Red Hat OpenShift con NetApp

NetApp Solutions

NetApp
April 26, 2024

Sommario

NVA-1160: Red Hat OpenShift con NetApp

Alan Cowles e Nikhil M Kulkarni, NetApp

Questo documento di riferimento fornisce la convalida dell'implementazione della soluzione Red Hat OpenShift, implementata tramite l'infrastruttura IPI (Installer Provised Infrastructure) in diversi ambienti di data center come validati da NetApp. Inoltre, descrive in dettaglio l'integrazione dello storage con i sistemi di storage NetApp utilizzando Astra Trident Storage orchestrator per la gestione dello storage persistente. Infine, vengono analizzate e documentate una serie di validazioni delle soluzioni e casi di utilizzo reali.

Casi di utilizzo

La soluzione Red Hat OpenShift con NetApp è progettata per offrire un valore eccezionale ai clienti con i seguenti casi di utilizzo:

- Facile da implementare e gestire Red Hat OpenShift implementato utilizzando IPI (Installer Provised Infrastructure) su bare metal, Red Hat OpenStack Platform, Red Hat Virtualization e VMware vSphere.
- Potenza combinata dei container aziendali e dei carichi di lavoro virtualizzati con Red Hat OpenShift implementato virtualmente su OSP, RHV o vSphere o su bare metal con la virtualizzazione OpenShift.
- Configurazione e casi d'utilizzo reali che evidenziano le funzionalità di Red Hat OpenShift se utilizzato con lo storage NetApp e Astra Trident, l'orchestrator dello storage open source per Kubernetes.

Valore di business

Le aziende stanno adottando sempre più pratiche DevOps per creare nuovi prodotti, abbreviare i cicli di rilascio e aggiungere rapidamente nuove funzionalità. A causa della loro natura innata e agile, i container e i microservizi svolgono un ruolo cruciale nel supporto delle pratiche DevOps. Tuttavia, la pratica di DevOps su scala di produzione in un ambiente aziendale presenta le proprie sfide e impone determinati requisiti all'infrastruttura sottostante, come ad esempio:

- Alta disponibilità a tutti i livelli dello stack
- Procedure di implementazione semplici
- Operazioni e aggiornamenti senza interruzioni
- Infrastruttura programmabile e basata su API per restare al passo con l'agilità dei microservizi
- Multi-tenancy con garanzie di performance
- Possibilità di eseguire contemporaneamente carichi di lavoro virtualizzati e containerizzati
- Possibilità di scalare l'infrastruttura in modo indipendente in base alle esigenze dei carichi di lavoro

Red Hat OpenShift con NetApp riconosce queste sfide e presenta una soluzione che aiuta a risolvere ogni problema implementando l'implementazione completamente automatizzata di RedHat OpenShift IPI nell'ambiente del data center scelto dal cliente.

Panoramica della tecnologia

La soluzione Red Hat OpenShift con NetApp comprende i seguenti componenti principali:

Red Hat OpenShift Container Platform

Red Hat OpenShift Container Platform è una piattaforma aziendale Kubernetes completamente supportata. Red Hat apporta diversi miglioramenti a Kubernetes open-source per offrire una piattaforma applicativa con tutti i componenti completamente integrati per la creazione, l'implementazione e la gestione delle applicazioni containerizzate.

Per ulteriori informazioni, visita il sito Web di OpenShift "qui".

Sistemi storage NetApp

NetApp dispone di diversi sistemi storage perfetti per data center aziendali e implementazioni di cloud ibrido. Il portfolio NetApp include i sistemi storage NetApp ONTAP, NetApp Element e NetApp e-Series, tutti in grado di fornire storage persistente per le applicazioni containerizzate.

Per ulteriori informazioni, visitare il sito Web di NetApp "qui".

Integrazioni di storage NetApp

NetApp Astra Control Center offre un'ampia gamma di servizi di gestione dei dati application-aware e storage per carichi di lavoro Kubernetes stateful, implementati in un ambiente on-premise e basati sulla tecnologia di protezione dei dati NetApp.

Per ulteriori informazioni, visitare il sito Web di NetApp Astra "qui".

Astra Trident è un orchestrator di storage open-source e completamente supportato per container e distribuzioni Kubernetes, incluso Red Hat OpenShift.

Per ulteriori informazioni, visita il sito web di Astra Trident "qui".

Opzioni di configurazione avanzate

Questa sezione è dedicata alle personalizzazioni che gli utenti reali dovrebbero eseguire durante l'implementazione di questa soluzione in produzione, ad esempio la creazione di un registro di immagini private dedicato o l'implementazione di istanze personalizzate di bilanciamento del carico.

Matrice di supporto corrente per le release validate

Tecnologia	Scopo	Versione del software
NetApp ONTAP	Storage	9.8, 9.9.1
NetApp Element	Storage	12.3
NetApp Astra Control Center	Gestione dei dati consapevole dell'applicazione	21.12.60
NetApp Astra Trident	Orchestrazione dello storage	22.01.0
Red Hat OpenShift	Orchestrazione di container	4.6 EUS, 4.7, 4.8
Piattaforma Red Hat OpenStack	Infrastruttura di cloud privato	16.1
Virtualizzazione Red Hat	Virtualizzazione del data center	4.4
VMware vSphere	Virtualizzazione del data center	6.7U3

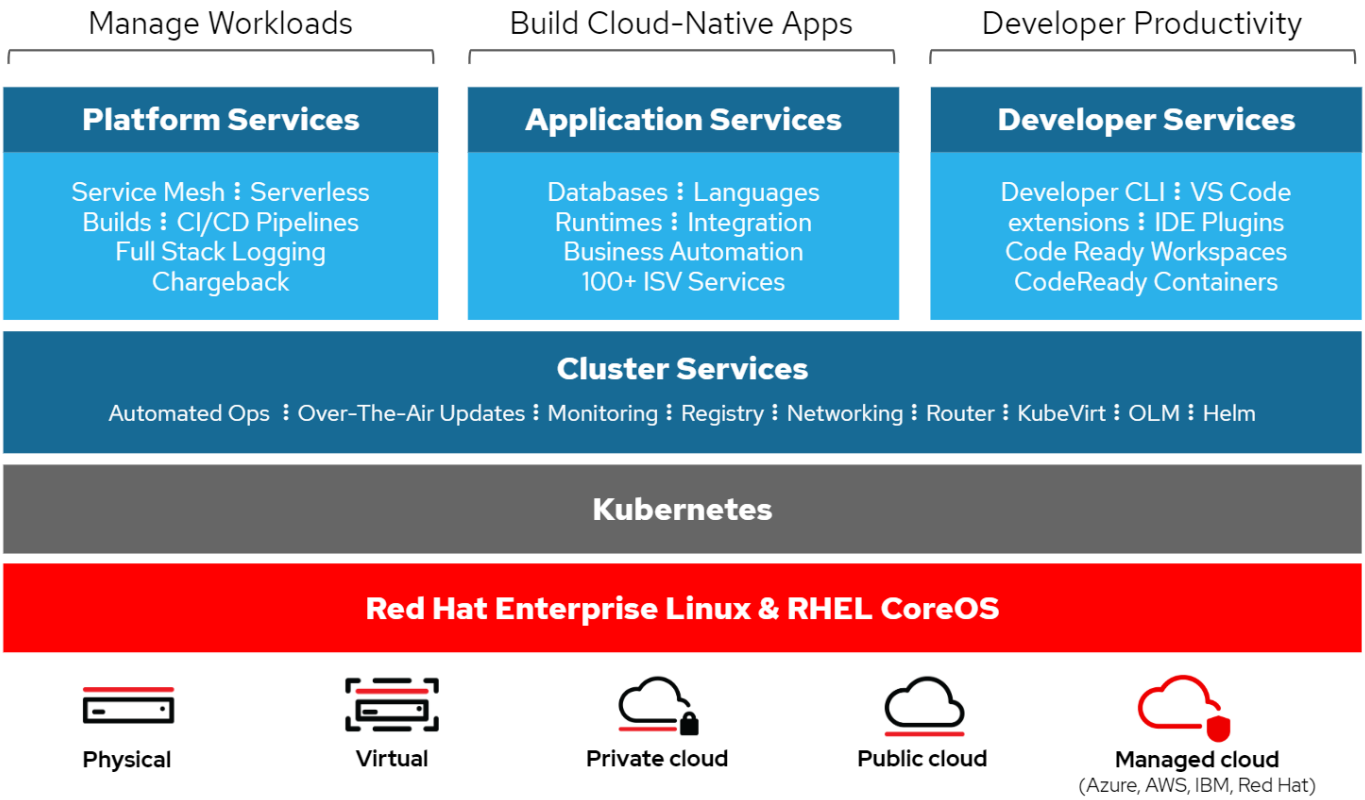
Panoramica di OpenShift

Red Hat OpenShift Container Platform unisce le operazioni IT e di sviluppo su un'unica piattaforma per creare, implementare e gestire le applicazioni in modo coerente tra infrastrutture cloud ibride e on-premise. Red Hat OpenShift si basa su innovazioni open-source e standard di settore, tra cui Kubernetes e Red Hat Enterprise Linux CoreOS, la distribuzione Linux aziendale leader a livello mondiale progettata per carichi di lavoro basati su container. OpenShift fa parte del programma Cloud Native Computing Foundation (CNCF) Certified Kubernetes, che offre portabilità e interoperabilità dei carichi di lavoro dei container.

Red Hat OpenShift offre le seguenti funzionalità:

- **Provisioning self-service** gli sviluppatori possono creare applicazioni su richiesta in modo rapido e semplice utilizzando gli strumenti più utilizzati, mentre le operazioni mantengono il pieno controllo sull'intero ambiente.
- **Storage persistente** grazie al supporto per lo storage persistente, OpenShift Container Platform consente di eseguire sia applicazioni stateful che applicazioni stateless native del cloud.
- **Integrazione continua e sviluppo continuo (ci/CD)** questa piattaforma di codice sorgente gestisce le immagini di build e distribuzione su larga scala.
- **Standard open-source** questi standard incorporano l'Open Container Initiative (OCI) e Kubernetes per l'orchestrazione dei container, oltre ad altre tecnologie open-source. Non ti limiterai alla tecnologia o alla roadmap di business di un vendor specifico.
- **Pipeline ci/CD** OpenShift fornisce un supporto immediato per le pipeline ci/CD, in modo che i team di sviluppo possano automatizzare ogni fase del processo di distribuzione dell'applicazione e assicurarsi che venga eseguito ad ogni modifica apportata al codice o alla configurazione dell'applicazione.
- **RBAC (Role-Based Access Control)** questa funzionalità fornisce il monitoraggio di team e utenti per aiutare a organizzare un gruppo di sviluppatori di grandi dimensioni.
- **Automated Build and Deploy** OpenShift offre agli sviluppatori la possibilità di creare le proprie applicazioni containerizzate o di far costruire i contenitori dal codice sorgente dell'applicazione o anche dai file binari. La piattaforma automatizza quindi l'implementazione di queste applicazioni nell'infrastruttura in base alle caratteristiche definite per le applicazioni. Ad esempio, la quantità di risorse da allocare e la posizione dell'infrastruttura da implementare per garantire la conformità con le licenze di terze parti.
- **Ambienti coerenti** OpenShift garantisce che l'ambiente fornito agli sviluppatori e per tutto il ciclo di vita dell'applicazione sia coerente dal sistema operativo alle librerie, alla versione runtime (ad esempio Java runtime), e persino il runtime dell'applicazione in uso (ad esempio, tomcat) per eliminare i rischi derivanti da ambienti incoerenti.
- **Gestione della configurazione** la configurazione e la gestione dei dati sensibili sono integrate nella piattaforma per garantire che all'applicazione venga fornita una configurazione di applicazione coerente e indipendente dall'ambiente, indipendentemente dalle tecnologie utilizzate per creare l'applicazione o dall'ambiente in cui si trova
implementati.
- **Registri delle applicazioni e parametri metrici.** il feedback rapido è un aspetto importante dello sviluppo delle applicazioni. Il monitoraggio integrato e la gestione dei log di OpenShift forniscono agli sviluppatori metriche immediate per studiare il comportamento dell'applicazione tra le modifiche e per risolvere i problemi il prima possibile nel ciclo di vita dell'applicazione.
- **Catalogo sicurezza e container** OpenShift offre la multi-tenancy e protegge l'utente dall'esecuzione di

codice dannoso utilizzando la sicurezza stabilita con Security-Enhanced Linux (SELinux), CGroups e Secure Computing Mode (seccomp) per isolare e proteggere i contenitori. Fornisce inoltre la crittografia tramite certificati TLS per i vari sottosistemi e l'accesso ai container certificati Red Hat (access.redhat.com/containers) sottoposti a scansione e classificati con un'enfasi specifica sulla sicurezza per fornire container applicativi certificati, affidabili e sicuri agli utenti finali.



Metodi di implementazione per Red Hat OpenShift

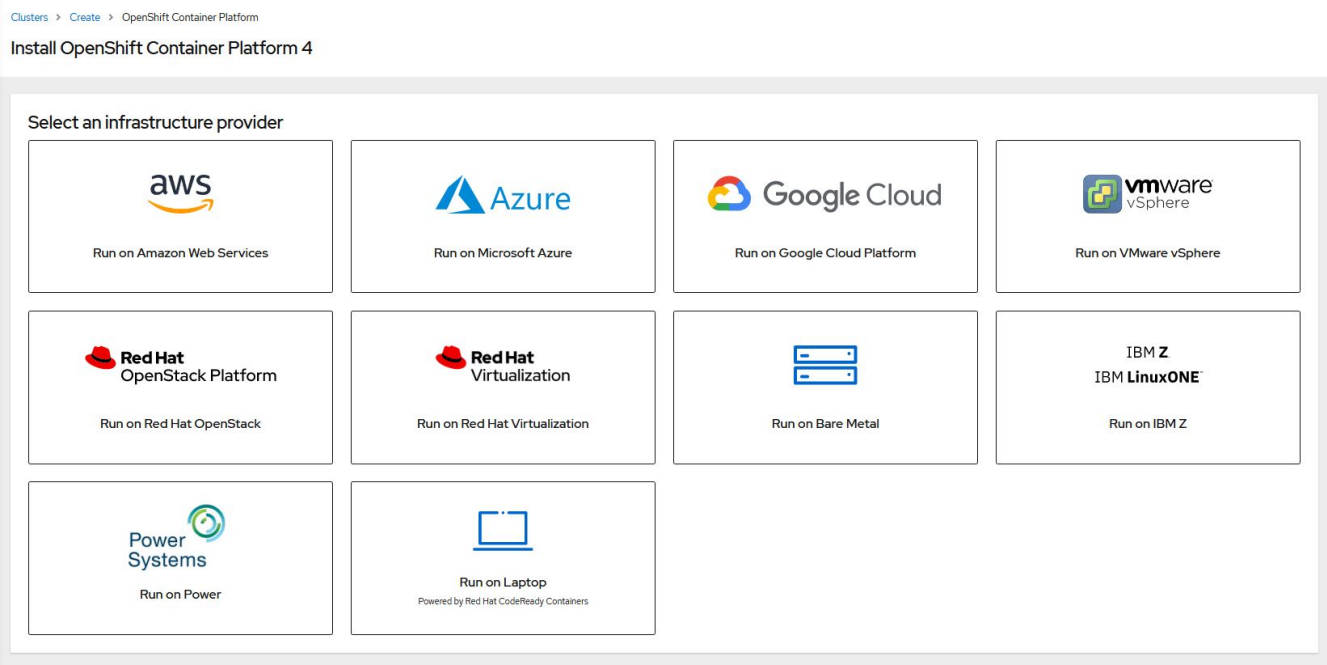
A partire da Red Hat OpenShift 4, i metodi di implementazione di OpenShift includono implementazioni manuali che utilizzano l'infrastruttura con provisioning utente (UPI) per implementazioni altamente personalizzate o implementazioni completamente automatizzate che utilizzano l'infrastruttura con provisioning dell'installatore (IPI).

