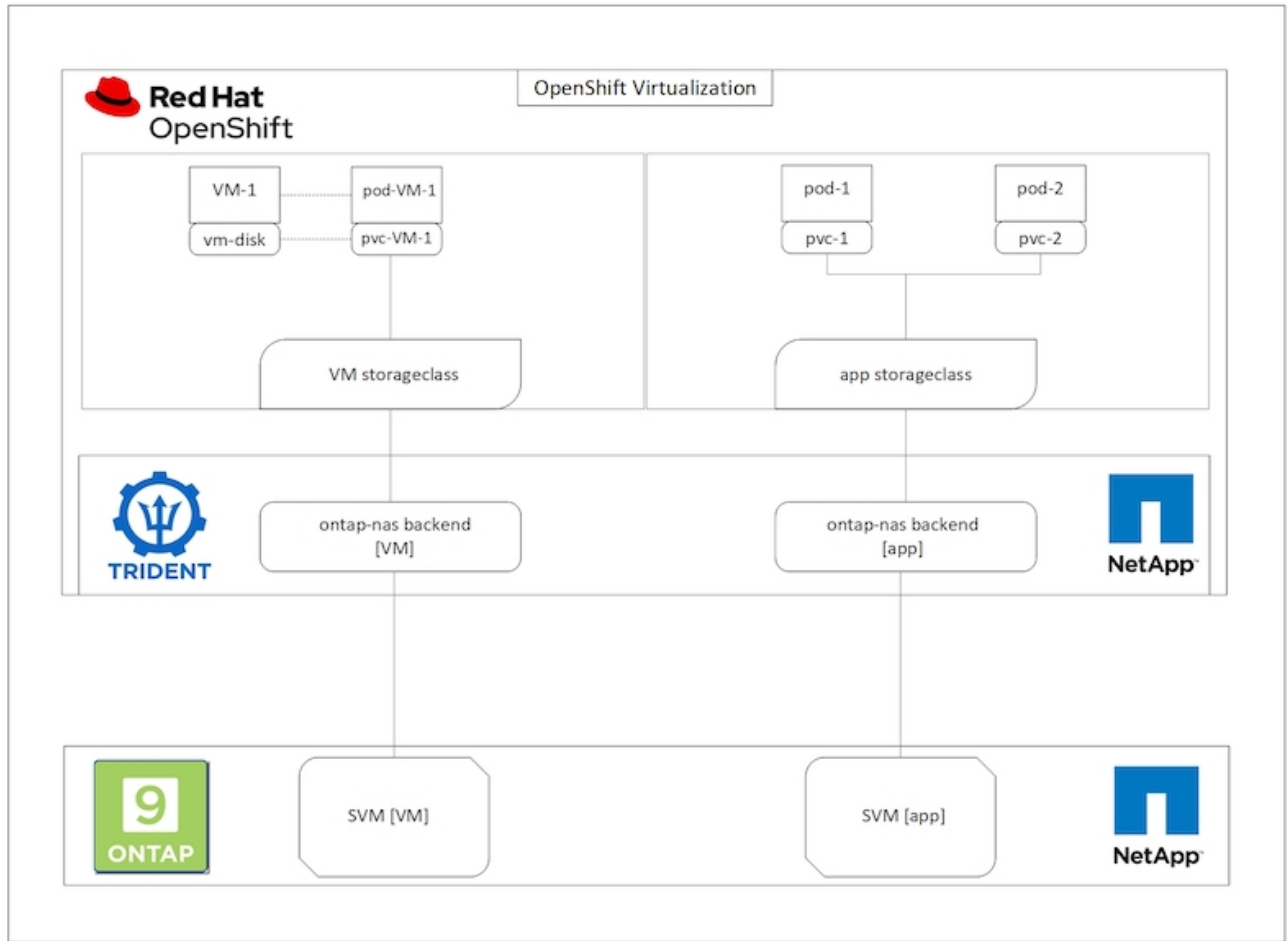
## 工作流程

工作流程：Red Hat OpenShift 虛擬化搭配 NetApp ONTAP 功能

## 建立 VM

VM 是有狀態的部署、需要磁碟區來裝載作業系統和資料。有了 CNV、因為 VM 是以 Pod 形式執行、所以 VM 有 NetApp ONTAP 透過 Trident 代管的 PV 作為後盾。這些磁碟區會附加為磁碟、並儲存整個檔案系統、包括 VM

的開機來源。



若要在OpenShift叢集上建立虛擬機器、請完成下列步驟：

1. 瀏覽至工作負載>虛擬化>虛擬機器、然後按一下建立>使用精靈。
2. 選取所需的作業系統、然後按「Next（下一步）」。
3. 如果選取的作業系統未設定開機來源、則必須加以設定。針對開機來源、選取是要從URL或登錄匯入OS映像、然後提供對應的詳細資料。展開「進階」、然後選取「Trident備份的StorageClass」。然後按「Next（下一步）」

## Boot source

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

### Boot source type \*

Import via URL (creates PVC) ▼

### Import URL \*

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

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

☒ Mount this as a CD-ROM boot source ?

### Persistent Volume Claim size \*

5

GiB ▼

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

### ▼ Advanced

#### Storage class \*

basic (default) ▼

#### Access mode \*

Single User (RWO) ▼

#### Volume mode \*

Filesystem ▼

4. 如果選取的作業系統已設定開機來源、則可以跳過上一個步驟。
5. 在「Review and Create（檢閱與建立）」窗格中、選取您要在其中建立VM的專案、並提供VM詳細資料。請確定已選取要複製的開機來源、並使用指派給所選作業系統的適當永久磁碟從CD-ROM開機。



- 1 Select template
- 2 Review and create

## Review and create

You are creating a virtual machine from the **Red Hat Enterprise Linux 8.0+** VM template.

Project \*

PR default

Virtual Machine Name \* ⓘ

rhel8-light-bat

Flavor \*

Small: 1 CPU | 2 GiB Memory

Storage

Workload profile ⓘ

40 GiB

server

Boot source

Clone and boot from CD-ROM

PVC rhel8

ⓘ A new disk has been added to support the CD-ROM boot source. Edit this disk by customizing the virtual machine.

▼ Disk details

rootdisk-install - Blank - 20GiB - virtio - default Storage class

☒ Start this virtual machine after creation

Create virtual machine

Customize virtual machine

Back

Cancel

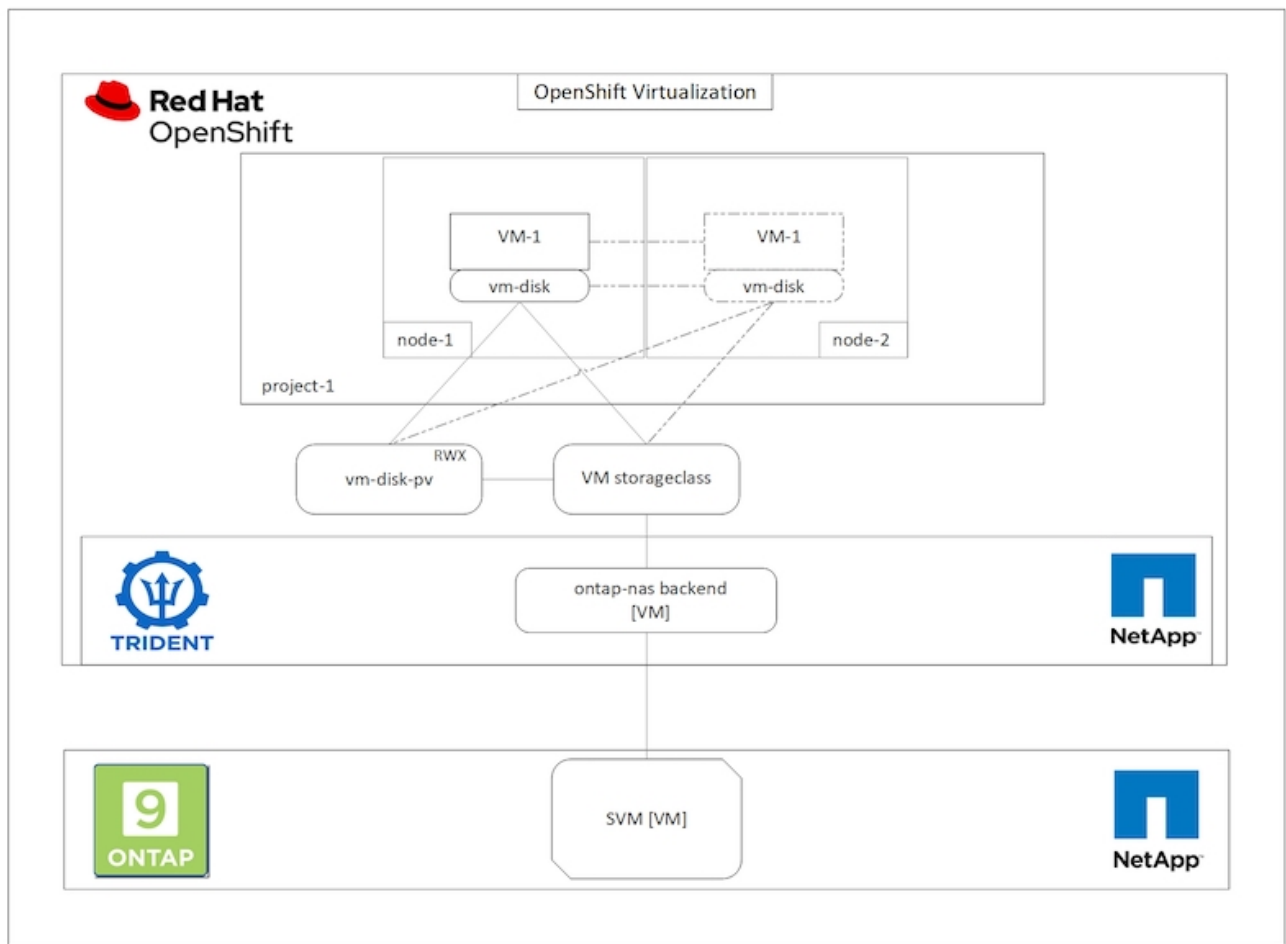
6. 如果您要自訂虛擬機器、請按一下「自訂虛擬機器」、然後修改所需的參數。
7. 按一下「Create Virtual Machine」（建立虛擬機器）以建立虛擬機器；這會使背景中的對應Pod旋轉。

當從URL或登錄設定範本或作業系統的開機來源時、會在「openshift-virtualization作業系統映像」專案中建立一個PVC、並將KVM來賓映像下載至PVC。您必須確定範本PVCS有足夠的資源配置空間、以容納對應作業系統的KVM來賓映像。這些PVCS隨後會以rootdisks的形式複製並附加至虛擬機器、並在任何專案中使用各自的範本建立。

工作流程：Red Hat OpenShift虛擬化搭配NetApp ONTAP 功能

## VM即時移轉

即時移轉是將VM執行個體從OpenShift叢集中的某個節點移轉到另一個節點的程序、不會造成停機。若要在OpenShift叢集中執行即時移轉、VM必須繫結至具有共用ReadWriteMany存取模式的PVCS。Astra Trident後端在啟用ONTAP NFS傳輸協定的NetApp支援叢集上設定SVM、可支援對PVCS進行共用的ReadWriteMany存取。因此、從採用NFS的SVM上的Trident配置的StorageClass所要求的具有PVCS的VM、可以在不中斷的情況下進行移轉。



若要建立連結至具有共用ReadWriteMany存取權之PVCS的VM：

1. 瀏覽至工作負載>虛擬化>虛擬機器、然後按一下建立>使用精靈。
2. 選取所需的作業系統、然後按「Next（下一步）」。讓我們假設所選的作業系統已經設定了開機來源。
3. 在「Review and Create（檢閱與建立）」窗格中、選取您要在其中建立VM的專案、並提供VM詳細資料。請確定已選取要複製的開機來源、並使用指派給所選作業系統的適當永久磁碟從CD-ROM開機。
4. 按一下[自訂虛擬機器]，然後按一下[儲存設備]
5. 按一下rootdisk旁的省略符號、並確定已選取使用Trident配置的儲存ageclasse。展開「進階」、然後選取「存取模式」的「共享存取（rwx）」。然後按一下「儲存」。

## Edit Disk

Type

Disk

Interface \*

virtio

Storage Class

basic (default)

▼ Advanced



Volume Mode

Filesystem

Volume Mode is set by Source PVC

Access Mode

Shared Access (RWX) - Not recommended for basic storage class

 **Access and Volume modes should follow storage feature matrix**  
[Learn more](#) 

Cancel

Save

6. 按一下「Review（檢閱）」並確認、然後按一下「Create Virtual Machine（建立虛  
若要將VM手動移轉至OpenShift叢集中的其他節點、請完成下列步驟。

1. 瀏覽至「工作負載」>「虛擬化」>「虛擬機器」。

2. 針對您要移轉的VM、按一下省略符號、然後按一下「移轉虛擬機器」。
3. 當訊息快顯以確認時、請按一下「移轉」。

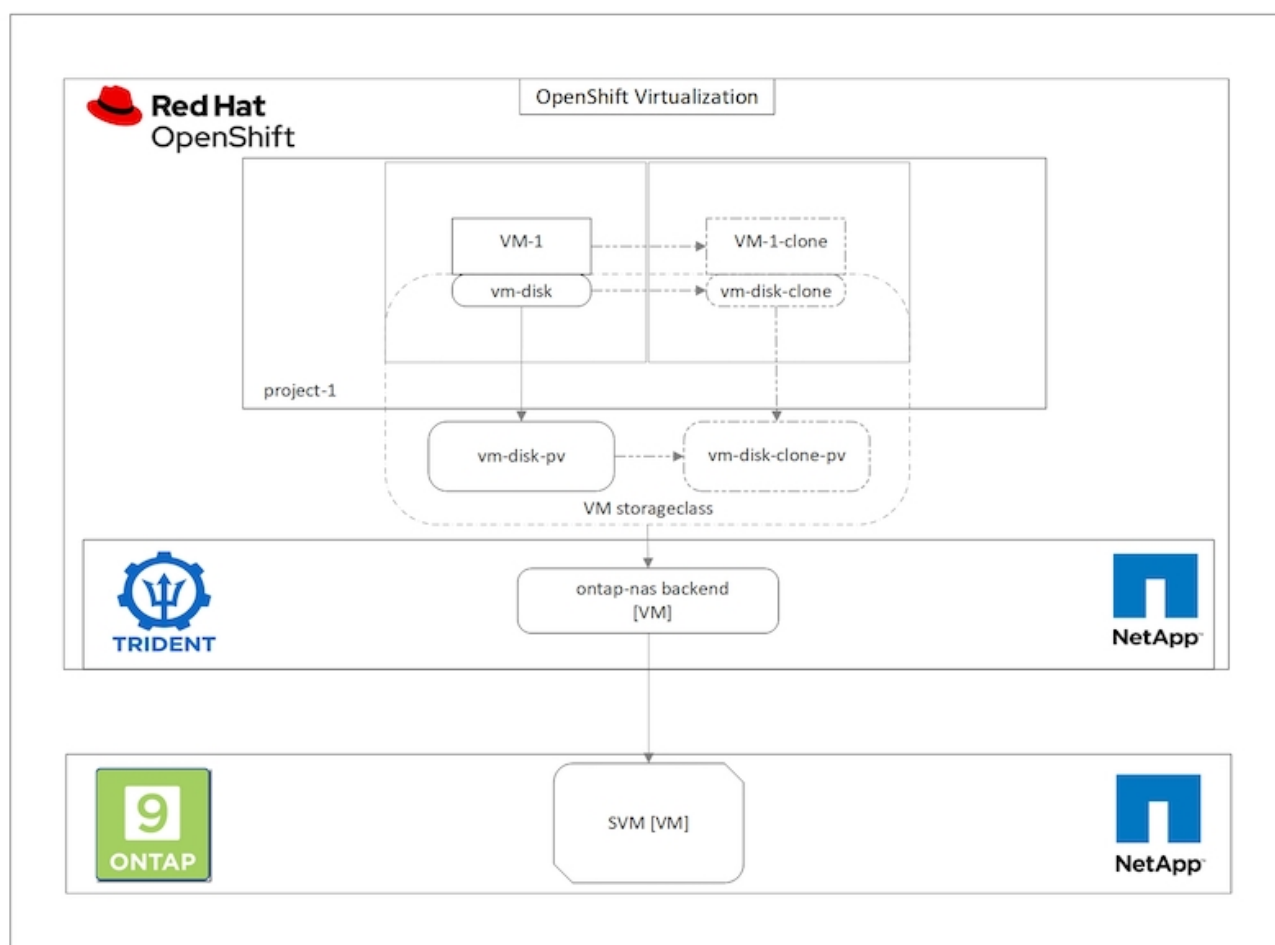


如果將設備策略設定為LiveMigrate、則當原始節點置於維護模式時、OpenShift叢集中的VM執行個體會自動移轉至其他節點。

工作流程：Red Hat OpenShift虛擬化搭配NetApp ONTAP 功能

## 虛擬機器複製

支援Astra Trident的Volume csi複製功能、即可在OpenShift中複製現有VM。透過複製現有的PV、可以使用現有的PVC作為資料來源來建立新的PVC。建立新的永久虛擬基礎架構之後、它會做為獨立實體運作、而且不會與來源永久虛擬基礎架構有任何連結或相依關係。