Il metodo di installazione IPI è il metodo preferito nella maggior parte dei casi, perché consente la rapida implementazione dei cluster OpenShift per gli ambienti di sviluppo, test e produzione.

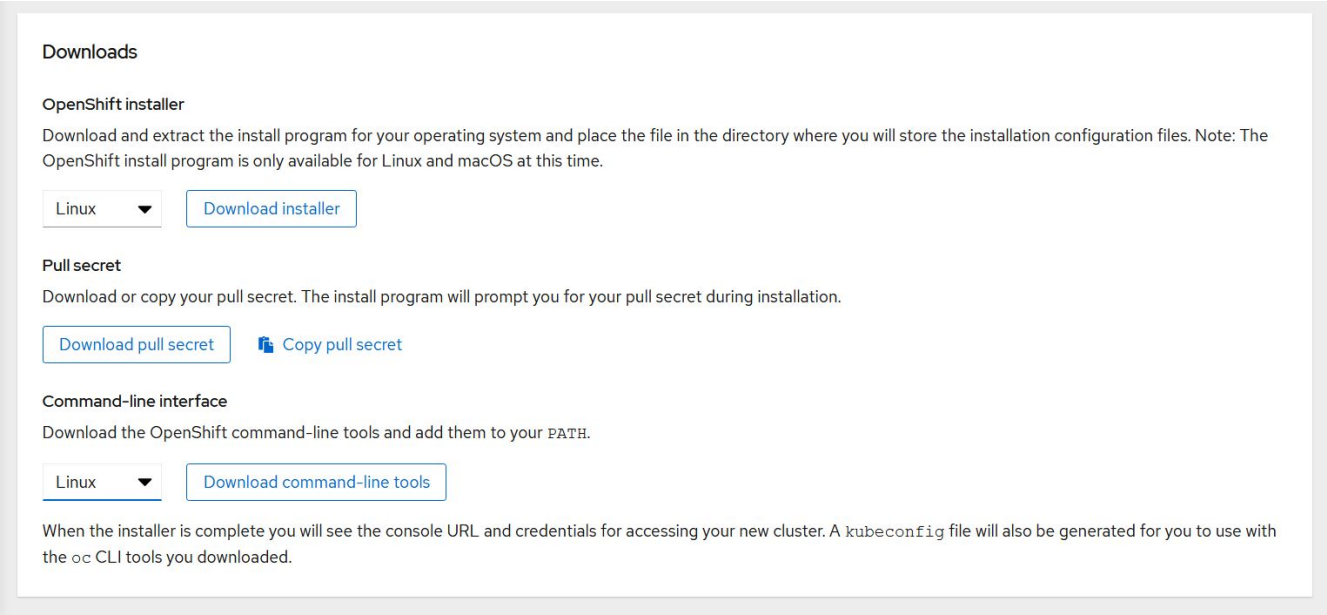
Installazione IPI di Red Hat OpenShift

La distribuzione dell'infrastruttura IPI (Installer Provised Infrastructure) di OpenShift prevede questi passaggi di alto livello:

1. Visita Red Hat OpenShift "sito web" E accedi con le tue credenziali SSO.
2. Seleziona l'ambiente in cui desideri implementare Red Hat OpenShift.



3. Nella schermata successiva, scaricare il programma di installazione, l'esclusivo segreto pull e gli strumenti CLI per la gestione.



4. Seguire la "istruzioni per l'installazione" Fornito da Red Hat per l'implementazione nel vostro ambiente preferito.

Implementazioni OpenShift validate da NetApp

NetApp ha testato e validato l'implementazione di Red Hat OpenShift nei propri laboratori utilizzando il metodo di implementazione IPI (Installer Provised Infrastructure) in ciascuno dei seguenti ambienti di data center:

- "OpenShift su bare metal"
- "OpenShift sulla piattaforma Red Hat OpenStack"
- "OpenShift sulla virtualizzazione Red Hat"

- "OpenShift su VMware vSphere"

OpenShift su bare metal

OpenShift su bare metal offre un'implementazione automatica della piattaforma container OpenShift sui server commodity.

OpenShift su bare metal è simile alle implementazioni virtuali di OpenShift, che offrono facilità di implementazione, provisioning rapido e scalabilità dei cluster OpenShift, supportando al contempo i carichi di lavoro virtualizzati per le applicazioni non pronte per essere containerizzate. L'implementazione su bare metal non richiede l'overhead aggiuntivo necessario per gestire l'ambiente dell'hypervisor host oltre all'ambiente OpenShift. Implementando direttamente sui server bare metal, è possibile ridurre anche i limiti di overhead fisico dovuti alla condivisione delle risorse tra l'host e l'ambiente OpenShift.

OpenShift su bare metal offre le seguenti funzionalità:

- **Implementazione IPI o assistita** con un cluster OpenShift distribuito da IPI (Installer Provisioning Infrastructure) su server bare metal, i clienti possono implementare un ambiente OpenShift altamente versatile e facilmente scalabile direttamente su commodity server, senza la necessità di gestire un livello di hypervisor.
- **Compact cluster design** per ridurre al minimo i requisiti hardware, OpenShift su bare metal consente agli utenti di implementare cluster di soli 3 nodi, consentendo ai nodi del piano di controllo OpenShift di agire anche come nodi di lavoro e contenitori host.
- **Virtualizzazione OpenShift** OpenShift può eseguire macchine virtuali all'interno dei container utilizzando OpenShift Virtualization. Questa virtualizzazione nativa per container esegue l'hypervisor KVM all'interno di un container e collega volumi persistenti per lo storage delle macchine virtuali.
- **Infrastruttura ottimizzata per ai/ML** implementate applicazioni come Kubeflow per applicazioni di apprendimento automatico incorporando nodi di lavoro basati su GPU nel vostro ambiente OpenShift e sfruttando OpenShift Advanced Scheduling.

Progettazione di rete

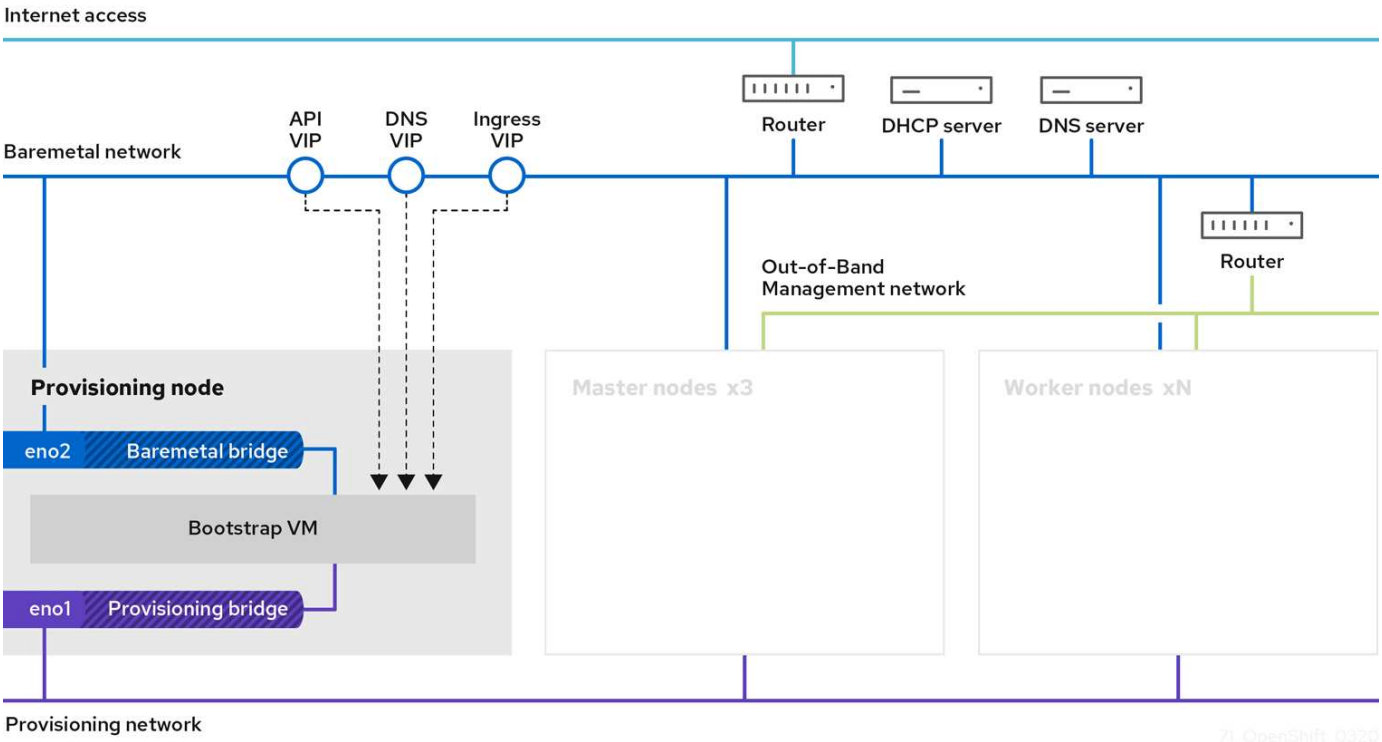
La soluzione Red Hat OpenShift su NetApp utilizza due switch dati per fornire connettività dati primaria a 25 Gbps. Utilizza inoltre due switch di gestione che forniscono connettività a 1 Gbps per la gestione in-band dei nodi di storage e gestione out-of-band per la funzionalità IPMI.

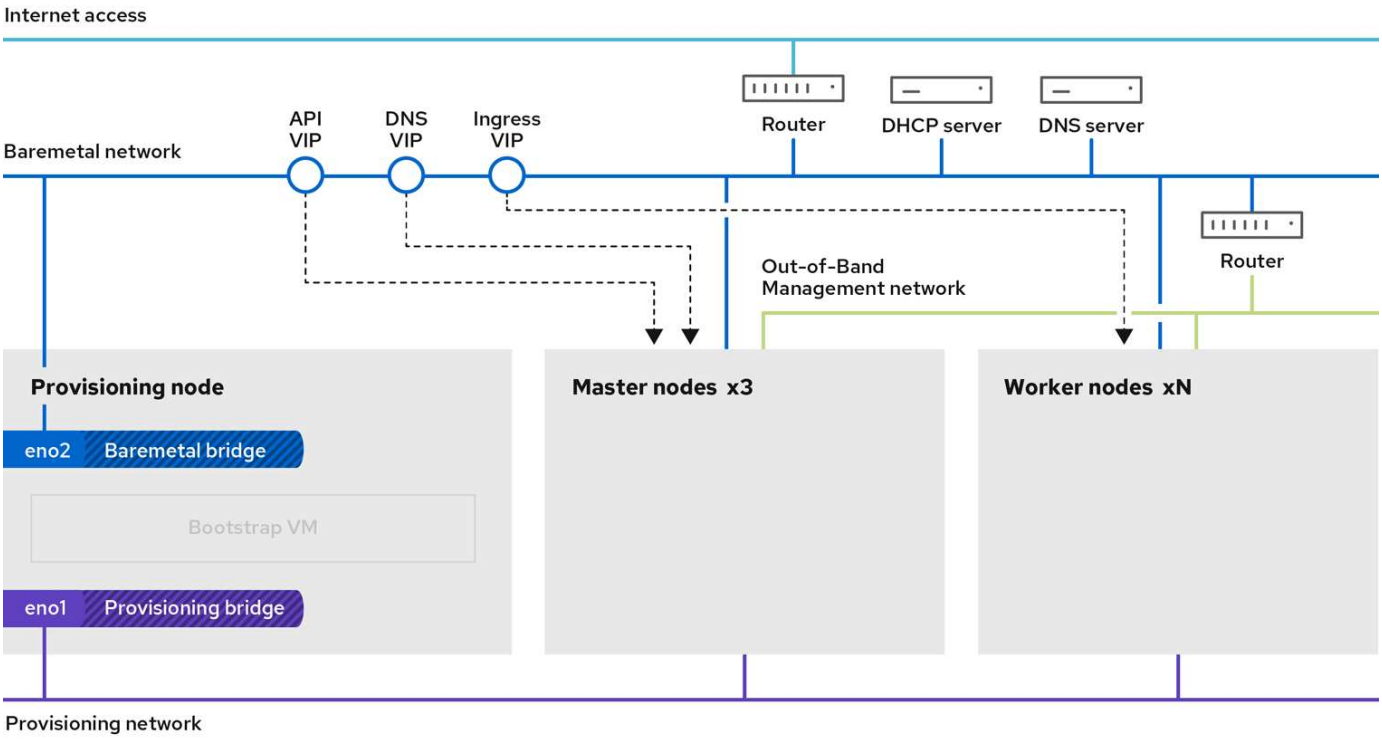
Per l'implementazione di OpenShift Bare-Metal IPI, è necessario creare un nodo di provisioning, una macchina Red Hat Enterprise Linux 8 che deve avere interfacce di rete collegate a reti separate.

- **Provisioning network** questa rete viene utilizzata per avviare i nodi bare-metal e installare le immagini e i pacchetti necessari per distribuire il cluster OpenShift.
- **Rete bare-metal** questa rete viene utilizzata per la comunicazione pubblica del cluster dopo la sua distribuzione.

Per la configurazione del nodo di provisioning, il cliente crea interfacce di bridge che consentono al traffico di instradare correttamente sul nodo stesso e sulla macchina virtuale Bootstrap fornita a scopo di implementazione. Una volta implementato il cluster, l'API e gli indirizzi VIP di ingresso vengono migrati dal nodo di boot strap al cluster appena implementato.

Le immagini seguenti illustrano l'ambiente sia durante l'implementazione IPI che al termine dell'implementazione.





Requisiti VLAN

La soluzione Red Hat OpenShift con NetApp è progettata per separare logicamente il traffico di rete per scopi diversi utilizzando Virtual Local Area Network (VLAN).

VLAN	Scopo	ID VLAN
Rete di gestione fuori banda	Gestione per nodi bare metal e IPMI	16
Rete bare-metal	Rete per i servizi OpenShift una volta che il cluster è disponibile	181
Rete di provisioning	Rete per l'avvio PXE e l'installazione di nodi bare metal tramite IPI	3485

Sebbene ciascuna di queste reti sia virtualmente separata da VLAN, ciascuna porta fisica deve essere impostata in modalità di accesso con la VLAN primaria assegnata, poiché non esiste alcun modo di passare un tag VLAN durante una sequenza di avvio PXE.