使用「csi Volume Cloning」時、必須考量下列限制：

1. 來源PVC和目的地PVC必須位於同一個專案中。
2. 同一儲存類別支援複製。
3. 只有在來源和目的地磁碟區使用相同的磁碟區模式設定時、才能執行複製；例如、區塊磁碟區只能複製到另一個區塊磁碟區。

OpenShift叢集中的VM可透過兩種方式複製：

1. 關閉來源VM
2. 讓來源VM保持運作


#### 關閉來源VM

關閉VM來複製現有VM是一項原生OpenShift功能、可透過Astra Trident的支援來實作。完成下列步驟以複製VM。

1. 瀏覽至「工作負載」>「虛擬化」>「虛擬機器」、然後按一下您要複製的虛擬機器旁的省略符號。
2. 按一下「Clone Virtual Machine（複製虛擬機器）」、並提供新VM的詳細資料。

## Clone Virtual Machine

Name *	<input type="text" value="rhel8-short-frog-clone"/>
Description	<div></div>
Namespace *	<div>default ▼</div>
	<input checked="" type="checkbox"/> Start virtual machine on clone
Configuration	<div><div>Operating System</div><div>Red Hat Enterprise Linux 8.0 or higher</div><div>Flavor</div><div>Small: 1 CPU   2 GiB Memory</div><div>Workload Profile</div><div>server</div><div>NICs</div><div>default - virtio</div><div>Disks</div><div>cloudinitdisk - cloud-init disk</div><div>rootdisk - 20Gi - basic</div></div>

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

Cancel

Clone Virtual Machine

- 按一下「Clone Virtual Machine（複製虛擬機器）」；這會關閉來源VM並開始建立複製VM。
- 完成此步驟之後、您可以存取並驗證複製的VM內容。

讓來源VM保持運作

也可以複製現有VM、方法是複製來源VM的現有PVC,然後使用複製的PVC,建立新VM。此方法不需要關閉來源VM。完成下列步驟、即可在不關閉VM的情況下複製VM。

- 1. 瀏覽至「Storage（儲存設備）」>「PeristentVolume Claims（永久磁碟區宣告）」、然後按一下附加至來源VM的永久磁碟旁的省略號。
- 2. 按一下Clone PVC（複製PVC）、並提供新PVC的詳細資料。

Clone

Name \*

rhel8-short-frog-rootdisk-28dvb-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. 然後按一下Clone（複製）這會為新VM建立一個永久虛擬機器。
- 4. 瀏覽至「工作負載」>「虛擬化」>「虛擬機器」、然後按一下「建立」>「使用Yaml」
- 5. 在SPEC >範本> SPEC > Volume區段中、附加複製的PVC而非容器磁碟。請根據您的需求、提供新VM的所有其他詳細資料。

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

6. 按一下「Create（建立）」以建立新的VM。
7. 成功建立VM之後、請存取並確認新VM是來源VM的複本。

工作流程：Red Hat OpenShift 虛擬化搭配 NetApp ONTAP 功能

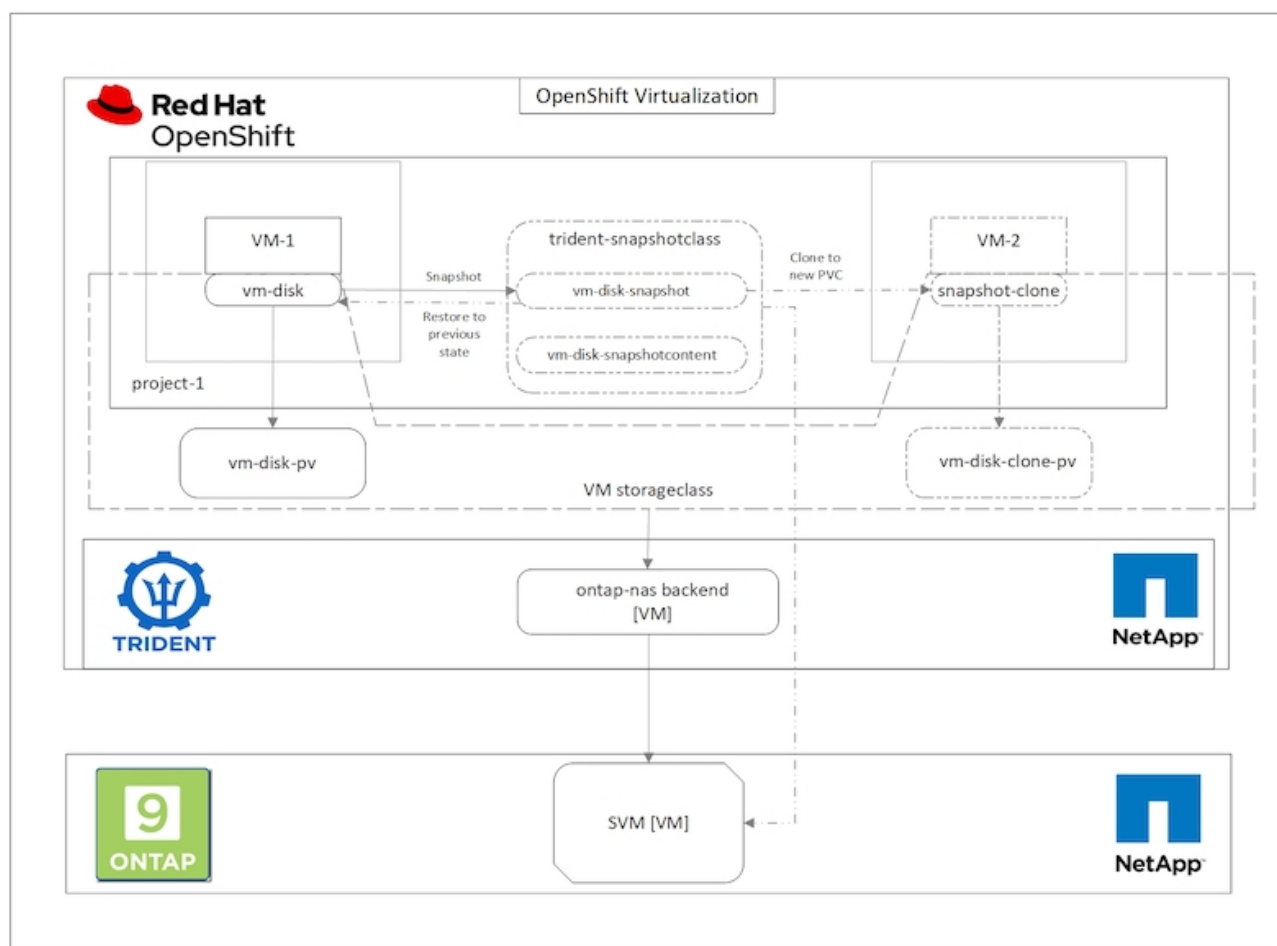
### 從Snapshot建立VM

有了Astra Trident和Red Hat OpenShift、使用者就能在IT資源配置的儲存類別上、對持續磁碟區進行快照。有了這項功能、使用者可以取得磁碟區的時間點複本、然後使用它來建立新的磁碟區、或將相同的磁碟區還原回先前的狀態。這可啟用或支援各種使用案例、從復原到複製到資料還原。

對於OpenShift中的Snapshot作業、必須定義Volume SnapshotClass、Volume Snapshot和Volume SnapshotContent等資源。

- Volume SnapshotContent是從叢集中的磁碟區擷取的實際快照。它是整個叢集的資源、類似於儲存的PersistentVolume。
- Volume Snapshot是建立Volume快照的要求。這類似於PersistentVolume Claim。
- Volume SnapshotClass可讓管理員為Volume Snapshot指定不同的屬性。它可讓您針對從相同磁碟區擷取的不同快照、擁有不同的屬性。





若要建立VM的Snapshot、請完成下列步驟：

1. 建立Volume SnapshotClass、然後使用該類別建立Volume Snapshot。瀏覽至「Storage（儲存設備）」>「Volume SnapshotClass（Volume SnapshotClass）」、然後按一下「Create Volume SnapshotClass」。
2. 輸入Snapshot Class的名稱、輸入驅動程式的csi.trident.netapp.io、然後按一下「Create（建立）」。

```
1  apiVersion: snapshot.storage.k8s.io/v1
2  kind: VolumeSnapshotClass
3  metadata:
4    name: trident-snapshot-class
5  driver: csi.trident.netapp.io
6  deletionPolicy: Delete
7
```

[Create](#)[Cancel](#)[Download](#)

3. 識別附加至來源VM的PVC、然後建立該PVC的Snapshot。瀏覽至「儲存> Volume Snapshots」、然後按一下「Create Volume Snapshots (建立Volume Snapshot)」。
4. 選取您要建立Snapshot的永久虛擬磁碟、輸入Snapshot名稱或接受預設值、然後選取適當的Volume SnapshotClass。然後按一下「建立」。

## 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](#)

5. 這會在該時間點建立永久虛擬資料快照。

## 從快照建立新的VM

1. 首先、將Snapshot還原成新的PVC。瀏覽至「Storage（儲存設備）」>「Volume Snapshots（Volume Snapshot）」、按一下您要還原的Snapshot旁邊的省略符號、然後按一下「Restore as new PVC（還原為新的PVC）」。
2. 輸入新的PVC詳細資料、然後按一下「還原」。這會產生新的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 \*

SC 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

NS default

Status

✅ Ready

API version

snapshot.storage.k8s.io/v1

Size

20 GiB

3. 接下來、從這個永久虛擬機器建立新的虛擬機器。瀏覽至「工作負載」>「虛擬化」>「虛擬機器」、然後按一下「建立」>「使用Yaml」

- 在SPEC > 範本> SPEC > Volume區段中、指定從Snapshot而非從Container磁碟建立的新永久虛擬磁碟。請根據您的需求、提供新VM的所有其他詳細資料。

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

- 按一下「Create（建立）」以建立新的VM。
- 成功建立虛擬機器之後、請存取並確認新虛擬機器的狀態與虛擬機器的狀態相同、而在建立快照時、虛擬機器的永久虛擬機器是用來建立快照的。

工作流程：Red Hat OpenShift 虛擬化搭配 NetApp ONTAP 功能

使用虛擬化移轉工具套件將 VM 從 VMware 移轉至 OpenShift 虛擬化

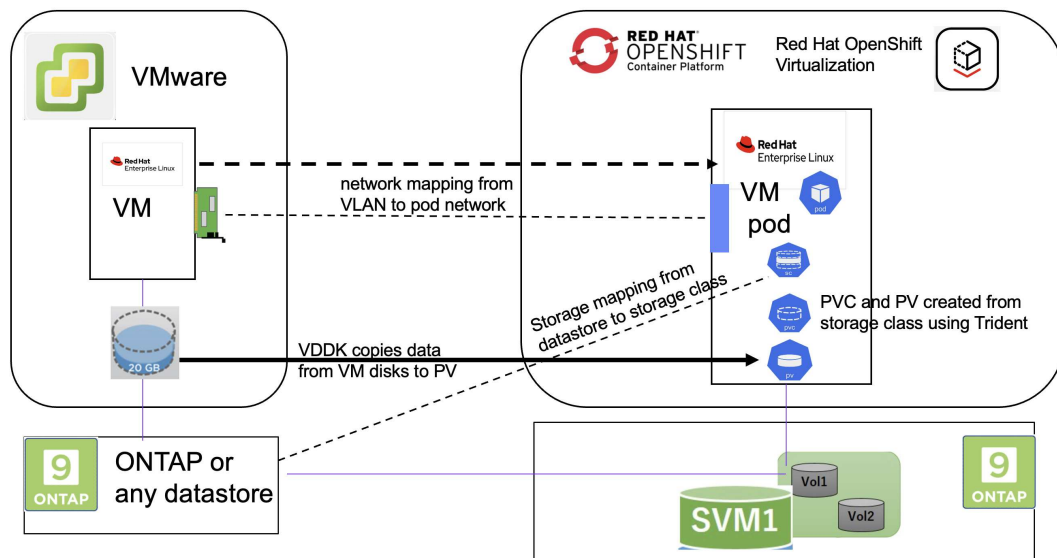
在本節中、我們將瞭解如何使用移轉工具套件來將虛擬機器從 VMware 移轉至 OpenShift Container 平台上執行的 OpenShift 虛擬化、並使用 Astra Trident 與 NetApp ONTAP 儲存設備整合。

以下影片展示 RHEL VM 從 VMware 移轉至 OpenShift 虛擬化的示範、使用 ONTAP SAN 進行持續儲存。

使用 Red Hat MTV 將 VM 移轉至使用 NetApp ONTAP 儲存設備的 OpenShift 虛擬化

下圖顯示虛擬機器從 VMware 移轉至 Red Hat OpenShift 虛擬化的高階檢視。

## Migration of VM from VMware to OpenShift Virtualization



範例移轉的先決條件

## 關於 VMware

- 已安裝使用 RHEL 9.3 的 RHEL 9 VM 、並搭配下列組態：
  - CPU ： 2 、記憶體： 20 GB 、硬碟： 20 GB
  - 使用者認證： root 使用者和管理員使用者認證
- VM 準備就緒後、即安裝 PostgreSQL 伺服器。
  - PostgreSQL 伺服器已啟動、並可在開機時啟動

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

- 新增 2 個資料庫、1 個資料表和 1 個資料列。請參閱 ["請按這裡"](#) 如需在 RHEL 上安裝 PostgreSQL 伺服器及建立資料庫和表格項目的指示。



請確定您啟動 PostgreSQL 伺服器、並讓服務在開機時啟動。

## 在 OpenShift 叢集上

下列安裝已在安裝 MTV 之前完成：

- OpenShift 叢集 4.13.34
- ["Astra Trident 23.10"](#)
- 叢集節點上啟用 iSCSI 的多重路徑（適用於 ONTAP - SAN 儲存類別）。請參閱提供的 yaml 、以建立在叢集中每個節點上啟用 iSCSI 的精靈集。
- Trident 後端和儲存類別、適用於使用 iSCSI 的 ONTAP SAN 。請參閱提供的 yaml 檔案、瞭解 Trident 後端和儲存類別。
- ["OpenShift虛擬化"](#)

若要在 OpenShift 叢集節點上安裝 iSCSI 和多重路徑、請使用以下提供的 yaml 檔案準備 **iSCSI** 的叢集節點

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

```

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

```

使用下列 yaml 檔案建立 Trident 後端組態、以使用 ONTAP SAN 儲存設備

## iSCSI 的 Trident 後端

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

使用下列 yaml 檔案建立 Trident 儲存類別組態、以使用 ONTAP SAN 儲存設備  
iSCSI 的 Trident 儲存等級

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

### \* 安裝 MTV\*

現在您可以安裝移轉工具套件（虛擬化）（MTV）。請參閱所提供的指示 ["請按這裡"](#) 取得安裝的說明。

移轉工具套件虛擬化（MTV）使用者介面已整合至 OpenShift 網路主控台。  
您可以參閱 ["請按這裡"](#) 開始使用使用者介面執行各種工作。

- 建立來源供應商 \*\*

為了將 RHEL VM 從 VMware 移轉至 OpenShift 虛擬化、您必須先建立 VMware 的來源供應商。請參閱說明 "[請按這裡](#)" 以建立來源供應商。

您需要下列項目來建立 VMware 來源供應商：

- vCenter URL
- vCenter 認證
- vCenter 伺服器指紋
- 儲存庫中的 VDDK 映像

建立範例來源供應商：

Select provider type \*

vm vSphere

Provider resource name \*

vmware-source

Unique Kubernetes resource name identifier

URL \*

URL of the vCenter SDK endpoint. Ensure the URL includes the "/sdk" path. For example: https://vCenter-host-example.com/sdk

VDDK init image

docker.repo.eng.netapp.com/banum/vddk:801

VDDK container image of the provider, when left empty some functionality will not be available

Username \*

administrator@vsphere.local

vSphere REST API user name.

Password \*

\*\*\*\*\*

vSphere REST API password credentials.

SSHA-1 fingerprint \*

The provider currently requires the SHA-1 fingerprint of the vCenter Server's TLS certificate in all circumstances. vSphere calls this the server's thumbprint.