Risorse di supporto dell'infrastruttura di rete

Prima dell'implementazione della piattaforma container OpenShift, è necessario installare la seguente infrastruttura:

- Almeno un server DNS che fornisce una risoluzione completa del nome host accessibile dalla rete di gestione in-band e dalla rete VM.
- Almeno un server NTP accessibile dalla rete di gestione in-band e dalla rete VM.
- (Opzionale) connettività Internet in uscita per la rete di gestione in banda e la rete VM.

OpenShift sulla piattaforma Red Hat OpenStack

La piattaforma Red Hat OpenStack offre una base integrata per creare, implementare e scalare un cloud privato OpenStack sicuro e affidabile.

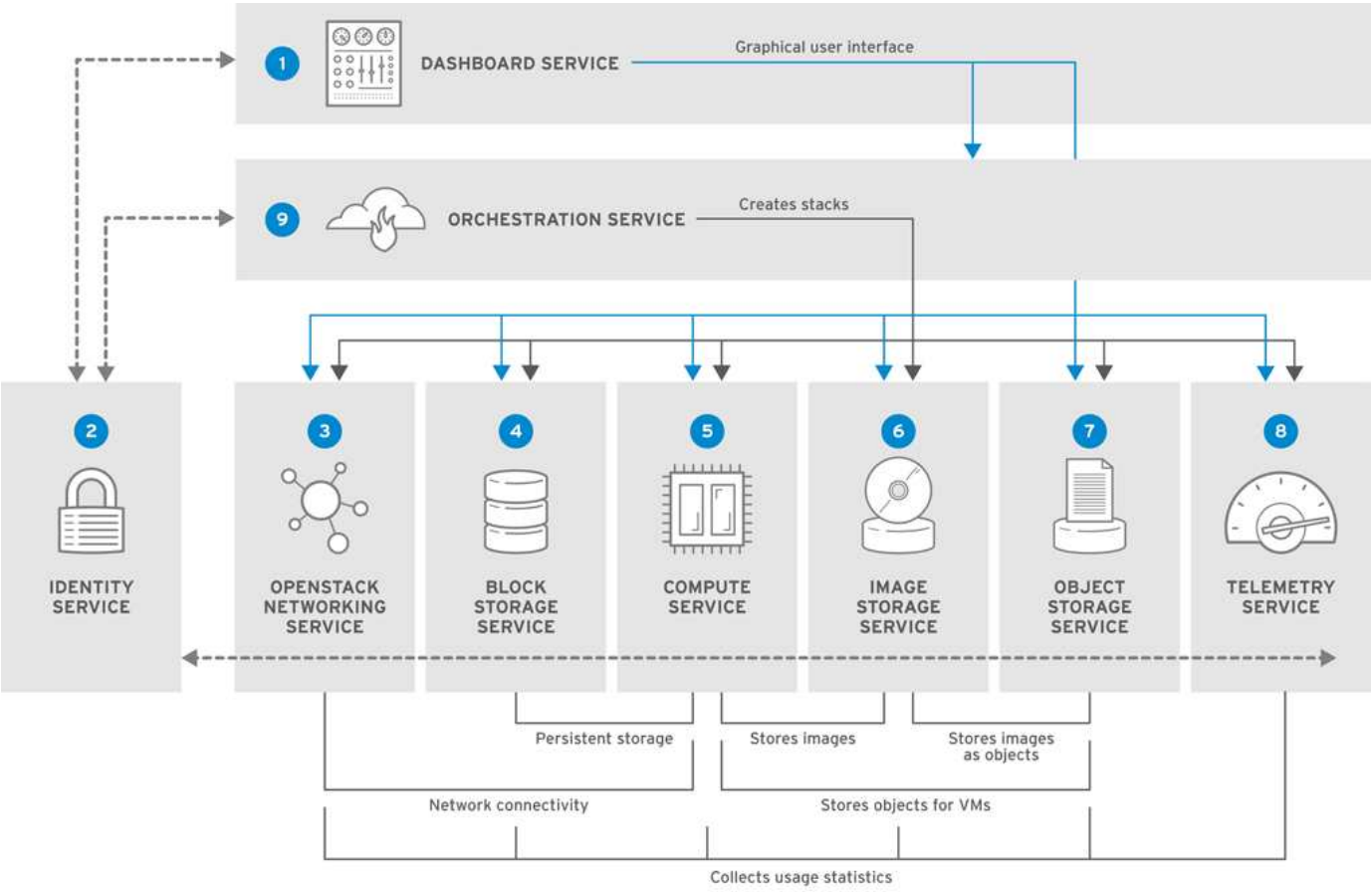
OSP è un cloud Infrastructure-as-a-service (IaaS) implementato da una raccolta di servizi di controllo che gestiscono risorse di calcolo, storage e networking. L'ambiente viene gestito tramite un'interfaccia basata su web che consente ad amministratori e utenti di controllare, eseguire il provisioning e automatizzare le risorse di OpenStack. Inoltre, l'infrastruttura OpenStack è facilitata da un'ampia interfaccia a riga di comando e API che consentono funzionalità di automazione complete per amministratori e utenti finali.

Il progetto OpenStack è un progetto della community sviluppato rapidamente che fornisce release aggiornate ogni sei mesi. Inizialmente Red Hat OpenStack Platform ha mantenuto il passo con questo ciclo di release pubblicando una nuova release insieme a ogni release upstream e fornendo supporto a lungo termine per ogni terza release. Di recente, con la release di OSP 16.0 (basata su OpenStack Train), Red Hat ha scelto di non tenere il passo con i numeri di release, ma ha invece trasferito le nuove funzionalità nelle subrelease. La versione più recente è Red Hat OpenStack Platform 16.1, che include funzionalità avanzate con backport delle release Usuri e Victoria in upstream.

Per ulteriori informazioni su OSP, vedere "Sito Web della piattaforma Red Hat OpenStack".

Servizi OpenStack

I servizi della piattaforma OpenStack vengono implementati come container, isolando i servizi l'uno dall'altro e consentendo facili aggiornamenti. La piattaforma OpenStack utilizza un set di container creati e gestiti con Kolla. L'implementazione dei servizi viene eseguita estraendo le immagini container dal Red Hat Custom Portal. Questi container di servizio vengono gestiti utilizzando il comando Podman e vengono implementati, configurati e gestiti con Red Hat OpenStack Director.



Servizio	Nome del progetto	Descrizione
Dashboard	Orizzonte	Dashboard basato su browser Web che consente di gestire i servizi OpenStack.
Identità	Keystone	Servizio centralizzato per l'autenticazione e l'autorizzazione dei servizi OpenStack e per la gestione di utenti, progetti e ruoli.
Networking OpenStack	Neutroni	Fornisce connettività tra le interfacce dei servizi OpenStack.
Storage a blocchi	Cinder	Gestisce volumi di storage a blocchi persistenti per macchine virtuali (VM).
Calcolo	Nova	Gestisce ed esegue il provisioning delle macchine virtuali in esecuzione sui nodi di calcolo.
Immagine	Panoramica	Servizio di registro utilizzato per memorizzare risorse come immagini di macchine virtuali e snapshot di volumi.
Storage a oggetti	Rapido	Consente agli utenti di memorizzare e recuperare file e dati arbitrari.
Telemetria	Ceilometro	Fornisce misurazioni dell'utilizzo delle risorse cloud.
Orchestrazione	Calore	Motore di orchestrazione basato su modelli che supporta la creazione automatica di stack di risorse.

Progettazione di rete

La soluzione Red Hat OpenShift con NetApp utilizza due switch dati per fornire connettività dati primaria a 25 Gbps. Utilizza inoltre due switch di gestione aggiuntivi che forniscono connettività a 1 Gbps per la gestione in-band dei nodi di storage e gestione out-of-band per la funzionalità IPMI.

Red Hat OpenStack Director richiede la funzionalità IPMI per implementare la piattaforma Red Hat OpenStack utilizzando il servizio di provisioning bare-metal ironico.

Requisiti VLAN

Red Hat OpenShift con NetApp è progettato per separare logicamente il traffico di rete per scopi diversi utilizzando Virtual Local Area Network (VLAN). Questa configurazione può essere scalata per soddisfare le esigenze dei clienti o per fornire un ulteriore isolamento per servizi di rete specifici. La seguente tabella elenca le VLAN necessarie per implementare la soluzione durante la convalida della soluzione in NetApp.

VLAN	Scopo	ID VLAN
Rete di gestione fuori banda	Rete utilizzata per la gestione dei nodi fisici e servizio IPMI per ironico.	16
Infrastruttura storage	Rete utilizzata per i nodi controller per mappare i volumi direttamente per supportare servizi di infrastruttura come Swift.	201
Storage Cinder	Rete utilizzata per mappare e collegare i volumi a blocchi direttamente alle istanze virtuali implementate nell'ambiente.	202
API interna	Rete utilizzata per la comunicazione tra i servizi OpenStack utilizzando la comunicazione API, i messaggi RPC e la comunicazione con il database.	301
Tenant	Neutron fornisce a ciascun tenant le proprie reti tramite il tunneling tramite VXLAN. Il traffico di rete viene isolato all'interno di ciascuna rete tenant. A ciascuna rete tenant è associata una subnet IP e gli spazi dei nomi di rete indicano che più reti tenant possono utilizzare lo stesso intervallo di indirizzi senza causare conflitti.	302
Gestione dello storage	OpenStack Object Storage (Swift) utilizza questa rete per sincronizzare gli oggetti dati tra i nodi di replica partecipanti. Il servizio proxy funge da interfaccia intermedia tra le richieste degli utenti e il livello di storage sottostante. Il proxy riceve le richieste in entrata e individua la replica necessaria per recuperare i dati richiesti.	303
PXE	OpenStack Director offre l'avvio PXE come parte dell'ironico servizio di provisioning bare metal per orchestrare l'installazione di OSP Overcloud.	3484
Esterno	Rete pubblicamente disponibile che ospita OpenStack Dashboard (Horizon) per la gestione grafica e consente alle chiamate API pubbliche di gestire i servizi OpenStack.	3485
Rete di gestione in-band	Fornisce l'accesso a funzioni di amministrazione del sistema come accesso SSH, traffico DNS e traffico NTP (Network Time Protocol). Questa rete funge anche da gateway per i nodi non controller.	3486

Risorse di supporto dell'infrastruttura di rete

Prima dell'implementazione della piattaforma container OpenShift, è necessario installare la seguente infrastruttura:

- Almeno un server DNS che fornisce una risoluzione completa del nome host.
- Almeno tre server NTP in grado di mantenere sincronizzato il tempo per i server della soluzione.
- (Opzionale) connettività Internet in uscita per l'ambiente OpenShift.

Best practice per le implementazioni in produzione

In questa sezione sono elencate diverse Best practice che un'organizzazione deve prendere in considerazione prima di implementare questa soluzione in produzione.

Implementa OpenShift su un cloud privato OSP con almeno tre nodi di calcolo

L'architettura verificata descritta in questo documento presenta l'implementazione hardware minima adatta per le operazioni ha implementando tre nodi controller OSP e due nodi di calcolo OSP. Questa architettura garantisce una configurazione a tolleranza di errore in cui entrambi i nodi di calcolo possono lanciare istanze virtuali e le macchine virtuali implementate possono migrare tra i due hypervisor.

Poiché Red Hat OpenShift inizialmente viene implementato con tre nodi master, una configurazione a due nodi potrebbe causare l'occupazione di almeno due master nello stesso nodo, il che può causare un'interruzione di OpenShift se tale nodo specifico non è disponibile. Pertanto, è una Best practice di Red Hat implementare almeno tre nodi di calcolo OSP in modo che i master OpenShift possano essere distribuiti in modo uniforme e la soluzione riceva un ulteriore livello di tolleranza agli errori.

Configurare l'affinità di macchine virtuali/host

La distribuzione dei master OpenShift tra più nodi hypervisor può essere ottenuta abilitando l'affinità VM/host.

Affinity è un modo per definire le regole per un insieme di macchine virtuali e/o host che determinano se le macchine virtuali vengono eseguite insieme sullo stesso host o su host del gruppo o su host diversi. Viene applicato alle macchine virtuali creando gruppi di affinità costituiti da macchine virtuali e/o host con un insieme di parametri e condizioni identici. A seconda che le macchine virtuali di un gruppo di affinità vengano eseguite sullo stesso host o su host del gruppo o separatamente su host diversi, i parametri del gruppo di affinità possono definire affinità positiva o affinità negativa. Nella piattaforma Red Hat OpenStack, è possibile creare e applicare le regole di affinità e anti-affinità degli host creando gruppi di server e configurando i filtri in modo che le istanze distribuite da Nova in un gruppo di server vengano distribuite su nodi di calcolo diversi.

Un gruppo di server dispone di un massimo predefinito di 10 istanze virtuali per le quali può gestire il posizionamento. È possibile modificare questa impostazione aggiornando le quote predefinite per Nova.

> Esiste un limite specifico di affinità/anti-affinità per i gruppi di server OSP; se non sono disponibili risorse sufficienti per l'implementazione su nodi separati o se non sono disponibili risorse sufficienti per consentire la condivisione dei nodi, la macchina virtuale non viene avviata.

Per configurare i gruppi di affinità, vedere "Come si configurano affinità e anti-affinità per le istanze di OpenStack?".

Utilizzare un file di installazione personalizzato per la distribuzione di OpenShift

IPI semplifica l'implementazione dei cluster OpenShift attraverso la procedura guidata interattiva descritta in precedenza in questo documento. Tuttavia, potrebbe essere necessario modificare alcuni valori predefiniti come parte di una distribuzione del cluster.

In questi casi, è possibile eseguire ed eseguire le procedure guidate senza implementare immediatamente un cluster; al contrario, viene creato un file di configurazione da cui il cluster può essere distribuito in un secondo momento. Questa funzione è molto utile se si desidera modificare le impostazioni predefinite dell'IPI o se si

desidera implementare più cluster identici nell'ambiente per altri utilizzi, ad esempio la multi-tenancy. Per ulteriori informazioni sulla creazione di una configurazione di installazione personalizzata per OpenShift, vedere "Red Hat OpenShift Installazione di un cluster su OpenStack con personalizzazioni".

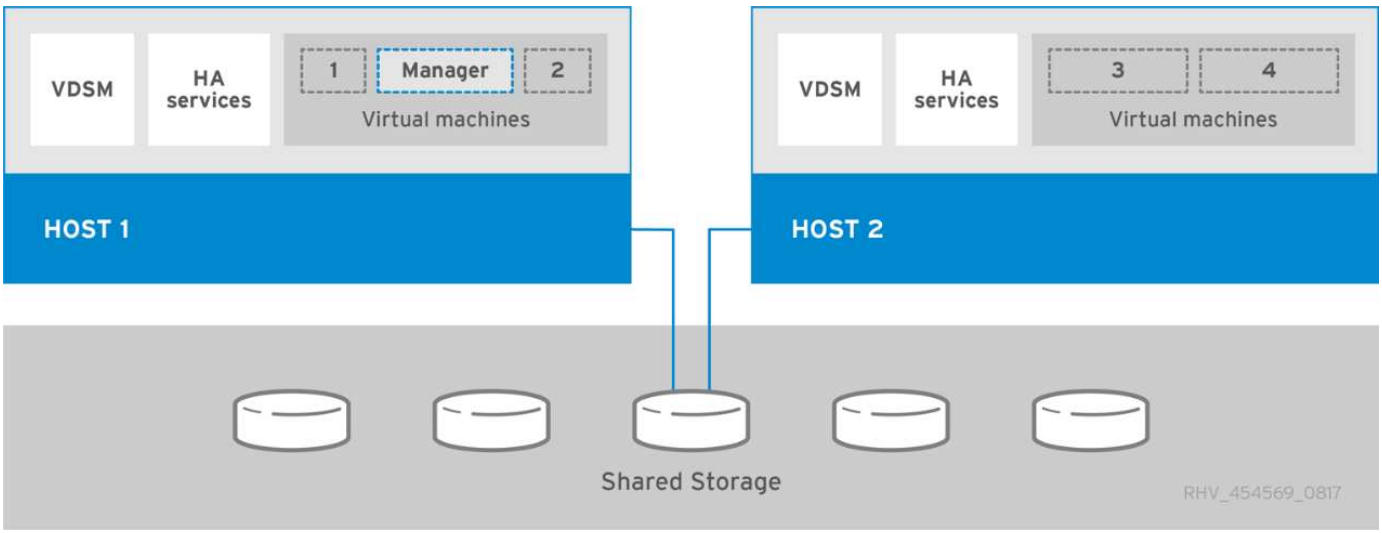
OpenShift sulla virtualizzazione Red Hat

Red Hat Virtualization (RHV) è una piattaforma per data center virtuale aziendale che viene eseguita su Red Hat Enterprise Linux (RHEL) e utilizza l'hypervisor KVM.

Per ulteriori informazioni su RHV, consultare "Sito Web Red Hat Virtualization".

RHV offre le seguenti funzionalità:

- **Gestione centralizzata di VM e host** il manager RHV viene eseguito come macchina fisica o virtuale (VM) nella distribuzione e fornisce una GUI basata sul Web per la gestione della soluzione da un'interfaccia centrale.
- **Motore auto-ospitato** per ridurre al minimo i requisiti hardware, RHV consente a RHV Manager (RHV-M) di essere installato come VM sugli stessi host che eseguono VM guest.
- **Alta disponibilità** per evitare interruzioni in caso di guasti all'host, RHV consente di configurare le VM per l'alta disponibilità. Le macchine virtuali ad alta disponibilità vengono controllate a livello di cluster utilizzando policy di resilienza.
- **Elevata scalabilità** Un singolo cluster RHV può avere fino a 200 host hypervisor che consentono al reparto IT di supportare i requisiti di macchine virtuali di grandi dimensioni per ospitare carichi di lavoro di classe Enterprise avidi di risorse.
- **La sicurezza avanzata** ereditata dalle tecnologie RHV, Secure Virtualization (sVirt) e Security Enhanced Linux (SELinux) è impiegata da RHV per scopi di elevata sicurezza e protezione avanzata per host e VM. Il vantaggio principale di queste funzionalità è l'isolamento logico di una macchina virtuale e delle risorse associate.



Progettazione di rete

La soluzione Red Hat OpenShift su NetApp utilizza due switch dati per fornire connettività dati primaria a 25 Gbps. Utilizza inoltre due switch di gestione aggiuntivi che forniscono connettività a 1 Gbps per la gestione in banda dei nodi di storage e gestione out-of-band per la funzionalità IPMI. OCP utilizza la rete logica della macchina virtuale su RHV per la gestione del cluster. Questa sezione descrive la disposizione e lo scopo di ciascun segmento di rete virtuale utilizzato nella soluzione e illustra i prerequisiti per l'implementazione della

soluzione.

Requisiti VLAN

Red Hat OpenShift su RHV è progettato per separare logicamente il traffico di rete per scopi diversi utilizzando Virtual Local Area Network (VLAN). Questa configurazione può essere scalata per soddisfare le esigenze dei clienti o per fornire un ulteriore isolamento per servizi di rete specifici. La seguente tabella elenca le VLAN necessarie per implementare la soluzione durante la convalida della soluzione in NetApp.

VLAN	Scopo	ID VLAN
Rete di gestione fuori banda	Gestione per nodi fisici e IPMI	16
Rete di macchine virtuali	Accesso alla rete guest virtuale	1172
Rete di gestione in-band	Gestione dei nodi RHV-H, RHV-Manager e della rete ovirtmgmt	3343
Rete di storage	Rete storage per iSCSI NetApp Element	3344
Rete di migrazione	Rete per la migrazione dei guest virtuali	3345

Risorse di supporto dell'infrastruttura di rete

Prima dell'implementazione della piattaforma container OpenShift, è necessario installare la seguente infrastruttura:

- Almeno un server DNS che fornisce una risoluzione completa del nome host accessibile dalla rete di gestione in-band e dalla rete VM.
- Almeno un server NTP accessibile dalla rete di gestione in-band e dalla rete VM.
- (Opzionale) connettività Internet in uscita per la rete di gestione in banda e la rete VM.

Best practice per le implementazioni in produzione

In questa sezione sono elencate diverse Best practice che un'organizzazione deve prendere in considerazione prima di implementare questa soluzione in produzione.

Implementare OpenShift in un cluster RHV di almeno tre nodi

L'architettura verificata descritta in questo documento presenta l'implementazione hardware minima adatta per le operazioni ha implementando due nodi hypervisor RHV-H e garantendo una configurazione a tolleranza di errore in cui entrambi gli host possono gestire il motore in hosting e le macchine virtuali implementate possono migrare tra i due hypervisor.

Poiché Red Hat OpenShift viene inizialmente implementato con tre nodi master, in una configurazione a due nodi è garantito che almeno due master occuperanno lo stesso nodo, il che può causare un'interruzione di OpenShift se quel nodo specifico non è disponibile. Pertanto, è una Best practice di Red Hat che almeno tre nodi di hypervisor RHV-H siano implementati come parte della soluzione, in modo che i master OpenShift possano essere distribuiti in modo uniforme e la soluzione riceva un ulteriore grado di tolleranza agli errori.

Configurare l'affinità di macchine virtuali/host

È possibile distribuire i master OpenShift su più nodi hypervisor abilitando l'affinità VM/host.

Affinity è un modo per definire le regole per un insieme di macchine virtuali e/o host che determinano se le macchine virtuali vengono eseguite insieme sullo stesso host o su host del gruppo o su host diversi. Viene

applicato alle macchine virtuali creando gruppi di affinità costituiti da macchine virtuali e/o host con un insieme di parametri e condizioni identici. A seconda che le macchine virtuali di un gruppo di affinità vengano eseguite sullo stesso host o su host del gruppo o separatamente su host diversi, i parametri del gruppo di affinità possono definire affinità positiva o affinità negativa.

Le condizioni definite per i parametri possono essere l'applicazione forzata o forzata. La rigida applicazione garantisce che le macchine virtuali di un gruppo di affinità seguano sempre l'affinità positiva o negativa rigorosamente senza alcun riferimento alle condizioni esterne. La soft enforcement garantisce che venga impostata una preferenza più elevata per le macchine virtuali di un gruppo di affinità per seguire l'affinità positiva o negativa quando possibile. Nella configurazione di due o tre hypervisor descritta in questo documento, l'impostazione consigliata è affinità soft. Nei cluster più grandi, la hard affinità può distribuire correttamente i nodi OpenShift.

Per configurare i gruppi di affinità, vedere "Red Hat 6.11. Documentazione di Affinity Groups".

Utilizzare un file di installazione personalizzato per la distribuzione di OpenShift

IPI semplifica l'implementazione dei cluster OpenShift attraverso la procedura guidata interattiva descritta in precedenza in questo documento. Tuttavia, è possibile che alcuni valori predefiniti debbano essere modificati nell'ambito dell'implementazione del cluster.

In questi casi, è possibile eseguire e gestire la procedura guidata senza implementare immediatamente un cluster. Viene invece creato un file di configurazione da cui è possibile implementare il cluster in un secondo momento. Questo è molto utile se si desidera modificare le impostazioni predefinite IPI o se si desidera implementare più cluster identici nel proprio ambiente per altri utilizzi, ad esempio la multi-tenancy. Per ulteriori informazioni sulla creazione di una configurazione di installazione personalizzata per OpenShift, vedere "Red Hat OpenShift Installazione di un cluster su RHV con personalizzazioni".

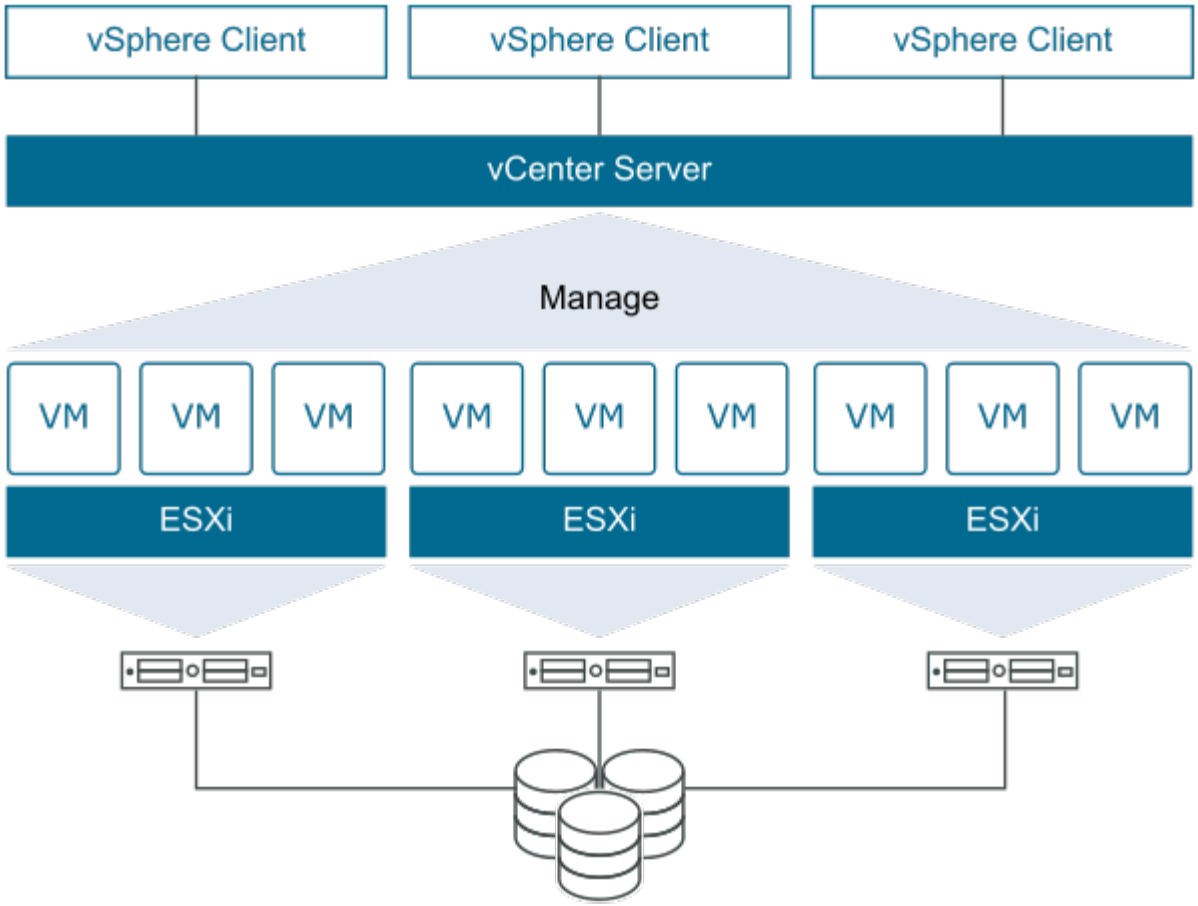
OpenShift su VMware vSphere

VMware vSphere è una piattaforma di virtualizzazione per la gestione centralizzata di un gran numero di server e reti virtualizzati in esecuzione sull'hypervisor ESXi.

Per ulteriori informazioni su VMware vSphere, consultare "Sito Web di VMware vSphere".

VMware vSphere offre le seguenti funzionalità:

- **VMware vCenter Server** VMware vCenter Server offre una gestione unificata di tutti gli host e le macchine virtuali da un'unica console e aggrega il monitoraggio delle prestazioni di cluster, host e macchine virtuali.
- **VMware vSphere vMotion** VMware vCenter consente di eseguire la migrazione a caldo delle macchine virtuali tra i nodi del cluster su richiesta in modo non disgregativo.
- **VSphere High Availability** per evitare interruzioni in caso di errori dell'host, VMware vSphere consente di raggruppare gli host e di configurarli per l'alta disponibilità. Le macchine virtuali che vengono interrompite da un guasto dell'host vengono riavviati a breve su altri host del cluster, ripristinando i servizi.
- **Distributed Resource Scheduler (DRS)** È possibile configurare Un cluster VMware vSphere per bilanciare il carico delle risorse richieste dalle VM che ospita. È possibile eseguire la migrazione a caldo delle macchine virtuali con risorse in altri nodi del cluster per assicurarsi che siano disponibili risorse sufficienti.



Progettazione di rete

La soluzione Red Hat OpenShift su NetApp utilizza due switch dati per fornire connettività dati primaria a 25 Gbps. Utilizza inoltre due switch di gestione aggiuntivi che forniscono connettività a 1 Gbps per la gestione in-band dei nodi di storage e gestione out-of-band per la funzionalità IPMI. OCP utilizza la rete logica VM su VMware vSphere per la gestione del cluster. Questa sezione descrive la disposizione e lo scopo di ciascun segmento di rete virtuale utilizzato nella soluzione e illustra i prerequisiti per l'implementazione della soluzione.

Requisiti VLAN

Red Hat OpenShift su VMware vSphere è progettato per separare logicamente il traffico di rete per scopi diversi utilizzando Virtual Local Area Network (VLAN). Questa configurazione può essere scalata per soddisfare le esigenze dei clienti o per fornire un ulteriore isolamento per servizi di rete specifici. La seguente tabella elenca le VLAN necessarie per implementare la soluzione durante la convalida della soluzione in NetApp.

VLAN	Scopo	ID VLAN
Rete di gestione fuori banda	Gestione per nodi fisici e IPMI	16
Rete di macchine virtuali	Accesso alla rete guest virtuale	181
Rete di storage	Rete di storage per NFS ONTAP	184
Rete di storage	Rete storage per iSCSI ONTAP	185
Rete di gestione in-band	Gestione per nodi ESXi, server vCenter, ONTAP Select	3480

VLAN	Scopo	ID VLAN
Rete di storage	Rete storage per iSCSI NetApp Element	3481
Rete di migrazione	Rete per la migrazione dei guest virtuali	3482

Risorse di supporto dell'infrastruttura di rete

Prima dell'implementazione della piattaforma container OpenShift, è necessario installare la seguente infrastruttura:

- Almeno un server DNS che fornisce una risoluzione completa del nome host accessibile dalla rete di gestione in-band e dalla rete VM.
- Almeno un server NTP accessibile dalla rete di gestione in-band e dalla rete VM.
- (Opzionale) connettività Internet in uscita per la rete di gestione in banda e la rete VM.