Skip certificate validation

☒





虛擬化移轉工具套件（MTV）使用 VMware 虛擬磁碟開發套件（VDDK） SDK 來加速從 VMware vSphere 傳輸虛擬磁碟。因此、強烈建議您建立 VDDK 映像（雖然是選用的）。若要使用此功能、請下載 VMware 虛擬磁碟開發套件（VDDK）、建置 VDDK 映像、然後將 VDDK 映像推入映像登錄。

請遵循所提供的指示 ["請按這裡"](#) 建立 VDDK 映像、並將其推送至可從 OpenShift 叢集存取的登錄。

- 建立目的地供應商 \*\*

當 OpenShift 虛擬化供應商是來源供應商時、主機叢集會自動新增。

- 建立移轉計畫 \*\*

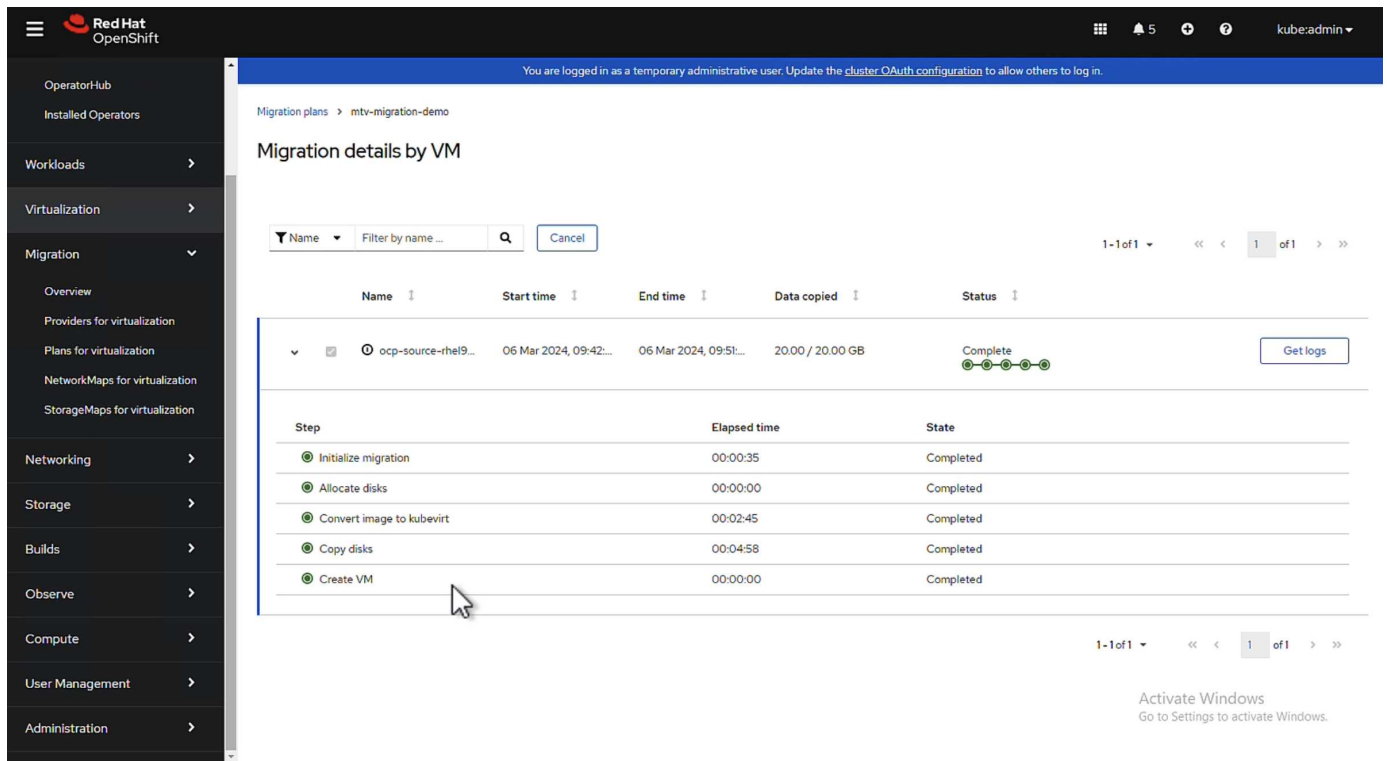
請遵循所提供的指示 ["請按這裡"](#) 以建立移轉計畫。

建立計畫時、如果尚未建立、則需要建立下列項目：

- 用於將來源網路對應至目標網路的網路對應。
- 將來源資料存放區對應至目標儲存類別的儲存對應。您可以選擇 ONTAP SAN 儲存類別。一旦建立移轉計畫、計畫的狀態應該會顯示 \* 就緒 \* 、您現在應該可以 \* 開始 \* 計畫。

Name	Source	Target	VMs	Status	Description
PL mtv-migration-demo cold	PR vmware	PR host	1	Ready	Plan for migrating VM to OpenShift Virt...
PL vmware-osv-migration cold	PR vmware2	PR host	1	Succeeded 1 of 1 VMs migrated	Migrating RHEL 9 vm to OpenShift Virt...
PL vmware-osv-migration-plan1 cold	PR vmware2	PR host	1	Succeeded 1 of 1 VMs migrated	
PL vmware-osv-migration-plan2 cold	PR vmware2	PR host	1	Succeeded 1 of 1 VMs migrated	migrating RHEL 9 vm using ONTAP NFS...

按一下 \* 「開始」 \* 將會執行一系列步驟、以完成虛擬機器的移轉。



完成所有步驟後、您可以按一下左側導覽功能表 \* 虛擬化 \* 下的 \* 虛擬機器 \* 來查看移轉的虛擬機器。提供存取虛擬機器的指示 "[請按這裡](#)"。

您可以登入虛擬機器並驗證 postgresql 資料庫的內容。資料表中的資料庫、資料表和項目應與在來源 VM 上建立的項目相同。

## 採用NetApp的Red Hat OpenShift上的Kubernetes進階叢集管理

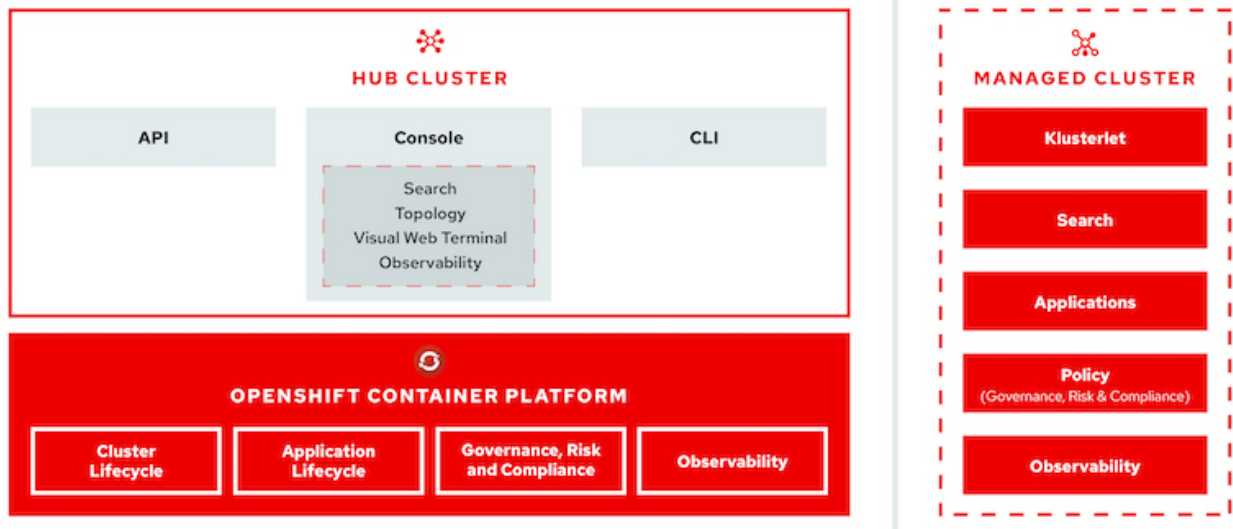
### 適用於Kubernetes的進階叢集管理：採用NetApp的Red Hat OpenShift

隨著容器化應用程式從開發移轉至正式作業、許多組織需要多個Red Hat OpenShift叢集來支援該應用程式的測試與部署。因此、組織通常會在OpenShift叢集上裝載多個應用程式或工作負載。因此、每個組織最終都必須管理一組叢集、因此OpenShift系統管理員必須面對新增的挑戰、即在橫跨多個內部部署資料中心和公有雲的各種環境中、管理及維護多個叢集。為了因應這些挑戰、Red Hat推出適用於Kubernetes的進階叢集管理。

適用於Kubernetes的Red Hat Advanced Cluster Management可讓您執行下列工作：

1. 跨資料中心和公有雲建立、匯入及管理多個叢集
2. 從單一主控台部署及管理多個叢集上的應用程式或工作負載
3. 監控及分析不同叢集資源的健全狀況與狀態
4. 監控並強制執行多個叢集的安全法規遵循

Red Hat Advanced Cluster Management for Kubernetes是以附加元件安裝至Red Hat OpenShift叢集的方式、它會將此叢集當作中央控制器來執行所有作業。此叢集稱為集線器叢集、會公開使用者連線至「進階叢集管理」的管理層面。透過進階叢集管理主控台匯入或建立的所有其他OpenShift叢集、均由中樞叢集管理、稱為託管叢集。它會在託管叢集上安裝名為Klusterlet的代理程式、將其連線至集線器叢集、並針對與叢集生命週期管理、應用程式生命週期管理、觀察性及安全性法規遵循相關的各種活動、提供服務要求。