Best practice per le implementazioni in produzione

In questa sezione sono elencate diverse Best practice che un'organizzazione deve prendere in considerazione prima di implementare questa soluzione in produzione.

Implementare OpenShift in un cluster ESXi di almeno tre nodi

L'architettura verificata descritta in questo documento presenta l'implementazione hardware minima adatta per le operazioni ha implementando due nodi hypervisor ESXi e garantendo una configurazione con tolleranza di errore abilitando VMware vSphere ha e VMware vMotion. Questa configurazione consente alle macchine virtuali implementate di migrare tra i due hypervisor e di riavviare il sistema in caso di mancata disponibilità di un host.

Poiché Red Hat OpenShift inizialmente viene implementato con tre nodi master, almeno due master in una configurazione a due nodi possono occupare lo stesso nodo in alcune circostanze, il che può causare un'interruzione di OpenShift se quel nodo specifico non è disponibile. Pertanto, è una Best practice di Red Hat che devono essere implementati almeno tre nodi di hypervisor ESXi in modo che i master OpenShift possano essere distribuiti in modo uniforme, fornendo un ulteriore grado di tolleranza agli errori.

Configurare l'affinità della macchina virtuale e dell'host

Garantire la distribuzione dei master OpenShift su più nodi hypervisor abilitando l'affinità di macchine virtuali e host.

Affinità o anti-affinità è un metodo per definire le regole per un insieme di macchine virtuali e/o host che determinano se le macchine virtuali vengono eseguite insieme sullo stesso host o su host del gruppo o su host diversi. Viene applicato alle macchine virtuali creando gruppi di affinità costituiti da macchine virtuali e/o host con un insieme di parametri e condizioni identici. A seconda che le macchine virtuali di un gruppo di affinità vengano eseguite sullo stesso host o su host del gruppo o separatamente su host diversi, i parametri del gruppo di affinità possono definire affinità positiva o affinità negativa.

Per configurare i gruppi di affinità, vedere "Documentazione vSphere 6.7: Utilizzo delle regole di affinità DRS".

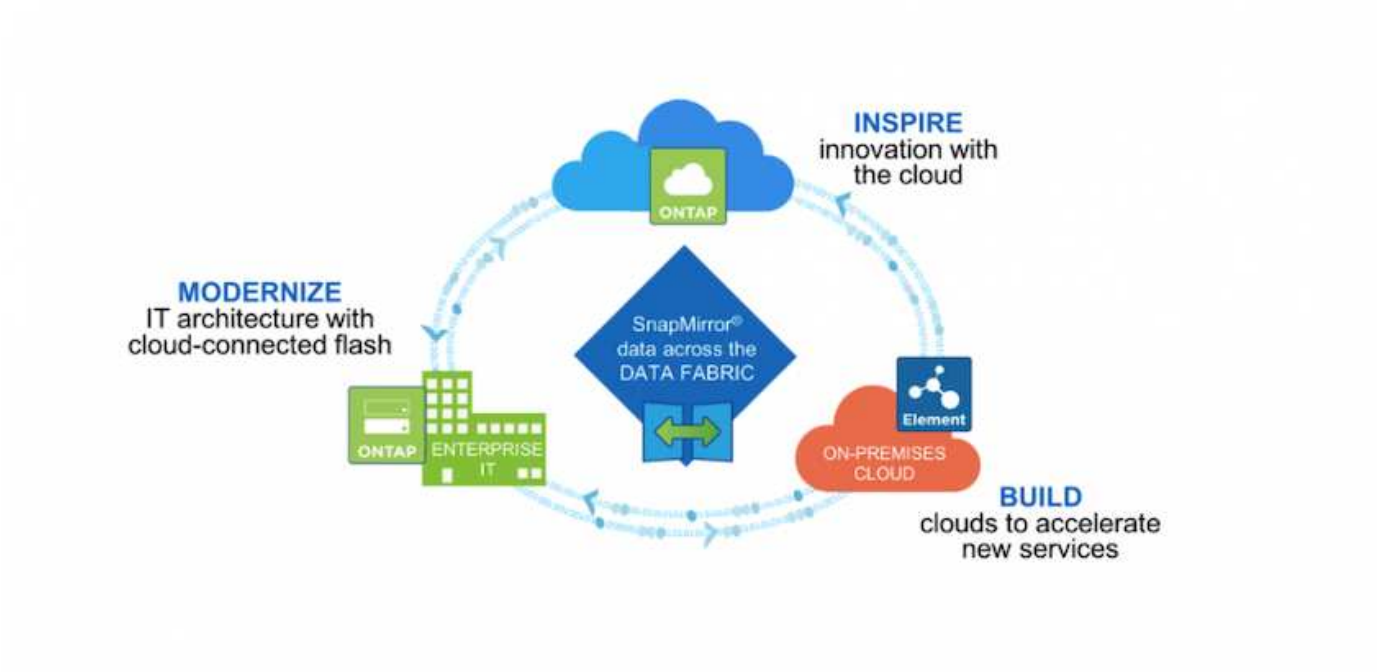
Utilizzare un file di installazione personalizzato per la distribuzione di OpenShift

IPI semplifica l'implementazione dei cluster OpenShift attraverso la procedura guidata interattiva descritta in precedenza in questo documento. Tuttavia, potrebbe essere necessario modificare alcuni valori predefiniti come parte di una distribuzione del cluster.

In questi casi, è possibile eseguire la procedura guidata senza implementare immediatamente un cluster, ma la procedura guidata crea un file di configurazione da cui il cluster può essere distribuito in un secondo momento. Questa funzione è molto utile se si desidera modificare le impostazioni predefinite dell'IPI o se si desidera implementare più cluster identici nell'ambiente per altri utilizzi, ad esempio la multi-tenancy. Per ulteriori informazioni sulla creazione di una configurazione di installazione personalizzata per OpenShift, vedere "Red Hat OpenShift Installazione di un cluster su vSphere con personalizzazioni".

Panoramica dello storage NetApp

NetApp dispone di diverse piattaforme di storage qualificate con Astra Trident Storage Orchestrator per il provisioning dello storage per le applicazioni implementate su Red Hat OpenShift.



- I sistemi AFF e FAS eseguono NetApp ONTAP e forniscono storage per i casi di utilizzo basati su file (NFS) e basati su blocchi (iSCSI).
- Cloud Volumes ONTAP e ONTAP Select offrono gli stessi vantaggi rispettivamente nel cloud e nello spazio virtuale.
- NetApp Cloud Volumes Service (AWS/GCP) e Azure NetApp Files offrono storage basato su file nel cloud.
- I sistemi storage NetApp Element offrono casi di utilizzo basati su blocchi (iSCSI) in un ambiente altamente scalabile.

> Ogni sistema storage del portfolio NetApp può semplificare la gestione dei dati e lo spostamento tra i siti on-premise e il cloud, garantendo che i dati si trovino nella posizione in cui si trovano le applicazioni.

Le pagine seguenti contengono informazioni aggiuntive sui sistemi di storage NetApp validati nella soluzione Red Hat OpenShift con NetApp:

- "NetApp ONTAP"
- "NetApp Element"

NetApp ONTAP

NetApp ONTAP è un potente tool software per lo storage con funzionalità come GUI intuitiva, API REST con integrazione dell'automazione, analisi predittive e azioni correttive informate dell'ai, aggiornamenti hardware senza interruzioni e importazione di storage incrociato.

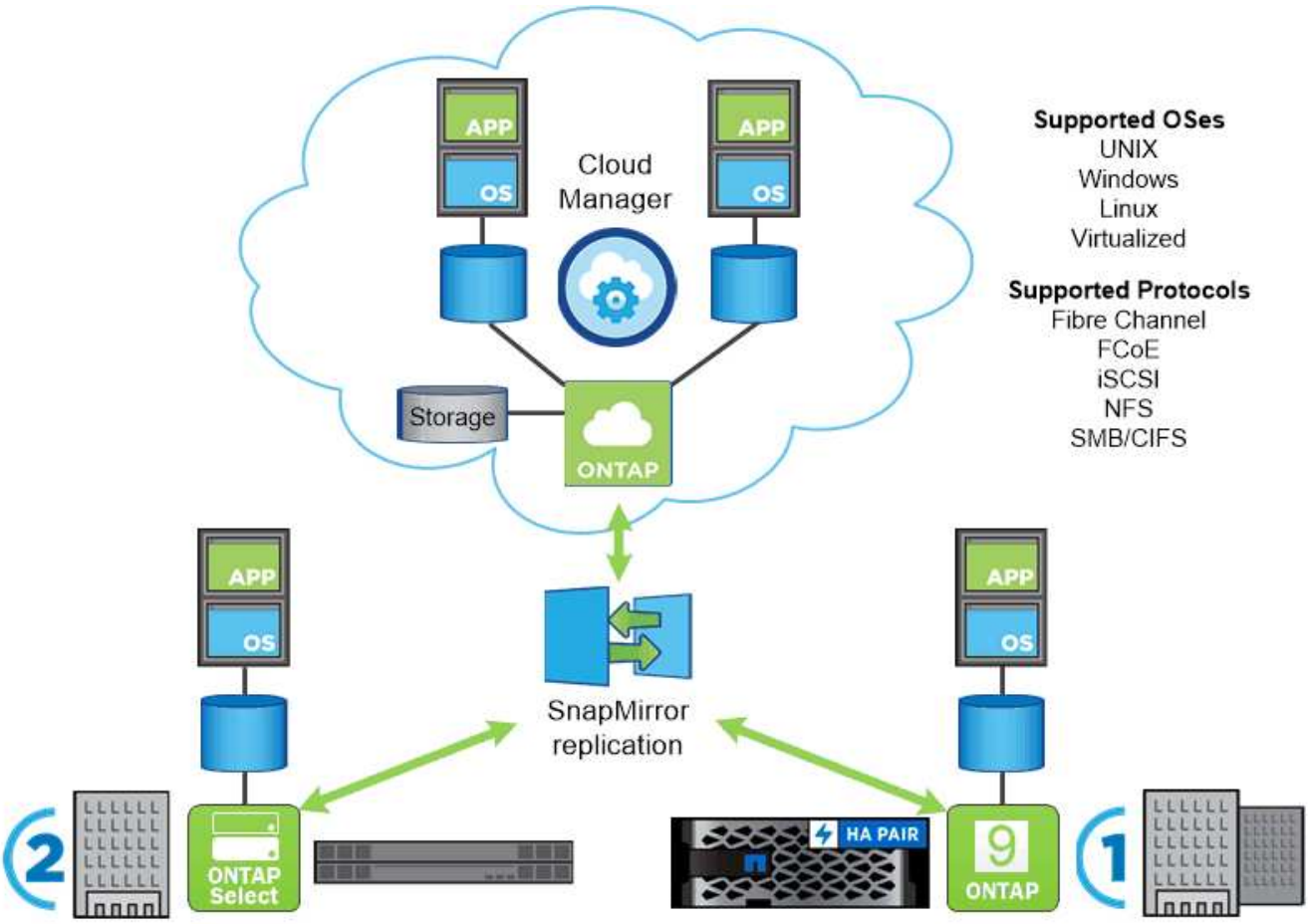
Per ulteriori informazioni sul sistema di storage NetApp ONTAP, visitare il sito "Sito Web di NetApp ONTAP".

ONTAP offre le seguenti funzionalità:

- Un sistema storage unificato con accesso e gestione simultanei dei dati di NFS, CIFS, iSCSI, FC, FCoE, E protocolli FC-NVMe.
- Diversi modelli di implementazione includono configurazioni hardware on-premise su all-flash, ibride e all-HDD, piattaforme di storage basate su VM su un hypervisor supportato come ONTAP Select e nel cloud come Cloud Volumes ONTAP.
- Maggiore efficienza dello storage dei dati sui sistemi ONTAP con supporto per tiering automatico dei dati, compressione dei dati inline, deduplica e compaction.
- Storage basato su workload e controllato dalla QoS.
- Integrazione perfetta con un cloud pubblico per tiering e protezione dei dati. ONTAP offre inoltre solide funzionalità di protezione dei dati che lo differenziano in qualsiasi ambiente:
  - **NetApp Snapshot Copies.** Backup rapido e point-in-time dei dati utilizzando una quantità minima di spazio su disco senza alcun overhead delle performance aggiuntivo.
  - **NetApp SnapMirror.** Mirror le copie Snapshot dei dati da un sistema storage a un altro. ONTAP supporta il mirroring dei dati su altre piattaforme fisiche e servizi nativi del cloud.
  - **NetApp SnapLock.** Amministrazione efficiente dei dati non riscrivibili, scrivendo su volumi speciali che non possono essere sovrascritti o cancellati per un determinato periodo.
  - **NetApp SnapVault.** esegue il backup dei dati da più sistemi storage in una copia Snapshot centrale che funge da backup su tutti i sistemi designati.
  - **NetApp SyncMirror.** offre mirroring dei dati in tempo reale a livello RAID su due diversi plessi di dischi collegati fisicamente allo stesso controller.
  - **NetApp SnapRestore.** offre un rapido ripristino dei dati di backup on-demand dalle copie Snapshot.
  - **NetApp FlexClone.** fornisce il provisioning istantaneo di una copia leggibile e scrivibile di un volume NetApp basata su una copia Snapshot.

Per ulteriori informazioni su ONTAP, consultare "Centro documentazione di ONTAP 9".

NetApp ONTAP è disponibile on-premise, virtualizzato o nel cloud.



Piattaforme NetApp

NetApp AFF/FAS

NetApp offre solide piattaforme di storage all-flash (AFF) e ibride scale-out (FAS), realizzate su misura con performance a bassa latenza, protezione integrata dei dati e supporto multiprotocollo.

Entrambi i sistemi sono basati sul software per la gestione dei dati NetApp ONTAP, il software per la gestione dei dati più avanzato del settore per una gestione dello storage semplificata, integrata nel cloud e ad alta disponibilità, in grado di offrire velocità, efficienza e sicurezza di livello Enterprise di cui ha bisogno il data fabric.

Per ulteriori informazioni sulle piattaforme NETAPP AFF/FAS, fare clic su "qui".

ONTAP Select

ONTAP Select è un'implementazione software-defined di NetApp ONTAP che può essere implementata su un hypervisor nel tuo ambiente. Può essere installato su VMware vSphere o su KVM e offre tutte le funzionalità e l'esperienza di un sistema ONTAP basato su hardware.

Per ulteriori informazioni su ONTAP Select, fare clic su "qui".

Cloud Volumes ONTAP

NetApp Cloud Volumes ONTAP è una versione di NetApp ONTAP implementata nel cloud e disponibile per

l'implementazione in diversi cloud pubblici, tra cui: Amazon AWS, Microsoft Azure e Google Cloud.

Per ulteriori informazioni su Cloud Volumes ONTAP, fare clic su "qui".

NetApp Element: Red Hat OpenShift con NetApp

Il software NetApp Element offre performance modulari e scalabili, con ogni nodo di storage che offre capacità e throughput garantiti all'ambiente. I sistemi NetApp Element possono scalare da 4 a 100 nodi in un singolo cluster e offrire una serie di funzionalità avanzate di gestione dello storage.

Per ulteriori informazioni sui sistemi di storage NetApp Element, visitare il sito "Sito Web di NetApp SolidFire".

Reindirizzamento dell'accesso iSCSI e funzionalità di riparazione automatica

Il software NetApp Element sfrutta il protocollo di storage iSCSI, un metodo standard per incapsulare i comandi SCSI su una rete TCP/IP tradizionale. Quando gli standard SCSI cambiano o quando le performance delle reti Ethernet migliorano, il protocollo di storage iSCSI beneficia senza la necessità di modifiche.

Sebbene tutti i nodi storage dispongano di un IP di gestione e di un IP di storage, il software NetApp Element annuncia un singolo indirizzo IP virtuale di storage (indirizzo SVIP) per tutto il traffico di storage nel cluster. Come parte del processo di accesso iSCSI, lo storage può rispondere che il volume di destinazione è stato spostato in un indirizzo diverso e quindi non può procedere con il processo di negoziazione. L'host quindi invia nuovamente la richiesta di accesso al nuovo indirizzo in un processo che non richiede alcuna riconfigurazione sul lato host. Questo processo è noto come reindirizzamento dell'accesso iSCSI.

Il reindirizzamento dell'accesso iSCSI è una parte chiave del cluster software NetApp Element. Quando viene ricevuta una richiesta di accesso all'host, il nodo decide quale membro del cluster deve gestire il traffico in base agli IOPS e ai requisiti di capacità per il volume. I volumi vengono distribuiti nel cluster software NetApp Element e ridistribuiti se un singolo nodo gestisce un volume eccessivo di traffico per i volumi o se viene aggiunto un nuovo nodo. Più copie di un determinato volume vengono allocate nell'array.

In questo modo, se un guasto di un nodo è seguito da una ridistribuzione del volume, non vi è alcun effetto sulla connettività dell'host oltre a una disconnessione e all'accesso con reindirizzamento alla nuova posizione. Con il reindirizzamento dell'accesso iSCSI, un cluster software NetApp Element è un'architettura scale-out con riparazione automatica in grado di eseguire operazioni e aggiornamenti senza interruzioni.

QoS del cluster software NetApp Element

Un cluster software NetApp Element consente di configurare dinamicamente la qualità del servizio in base al volume. È possibile utilizzare le impostazioni QoS per volume per controllare le performance dello storage in base agli SLA definiti dall'utente. I seguenti tre parametri configurabili definiscono la QoS:

- **IOPS minimi.** il numero minimo di IOPS sostenuti che il cluster software NetApp Element fornisce a un volume. Il livello minimo di IOPS configurato per un volume è il livello garantito di performance per un

volume. Le performance per volume non scendono al di sotto di questo livello.

- **IOPS massimo.** numero massimo di IOPS sostenuti che il cluster software NetApp Element fornisce a un determinato volume.
- **IOPS burst.** numero massimo di IOPS consentito in uno scenario a burst breve. L'impostazione della durata del burst è configurabile, con un valore predefinito di 1 minuto. Se un volume è stato eseguito al di sotto del livello IOPS massimo, vengono accumulati i crediti burst. Quando i livelli di performance diventano molto elevati e vengono spinti, sul volume sono consentiti brevi burst di IOPS oltre i massimi IOPS.

Multi-tenancy

La multi-tenancy sicura si ottiene con le seguenti funzionalità:

- **Autenticazione sicura.** il protocollo CHAP (Challenge-Handshake Authentication Protocol) viene utilizzato per l'accesso sicuro ai volumi. Il protocollo LDAP (Lightweight Directory Access Protocol) viene utilizzato per l'accesso sicuro al cluster per la gestione e la creazione di report.
- **Volume Access Group (VAG).** facoltativamente, i VAG possono essere utilizzati al posto dell'autenticazione, mappando qualsiasi numero di iSCSI Initiator-Specific iSCSI Qualified Name (IQN) in uno o più volumi. Per accedere a un volume in un VAG, l'IQN dell'iniziatore deve essere nell'elenco IQN consentito per il gruppo di volumi.
- **LAN virtuali tenant (VLAN).** a livello di rete, la sicurezza di rete end-to-end tra gli iniziatori iSCSI e il cluster software NetApp Element è facilitata dall'utilizzo di VLAN. Per qualsiasi VLAN creata per isolare un carico di lavoro o un tenant, il software NetApp Element crea un indirizzo SVIP di destinazione iSCSI separato accessibile solo attraverso la VLAN specifica.
- **VLAN abilitate per VRF.** per supportare ulteriormente la sicurezza e la scalabilità nel data center, il software NetApp Element consente di abilitare qualsiasi VLAN tenant per funzionalità simili a VRF. Questa funzionalità aggiunge queste due funzionalità chiave:
  - Routing **L3 a un indirizzo SVIP tenant.** questa funzione consente di posizionare gli iniziatori iSCSI su una rete o VLAN separata da quella del cluster software NetApp Element.
  - **Subnet IP sovrapposte o duplicate.** questa funzione consente di aggiungere un modello agli ambienti tenant, consentendo a ciascuna VLAN tenant di essere assegnata a indirizzi IP della stessa subnet IP. Questa funzionalità può essere utile per gli ambienti dei provider di servizi in cui la scalabilità e la conservazione di IPSpace sono importanti.

Efficienze dello storage aziendale

Il cluster software NetApp Element aumenta l'efficienza e le performance generali dello storage. Le seguenti funzioni vengono eseguite inline, sono sempre attive e non richiedono alcuna configurazione manuale da parte dell'utente:

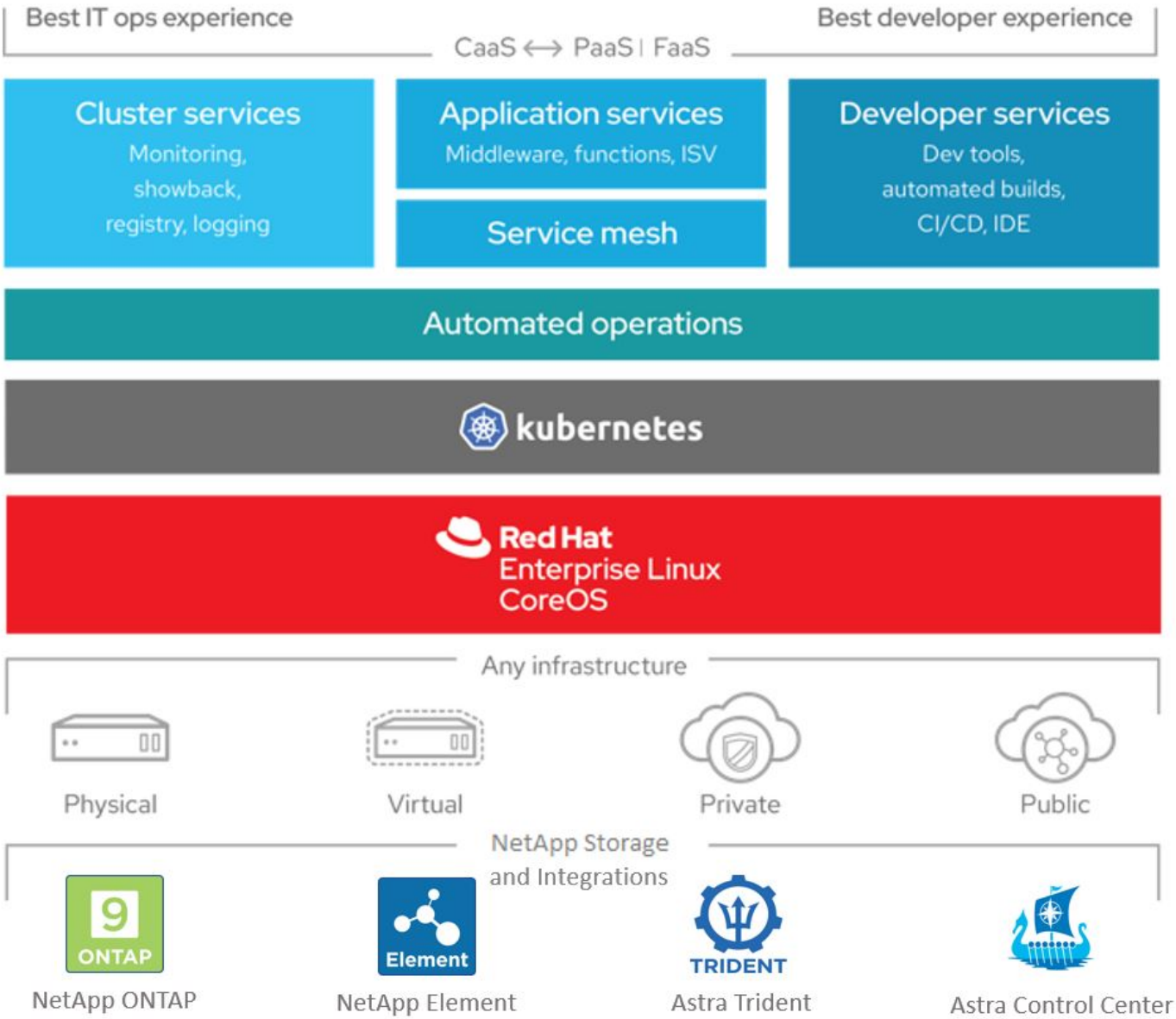
- **Deduplica.** il sistema memorizza solo blocchi 4K univoci. Tutti i blocchi 4K duplicati vengono automaticamente associati a una versione dei dati già memorizzata. I dati si trovano su dischi a blocchi e vengono mirrorati utilizzando la protezione dei dati del software NetApp Element Helix. Questo sistema riduce significativamente il consumo di capacità e le operazioni di scrittura all'interno del sistema.
- **Compressione.** la compressione viene eseguita inline prima che i dati vengano scritti nella NVRAM. I dati vengono compressi, memorizzati in blocchi 4K e rimangono compressi nel sistema. Questa compressione riduce significativamente il consumo di capacità, le operazioni di scrittura e il consumo di larghezza di banda nel cluster.
- **Thin-provisioning.** questa funzionalità fornisce la giusta quantità di storage al momento necessario, eliminando il consumo di capacità causato da volumi con overprovisioning o volumi sottoutilizzati.

- **Helix.** i metadati di un singolo volume vengono memorizzati su un'unità di metadati e replicati su un'unità di metadati secondaria per la ridondanza.

> Element è stato progettato per l'automazione. Tutte le funzionalità di storage sono disponibili tramite API. Queste API sono l'unico metodo utilizzato dall'interfaccia utente per controllare il sistema.

Panoramica sull'integrazione dello storage NetApp

NetApp offre una serie di prodotti per aiutarvi nell'orchestrazione e nella gestione dei dati persistenti in ambienti basati su container, come Red Hat OpenShift.



NetApp Astra Control offre un set completo di servizi di gestione dei dati application-aware e storage per carichi di lavoro Kubernetes stateful, basati sulla tecnologia di protezione dei dati di NetApp. Astra Control Service è disponibile per supportare carichi di lavoro stateful nelle implementazioni Kubernetes native nel cloud. Astra Control Center è disponibile per supportare carichi di lavoro stateful in implementazioni on-premise, come Red Hat OpenShift. Per ulteriori informazioni, visita il sito Web di NetApp Astra Control "qui".

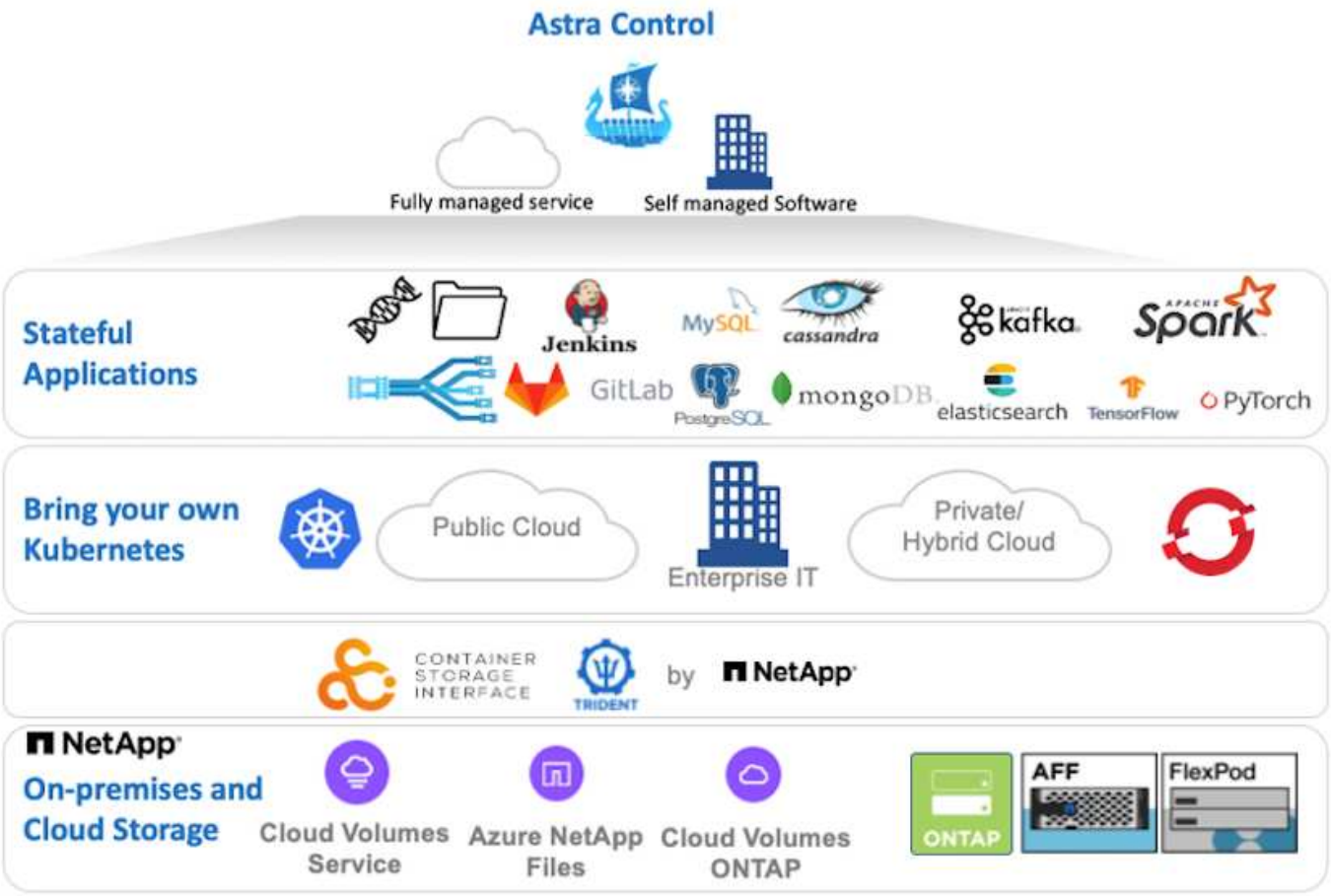
NetApp Astra Trident è un orchestrator di storage open-source e completamente supportato per container e distribuzioni Kubernetes, tra cui Red Hat OpenShift. Per ulteriori informazioni, visita il sito web di Astra Trident "qui".