如需詳細資訊、請參閱文件 ["請按這裡"](#)。

## 部署

### 為Kubernetes部署進階叢集管理

#### 先決條件

1. 用於顯示中樞叢集的Red Hat OpenShift叢集（高於4.5版）
2. 適用於託管叢集的Red Hat OpenShift叢集（高於4.4.3版）
3. 叢集管理存取Red Hat OpenShift叢集
4. 適用於Kubernetes的進階叢集管理Red Hat訂購

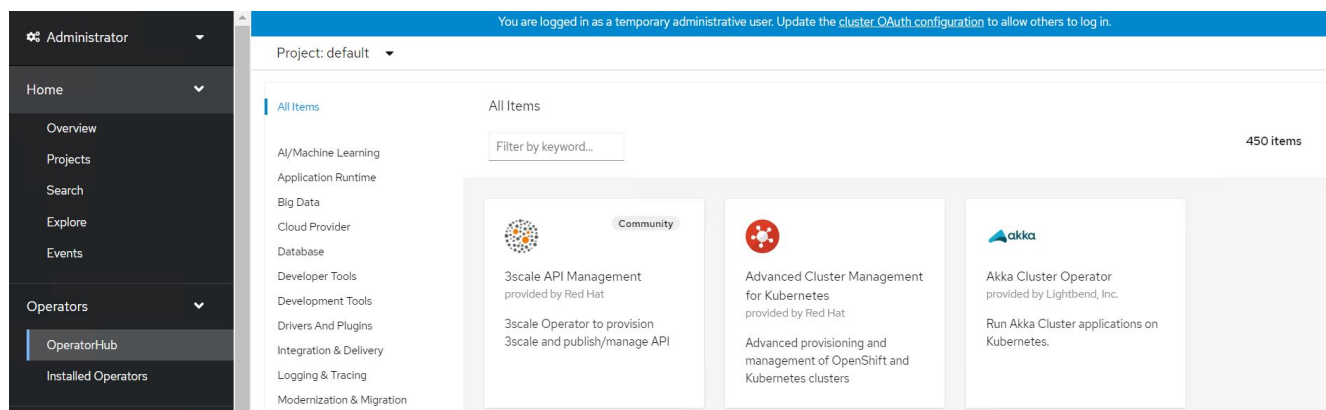
進階叢集管理是OpenShift叢集的附加元件、因此根據整個集線器和託管叢集所使用的功能、對硬體資源有特定的需求和限制。在調整叢集規模時、您必須將這些問題納入考量。請參閱文件 ["請按這裡"](#) 以取得更多詳細資料。

或者、如果集線器叢集有專屬節點來裝載基礎架構元件、而且您只想在這些節點上安裝「進階叢集管理」資源、則必須相應地將容許值和選取器新增至這些節點。如需詳細資料、請參閱文件 ["請按這裡"](#)。


### 為Kubernetes部署進階叢集管理

若要在OpenShift叢集上安裝適用於Kubernetes的進階叢集管理、請完成下列步驟：

1. 選擇OpenShift叢集做為中樞叢集、並以叢集管理權限登入。
2. 瀏覽至「運算子」>「運算子中樞」、然後搜尋Kubernetes的「進階叢集管理」。



3. 選取適用於Kubernetes的進階叢集管理、然後按一下安裝。



## Advanced Cluster Management for Kubernetes

2.2.3 provided by Red Hat

**Install**

**Latest version**

2.2.3

**Capability level**

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

**Provider type**

Red Hat

**Provider**

Red Hat

**Infrastructure features**

Disconnected

Red Hat Advanced Cluster Management for Kubernetes provides the multicluster hub, a central management console for managing multiple Kubernetes-based clusters across data centers, public clouds, and private clouds. You can use the hub to create Red Hat OpenShift Container Platform clusters on selected providers, or import existing Kubernetes-based clusters. After the clusters are managed, you can set compliance requirements to ensure that the clusters maintain the specified security requirements. You can also deploy business applications across your clusters.

Red Hat Advanced Cluster Management for Kubernetes also provides the following operators:

- Multicluster subscriptions: An operator that provides application management capabilities including subscribing to resources from a channel and deploying those resources on MCH-managed Kubernetes clusters based on placement rules.
- Hive for Red Hat OpenShift: An operator that provides APIs for provisioning and performing initial configuration of OpenShift clusters. These operators are used by the multicluster hub to provide its provisioning and application-management capabilities.

**How to Install**

Use of this Red Hat product requires a licensing and subscription agreement.

4. 在Install Operator（安裝操作員）畫面上、提供必要的詳細資料（NetApp建議保留預設參數）、然後按一下Install（安裝）。

## Install Operator

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

### Update channel \*

- ☐ release-2.0
- ☐ release-2.1
- ☒ release-2.2

### Installation mode \*

- ☐ All namespaces on the cluster (default)  
This mode is not supported by this Operator
- ☒ A specific namespace on the cluster  
Operator will be available in a single Namespace only.

### Installed Namespace \*

- ☒ Operator recommended Namespace: **PR** open-cluster-management

#### Namespace creation

Namespace **open-cluster-management** does not exist and will be created.

- ☐ Select a Namespace

### Approval strategy \*

- ☒ Automatic
- ☐ Manual

**Install**

Cancel

5. 等待操作員安裝完成。



**Advanced Cluster Management for Kubernetes**  
2.2.3 provided by Red Hat

## Installing Operator

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

[View installed Operators in Namespace open-cluster-management](#)

6. 安裝操作員之後、按一下「Create MultiClusterHub（建立MultiClusterHub）」。



## Advanced Cluster Management for Kubernetes

2.2.3 provided by Red Hat



### Installed operator - operand required

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

**MCH** MultiClusterHub **Required**

Advanced provisioning and management of OpenShift and Kubernetes clusters

Create MultiClusterHub

[View installed Operators in Namespace open-cluster-management](#)

- 在Create MultiClusterHub（建立MultiClusterHub）畫面上、按一下「建立」（在提供詳細資料之後）。這會啟動多叢集集集線器的安裝。

Project: open-cluster-management

[Advanced Cluster Management for Kubernetes](#) > [Create MultiClusterHub](#)

### Create MultiClusterHub

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

Configure via: ☒ Form view ☐ YAML view

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



MultiClusterHub

provided by Red Hat

MultiClusterHub defines the configuration for an instance of the MultiCluster Hub

Name \*

multiclusterhub

Labels

app=frontend

> [Advanced configuration](#)

Create


Cancel

- 在所有Pod移至開放式叢集管理命名空間中的執行狀態、且操作員移至成功狀態之後、就會安裝適用於Kubernetes的進階叢集管理。




## 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	Managed Namespaces	Status	Provided APIs
 <b>Advanced Cluster Management for Kubernetes</b> 2.2.3 provided by Red Hat	NS open-cluster-management	✓ Succeeded Up to date	MultiClusterHub ClusterManager ClusterDeployment ClusterState <a href="#">View 25 more...</a>

9. 完成集線器安裝需要一些時間、完成之後、MultiCluster集線器會移至執行中狀態。

Installed Operators > Operator details

 **Advanced Cluster Management for Kubernetes**  
2.2.3 provided by Red Hat

Actions

Details | **YAML** | Subscription | Events | All instances | **MultiClusterHub** | ClusterManager | ClusterDeployment | ClusterState

### MultiClusterHubs

Create MultiClusterHub

Name	Kind	Status	Labels
MCH multiclusterhub	MultiClusterHub	Phase: ✓ Running	No labels

10. 它會在開放式叢集管理命名空間中建立路由。連線至路由中的URL、以存取進階叢集管理主控台。

## Routes

[Create Route](#)

Filter Name mul

Name mul Clear all filters

Name	Status	Location	Service
RT multcloud-console	✓ Accepted	<a href="https://multicloud-console.apps.ocp-vmware2.cie.netapp.com">https://multicloud-console.apps.ocp-vmware2.cie.netapp.com</a>	S management-ingress

功能

特色：採用NetApp的Red Hat OpenShift上的Kubernetes進階叢集管理

## 叢集生命週期管理

若要管理不同的OpenShift叢集、您可以建立或匯入至「進階叢集管理」。

1. 首先瀏覽至自動化基礎架構>叢集。
2. 若要建立新的OpenShift叢集、請完成下列步驟：
  - a. 建立供應商連線：瀏覽至「提供者連線」、然後按一下「新增連線」、提供與所選提供者類型對應的所有詳細資料、然後按一下「新增」。

Select a provider and enter basic information

Provider \* ⓘ

aws Amazon Web Services

Connection name \* ⓘ

nik-hcl-aws

Namespace \* ⓘ

default

Configure your provider connection

Base DNS domain ⓘ

cie.netapp.com

AWS access key ID \* ⓘ

AKIATCFBZDOIASDSA

AWS secret access key \* ⓘ

.....