Le pagine seguenti contengono informazioni aggiuntive sui prodotti NetApp validati per la gestione delle applicazioni e dello storage persistente nella soluzione Red Hat OpenShift con NetApp:

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

Panoramica di NetApp Astra Control Center

NetApp Astra Control Center offre un'ampia gamma di servizi di gestione dei dati basati su applicazioni e storage per carichi di lavoro Kubernetes stateful implementati in un ambiente on-premise e basati sulla tecnologia di protezione dei dati di NetApp.



È possibile installare NetApp Astra Control Center su un cluster Red Hat OpenShift che dispone di Astra Trident Storage orchestrator implementato e configurato con classi di storage e backend di storage per i sistemi storage NetApp ONTAP.

Per l'installazione e la configurazione di Astra Trident per il supporto di Astra Control Center, vedere "questo documento qui".

In un ambiente connesso al cloud, il centro di controllo Astra utilizza Cloud Insights per fornire monitoraggio avanzato e telemetria. In assenza di una connessione Cloud Insights, sono disponibili funzioni limitate di monitoraggio e telemetria (7 giorni di metriche) ed esportate negli strumenti di monitoraggio nativi di

Kubernetes (Prometheus e Grafana) attraverso endpoint di metriche aperte.

Il centro di controllo Astra è completamente integrato nell'ecosistema NetApp AutoSupport e Active IQ per fornire supporto agli utenti, fornire assistenza per la risoluzione dei problemi e visualizzare le statistiche di utilizzo.

Oltre alla versione a pagamento di Astra Control Center, è disponibile una licenza di valutazione di 90 giorni. La versione di valutazione è supportata tramite e-mail e community (canale slack). I clienti hanno accesso a questi e ad altri articoli della Knowledge base e alla documentazione disponibile nella dashboard di supporto dei prodotti.

Per iniziare a utilizzare NetApp Astra Control Center, visita il "Sito web Astra".

Prerequisiti per l'installazione di Astra Control Center

1. Uno o più cluster Red Hat OpenShift. Le versioni 4.6 EUS e 4.7 sono attualmente supportate.
2. Astra Trident deve essere già installato e configurato su ogni cluster Red Hat OpenShift.
3. Uno o più sistemi storage NetApp ONTAP con ONTAP 9.5 o superiore.

   > Per ogni installazione di OpenShift in un sito è consigliabile disporre di una SVM dedicata per lo storage persistente. Le implementazioni multi-sito richiedono sistemi storage aggiuntivi.

4. È necessario configurare un backend di storage Trident su ciascun cluster OpenShift con una SVM supportata da un cluster ONTAP.
5. StorageClass predefinita configurata su ciascun cluster OpenShift con Astra Trident come storage provisioning.
6. È necessario installare e configurare un bilanciamento del carico su ciascun cluster OpenShift per il bilanciamento del carico e l'esposizione dei servizi OpenShift.

   > Vedere il link "qui" per informazioni sui bilanciatori di carico validati per questo scopo.

7. È necessario configurare un registro di immagini privato per ospitare le immagini di NetApp Astra Control Center.

   > Vedere il link "qui" Per installare e configurare un registro privato OpenShift a tale scopo.

8. È necessario disporre dell'accesso Cluster Admin al cluster Red Hat OpenShift.
9. È necessario disporre dell'accesso come amministratore ai cluster NetApp ONTAP.
10. Una workstation di amministrazione con i tool docker o podman, tridentctl e oc o kubectl installati e aggiunti al percorso dei dollari.

    > Le installazioni di Docker devono avere una versione di Docker superiore alla 20.10 e le installazioni di Podman devono avere una versione di podman superiore alla 3.0.

Installare Astra Control Center

Utilizzo di OperatorHub

1. Accedere al NetApp Support Site e scaricare l'ultima versione di NetApp Astra Control Center. Per farlo, è necessaria una licenza allegata al tuo account NetApp. Dopo aver scaricato il tarball, trasferirlo sulla workstation di amministrazione.

   > Per iniziare a utilizzare una licenza di prova per Astra Control, visitare il sito "Sito di registrazione Astra".

2. Disimballare il tar ball e modificare la directory di lavoro nella cartella risultante.

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

3. Prima di iniziare l'installazione, trasferire le immagini di Astra Control Center in un registro di immagini. Puoi scegliere di farlo con Docker o Podman; in questo passaggio vengono fornite le istruzioni per entrambi.

Podman

a. Esportare ‘reFQDN del Registro di sistema con il nome dell’organizzazione/namespace/progetto come variabile di ambiente 'gistry'.

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

b. Accedere al Registro di sistema.

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

> Se si utilizza `kubeadmin` utente per accedere al registro privato, quindi utilizzare il token invece della password - `podman login -u ocp-user -p token --tls-verify=false astra-registry.apps.ocp-vmw.cie.netapp.com`.

> In alternativa, è possibile creare un account di servizio, assegnare un ruolo di editor del Registro di sistema e/o di visualizzatore del Registro di sistema (a seconda che si richieda l’accesso push/pull) e accedere al Registro di sistema utilizzando il token dell’account di servizio.

c. Creare un file script della shell e incollarne il contenuto seguente.

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

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

> Se si utilizzano certificati non attendibili per il Registro di sistema, modificare lo script della shell e utilizzare `--tls-verify=false` per il comando podman push `podman push $REGISTRY/$(echo $astraImage | sed 's/[\/]\+\///') --tls-verify=false`.

d. Rendere il file eseguibile.

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

e. Eseguire lo script della shell.

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

Docker

a. Esportare ‘reFQDN del Registro di sistema con il nome dell’organizzazione/namespace/progetto come variabile di ambiente 'gistry'.

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

b. Accedere al Registro di sistema.

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

> Se si utilizza `kubeadmin` utente per accedere al registro privato, quindi utilizzare il token invece della password - `docker login -u ocp-user -p token astra-registry.apps.ocp-vmw.cie.netapp.com`.

> In alternativa, è possibile creare un account di servizio, assegnare un ruolo di editor del Registro di sistema e/o di visualizzatore del Registro di sistema (a seconda che si richieda l’accesso push/pull) e accedere al Registro di sistema utilizzando il token dell’account di servizio.

c. Creare un file script della shell e incollarne il contenuto seguente.

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

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

d. Rendere il file eseguibile.

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

e. Eseguire lo script della shell.

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

4. Quando si utilizzano registri di immagini private non pubblicamente attendibili, caricare i certificati TLS del registro di immagini nei nodi OpenShift. A tale scopo, creare una configurazione nello spazio dei nomi openshift-config utilizzando i certificati TLS e applicarla alla configurazione dell'immagine del cluster per rendere attendibile il certificato.

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

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

> Se si utilizza un registro interno di OpenShift con certificati TLS predefiniti dall'operatore di ingresso con un percorso, è comunque necessario seguire la procedura precedente per applicare la patch ai certificati con il nome host del percorso. Per estrarre i certificati dall'operatore di ingresso, è possibile utilizzare il comando `oc extract secret/router-ca --keys=tls.crt -n openshift-ingress-operator`.

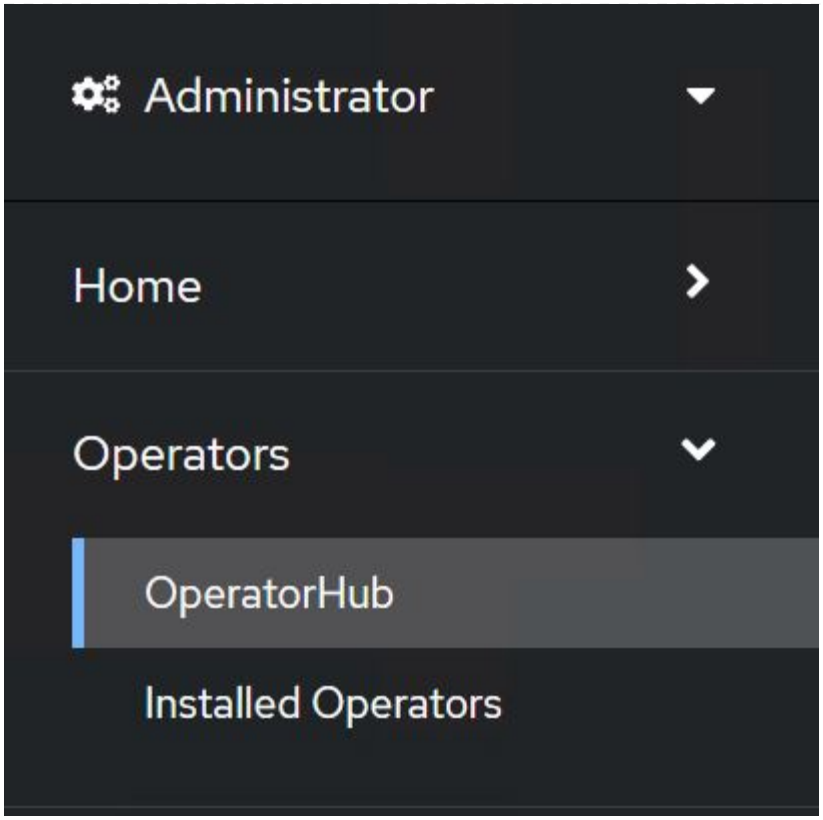
5. Creare uno spazio dei nomi `netapp-acc-operator` Per Astra Control Center.

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

namespace/netapp-acc-operator created
```

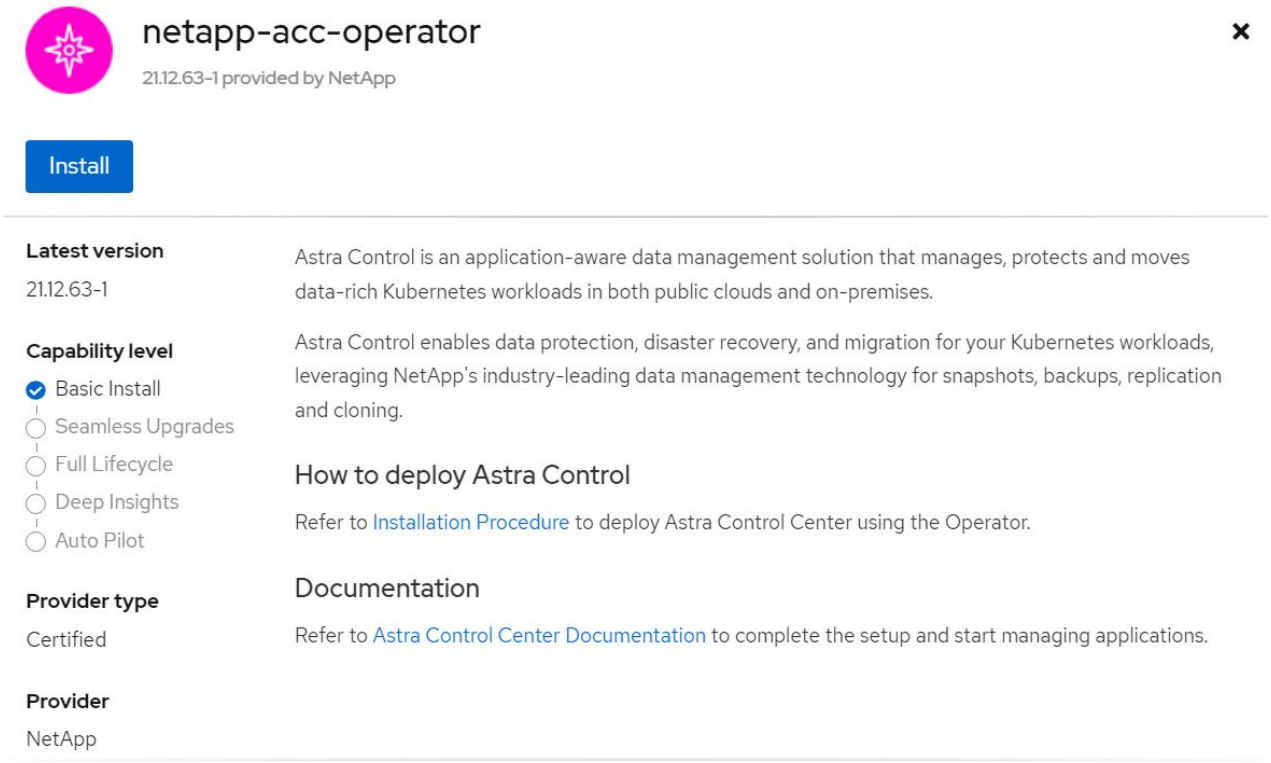
6. Creare un segreto con le credenziali per accedere al registro delle immagini in `netapp-acc-operator` namespace.

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

secret/astra-registry-cred created
```

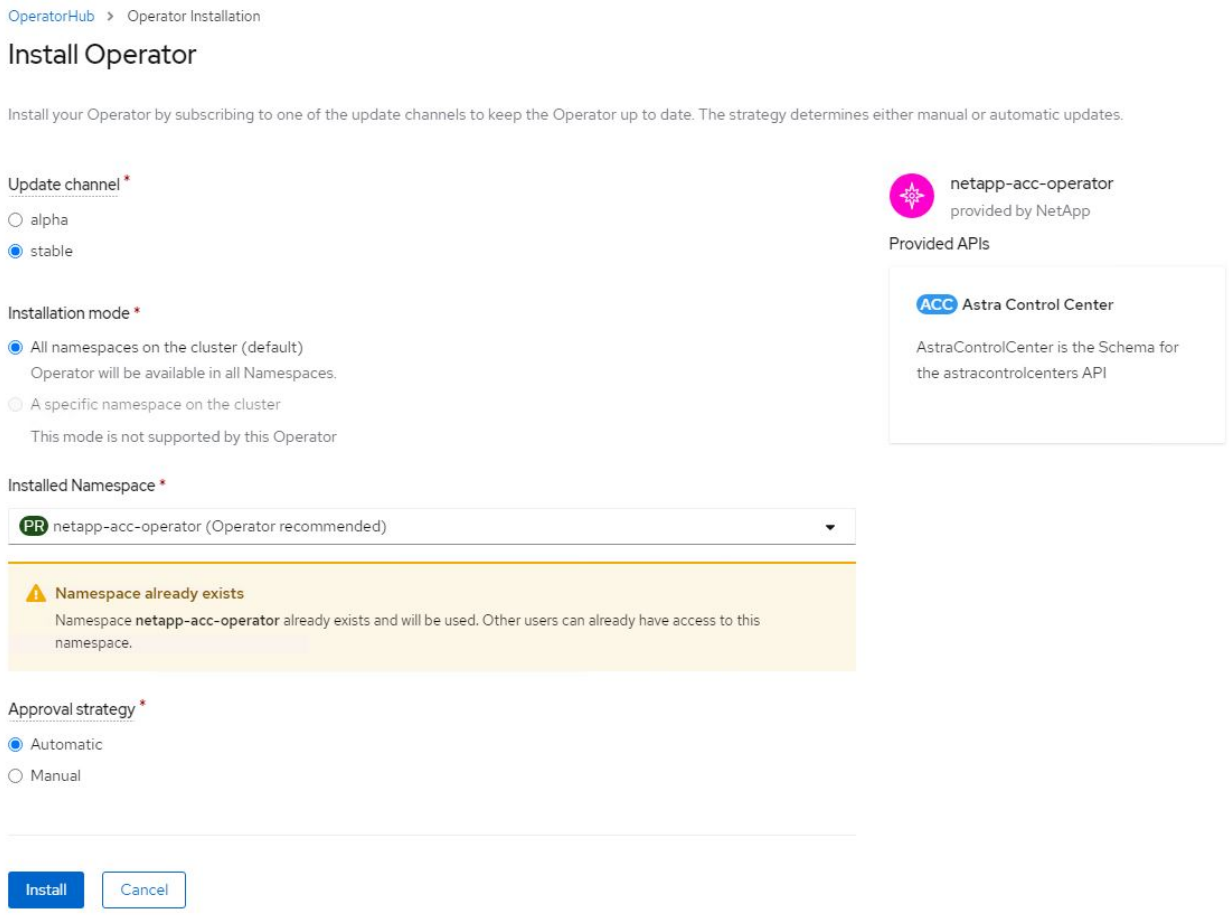
7. Accedi alla console GUI di Red Hat OpenShift con accesso cluster-admin.
8. Selezionare Administrator (Amministratore) dal menu a discesa Perspective (prospettiva).
9. Accedere a Operator > OperatorHub e cercare Astra.