Red Hat OpenShift pull secret \* ⓘ

```
FuS3pNbktVaHpINfc2MkZsbmtBVGN6TktmUIZXcHcxOW9teEZwQ0lYIzId3cjJobGxJeDBQNOxIzE0yeGM5Q0ZwZk5RR2JUanlxNnNUM2IRbOFJb
UFjNCIBYlpEWVZEOHItNkxTMDZPUVpoWFRHcGwtREIDQ2RSYlURaTlxblLT2oyQ3pVeUJfNllwcENSa2YyOUUyLWZGSFVfNA==", "email": "Nikhil.k
ulkarni@netapp.com"}, "registry.redhat.io":
```

SSH private key \* ⓘ

```
-----BEGIN OPENSSH PRIVATE KEY-----
b3BlbnNzaC1rZXktdjEAAAABG5vbmUAAAABasdadssadm9uZQAAAAAAAAABAAAMwAAAAatzc2gtZW
QyNTUxOQAAACCLcwLgAvSIHAEp+DevIRNzaG2zkNreMIZ/UHyf0UWvAAAAAJhy/wa6xf8Gu
```

SSH public key \* ⓘ

```
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIltzAuAC746agdh2lcB4/4N6/VE3NobbOQ2t4zVn9QfJ/RRa8A root@nik-rhel8
```

- b. 若要建立新叢集、請瀏覽至「叢集」、然後按一下「新增叢集」>「建立叢集」。提供叢集和對應供應商的詳細資料、然後按一下「Create（建立）」。



Configuration

Cluster name \* ⓘ

rh-aws

Distribution

Select the type of Kubernetes distribution to use for your cluster.

Red Hat OpenShift

Select an infrastructure provider to host your Red Hat OpenShift cluster.

aws Amazon Web Services

Google Cloud

Microsoft Azure

VMware vSphere

Bare Metal

Release image \* ⓘ

quay.io/openshift-release-dev/ocp-release:4.7.12-x86\_64

Provider connection \* ⓘ

nik-hcl-aws

Add a connection

- c. 建立叢集之後、它會顯示在叢集清單中、並顯示「Ready（就緒）」狀態。
3. 若要匯入現有的叢集、請完成下列步驟：
- 瀏覽至「叢集」、然後按一下「新增叢集」>「匯入現有的叢集」。
  - 輸入叢集名稱、然後按一下「Save Import（儲存匯入）」和「Generate Code（產生程式碼）」。此時會顯示新增現有叢集的命令。
  - 按一下「Copy Command（複製命令）」、然後在要新增至集線器叢集的叢集上執行命令。這會在叢集上啟動必要代理程式的安裝、完成此程序之後、叢集會顯示在叢集清單中、並顯示「Ready（就緒）」狀態。

**Name \***

ocp-vmw1

**Additional labels**

Once you click on "Save import and generate code", the information you entered will be used to generate the code and cannot be modified anymore. If you wish to change any information, you will have to delete and re-import this cluster.

Code generated successfully Import saved

**Run a command**

**1. Copy this command**

Click the button to have the command automatically copied to your clipboard.

Copy command

**2. Run this command with kubectl configured for your targeted cluster to start the import**

Log in to the existing cluster in your terminal and run the command.

View cluster Import another

4. 建立及匯入多個叢集之後、您可以從單一主控台監控及管理這些叢集。

特色：採用NetApp的Red Hat OpenShift上的Kubernetes進階叢集管理

應用程式生命週期管理

若要在一組叢集之間建立應用程式並加以管理、

1. 從側邊列瀏覽至「管理應用程式」、然後按一下「建立應用程式」。提供您要建立的應用程式詳細資料、然後按一下「Save（儲存）」。

Create an application YAML: Off

Cancel

Save


**Name\*** ⓘ  
demo-app

**Namespace\*** ⓘ  
default

^ **Repository location for resources**

^ **Repository types**

Select the type of repository where resources that you want to deploy are located

 Git ☒

**URL\*** ⓘ  
https://github.com/open-cluster-management/acm-hive-openshift-releases.git

**Branch** ⓘ  
main

**Path** ⓘ  
clusterImageSets/fast/4.7

2. 安裝應用程式元件之後、應用程式會出現在清單中。

## Applications

Refresh every 15s

Last update: 7:36:23 PM

Overview

Advanced configuration

Create application

Search						
Name	Namespace	Clusters	Resource	Time window	Created	
demo-app	default	Local	Git		8 days ago	
1 - 1 of 1						

3. 現在可以從主控台監控及管理應用程式。

特色：採用NetApp的Red Hat OpenShift上的Kubernetes進階叢集管理

## 治理與風險


此功能可讓您針對不同的叢集定義法規遵循原則、並確保叢集符合此原則。您可以設定原則來通知或修正任何偏離或違反規則的情況。

1. 從側邊列導覽至「治理與風險」。
2. 若要建立規範原則、請按一下「建立原則」、輸入原則標準的詳細資料、然後選取應遵守此原則的叢集。如果您想要自動修正此原則的違規、請選取「強制執行（若有支援）」核取方塊、然後按一下「建立」。


# Create policy YAML: Off

**Name \***

policy-complianceoperator

**Namespace \*** 

default

**Specifications \***  ComplianceOperator**Cluster selector**  local-cluster: "true"**Standards**  NIST-CSF**Categories**  PR.IP Information Protection Processes and Procedures**Controls**  PR.IP-1 Baseline Configuration☐ **Enforce if supported** ☐ **Disable policy** 

3. 設定所有必要的原則之後、即可從「進階叢集管理」監控及修正任何原則或叢集違規。

## Governance and risk ⓘ

[Filter](#)[Refresh every 10s](#) ▼

Last update: 12:54:01 PM

[Create policy](#)

Summary 1

Standards ▼

### NIST-CSF

**No violations found**

Based on the industry standards, there are no cluster or policy violations.

[Policies](#)[Cluster violations](#)

Policy name ↑	Namespace ↑	Remediation ↑	Cluster violations	Standards ↑	Categories ↑	Controls ↑	Created ↓
policy-complianceoperator	default	inform	✓ 0/1	NIST-CSF	PR.IP Information Protection Processes and Procedures	PR.IP-1 Baseline Configuration	32 minutes ago ⋮

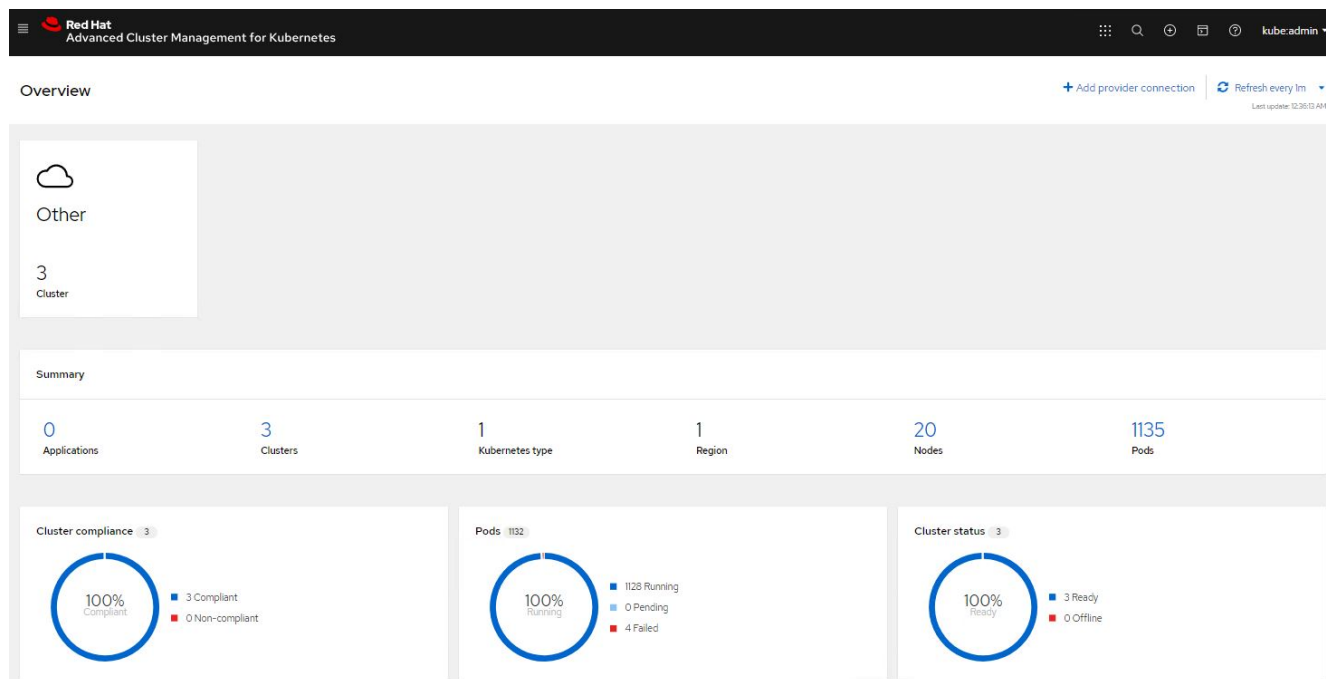
1 - 1 of 1 ▼ << < 1 of 1 > >>

特色：採用NetApp的Red Hat OpenShift上的Kubernetes進階叢集管理

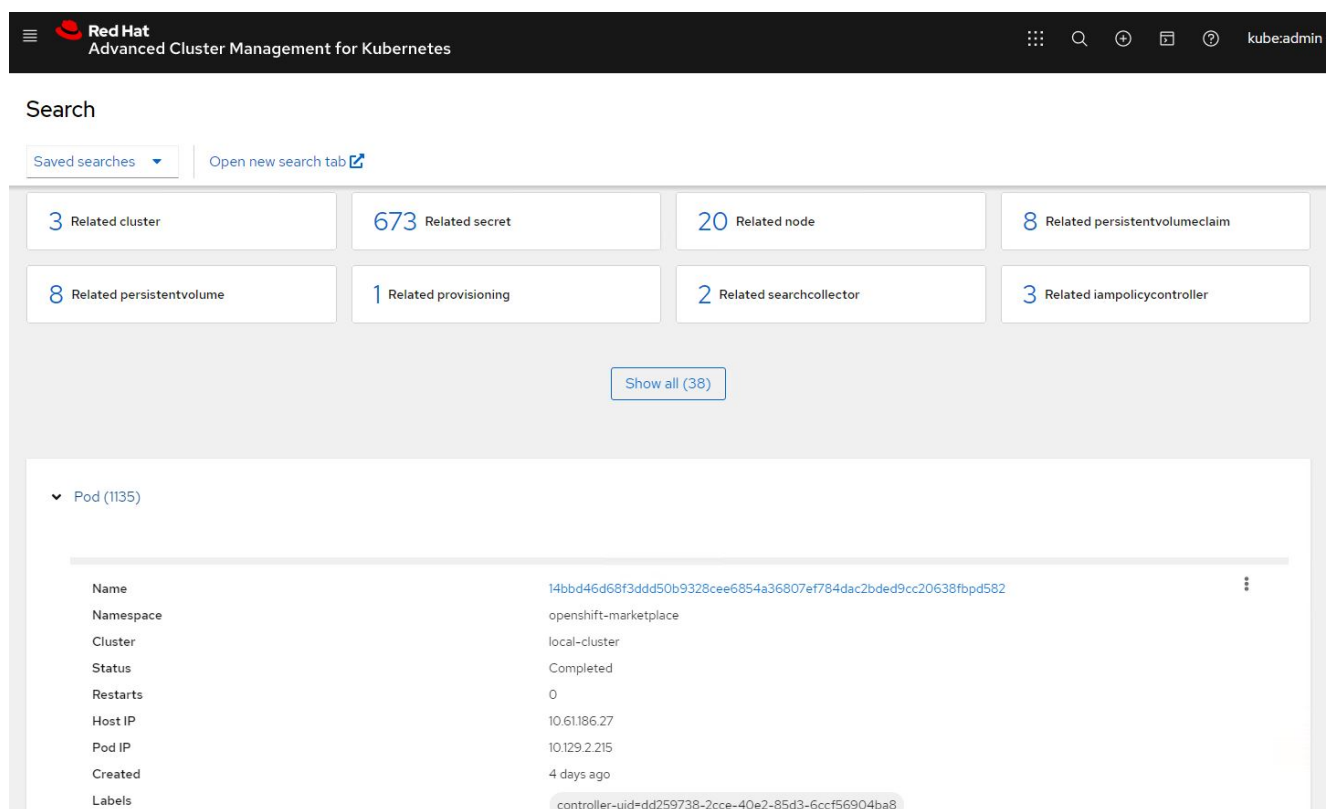
### 觀察能力

適用於Kubernetes的進階叢集管理提供一種方法、可監控所有叢集上的節點、Pod、應用程式和工作負載。

1. 瀏覽至「觀察環境」>「總覽」。



2. 所有叢集的所有Pod和工作負載都會根據各種篩選器進行監控和排序。按一下「Pod」以檢視對應的資料。



3. 叢集內的所有節點都會根據各種資料點進行監控與分析。按一下節點、深入瞭解對應的詳細資料。

## Search

Saved searches ▾ Open new search tab [🔗](#)

3 Related cluster 1k Related pod 12 Related service

Show all (3)

▼ Node (20)

Name ↑	Cluster ▾	Role ▾	Architecture ▾	OS image ▾	CPU ▾	Created ▾	Labels ▾
<a href="#">ocp-master-1.ocp-bare-metal.cie.netapp.com</a>	ocp-bare-metal	master; worker	amd64	Red Hat Enterprise Linux CoreOS 4783.202103292105-0 (Ootpa)	48	a month ago	beta.kubernetes.io/arch=amd64 beta.kubernetes.io/os=linux kubernetes.io/arch=amd64 5 more
<a href="#">ocp-master-2.ocp-bare-metal.cie.netapp.com</a>	ocp-bare-metal	master; worker	amd64	Red Hat Enterprise Linux CoreOS 4783.202103292105-0 (Ootpa)	48	a month ago	beta.kubernetes.io/arch=amd64 beta.kubernetes.io/os=linux kubernetes.io/arch=amd64 5 more
<a href="#">ocp-master-3.ocp-bare-metal.cie.netapp.com</a>	ocp-bare-metal	master; worker	amd64	Red Hat Enterprise Linux CoreOS 4783.202103292105-0 (Ootpa)	48	a month ago	beta.kubernetes.io/arch=amd64 beta.kubernetes.io/os=linux kubernetes.io/arch=amd64 5 more

4. 所有叢集都會根據不同的叢集資源和參數進行監控和組織。按一下叢集以檢視叢集詳細資料。

## Search

Saved searches ▾ Open new search tab [🔗](#)

3k Related secret 787 Related pod 15 Related persistentvolumeclaim 17 Related node 1 Related application

15 Related persistentvolume 1 Related searchcollector 8 Related clusterclaim 3 Related resourcequota 5 Related identity

Show all (159)

▼ Cluster (2)

Name ↑	Available ▾	Hub accepted ▾	Joined ▾	Nodes ▾	Kubernetes version ▾	CPU ▾	Memory ▾	Console URL ▾	Labels ▾
<a href="#">local-cluster</a>	True	True	True	8	v1.20.0+c8905da	84	418501Mi	<a href="#">Launch</a>	cloud-VSphere clusterID=148632d9-69d5-4ae4-98ee-8dff886463c3 installer.name=multiclusterhub 4 more
<a href="#">ocp-vmw</a>	True	True	True	9	v1.20.0+df9c838	28	111981Mi	<a href="#">Launch</a>	cloud-VSphere clusterID=9d76ac4e-4aae-4d45-a2e8-11b6b54282fe name=ocp-vmw 1 more

特色：採用NetApp的Red Hat OpenShift上的Kubernetes進階叢集管理

## 在多個叢集上建立資源

Kubernetes的進階叢集管理功能可讓使用者從主控台同時在一或多個託管叢集上建立資源。舉例來ONTAP 說、如果您在不同站台有OpenShift叢集、並以不同的NetApp支援叢集做為後盾、而且想要在兩個站台上配置PVc、您可以按一下頂端列上的 (+) 符號。然後選取您要在其中建立永久虛擬基礎虛擬基礎網路的叢集、貼上資源Yaml、然後按一下「Create (建立)」。



# Create resource

[Cancel](#)[Create](#)

Clusters | Select the clusters where the resource(s) will be deployed.

2 x local-cluster,  
ocp-vmw

Resource configuration | Enter the configuration manifest for the resource(s).

YAML

```
1 kind: PersistentVolumeClaim
2 apiVersion: v1
3 metadata:
4   name: demo-pvc
5 spec:
6   accessModes:
7     - ReadWriteOnce
8   resources:
9     requests:
10      storage: 1Gi
11   storageClassName: ocp-trident
```

## 版權資訊

Copyright © 2024 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。