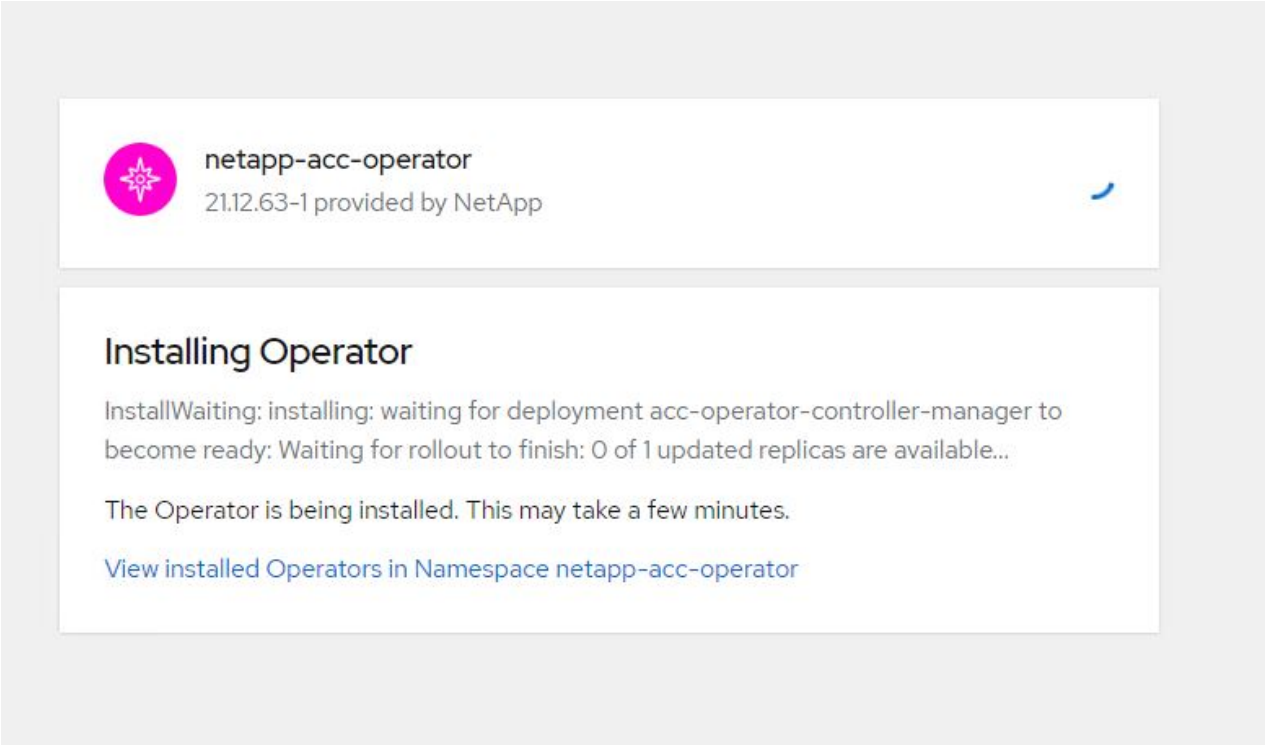
10. Selezionare `netapp-acc-operator` affiancare e fare clic su `Install`.



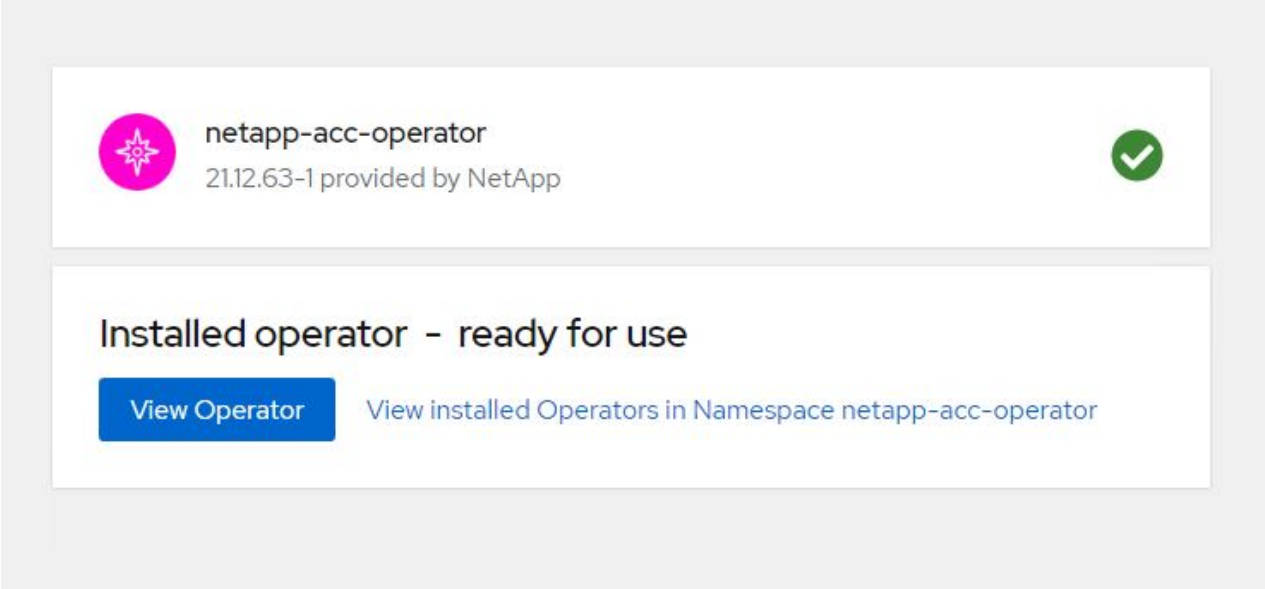
11. Nella schermata Install Operator (Installa operatore), accettare tutti i parametri predefiniti e fare clic su `Install`.



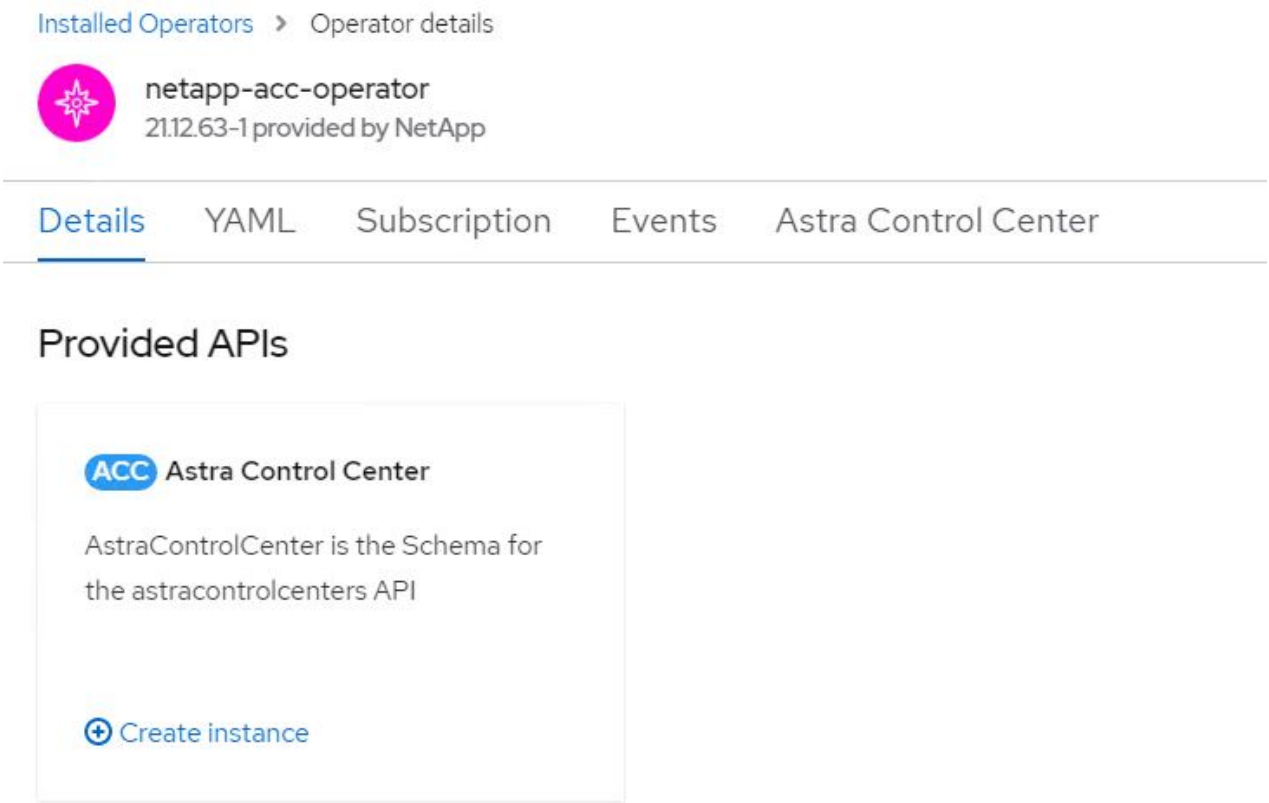
12. Attendere il completamento dell'installazione da parte dell'operatore.



13. Una volta completata l'installazione dell'operatore, selezionare per fare clic su `View Operator`.



14. Quindi fare clic su `Create Instance` Nel riquadro Astra Control Center dell'operatore.



15. Riempire `Create AstraControlCenter` campi del modulo e fare clic su `Create`.

    a. Se si desidera, modificare il nome dell'istanza di Astra Control Center.
    b. Se si desidera, attivare o disattivare il supporto automatico. Si consiglia di mantenere la funzionalità di supporto automatico.
    c. Inserire il nome FQDN per Astra Control Center.
    d. Inserire la versione di Astra Control Center; per impostazione predefinita viene visualizzata la

versione più recente.

e. Inserisci un nome account per Astra Control Center e i dettagli dell'amministratore come nome, cognome e indirizzo e-mail.

f. Inserire il criterio di recupero del volume, l'impostazione predefinita è Mantieni.

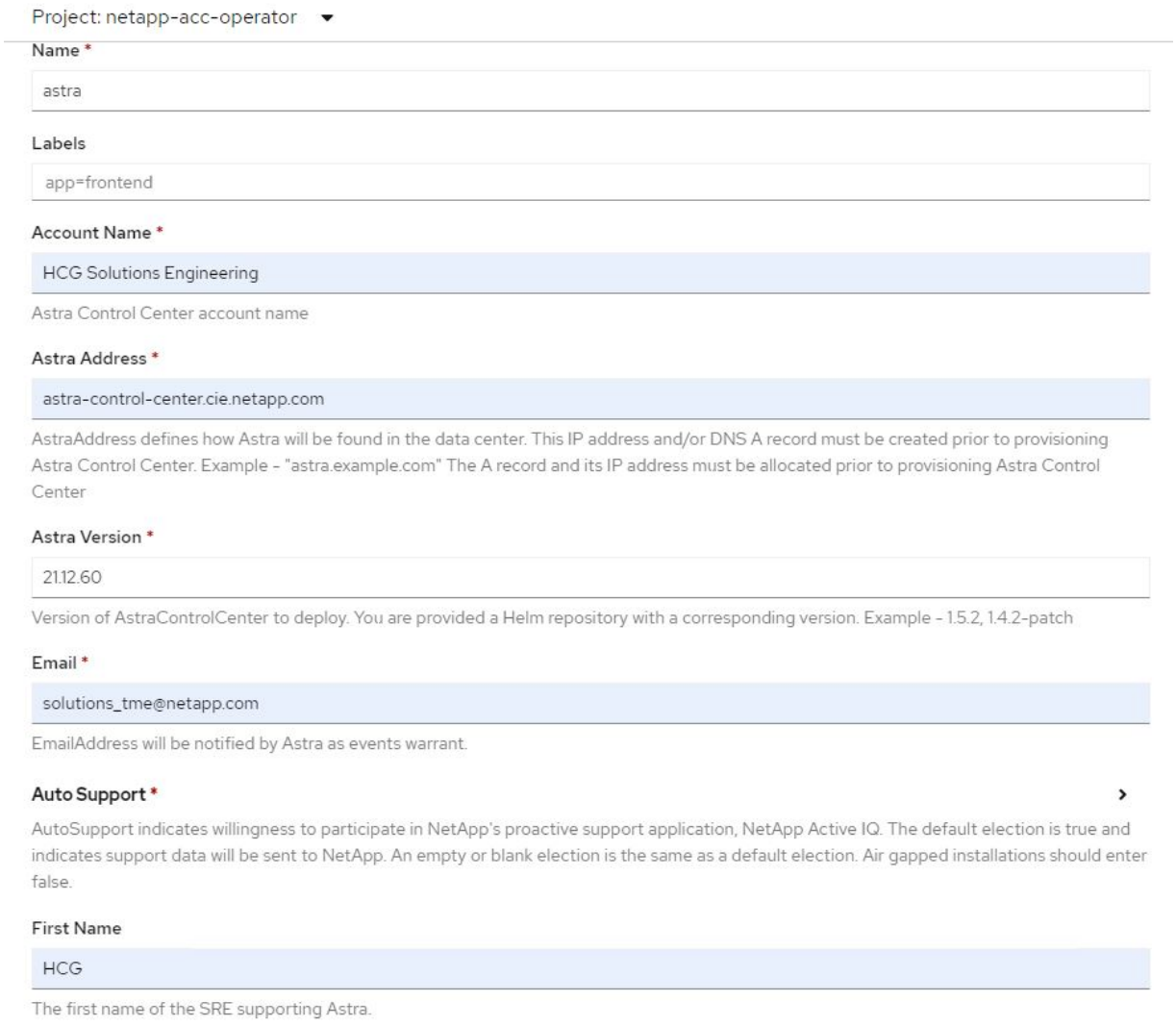
g. In Image Registry (Registro immagini), immettere l'FQDN del registro insieme al nome dell'organizzazione assegnato durante l'invio delle immagini al registro (in questo esempio, `astra-registry.apps.ocp-vmw.cie.netapp.com/netapp-astra`)

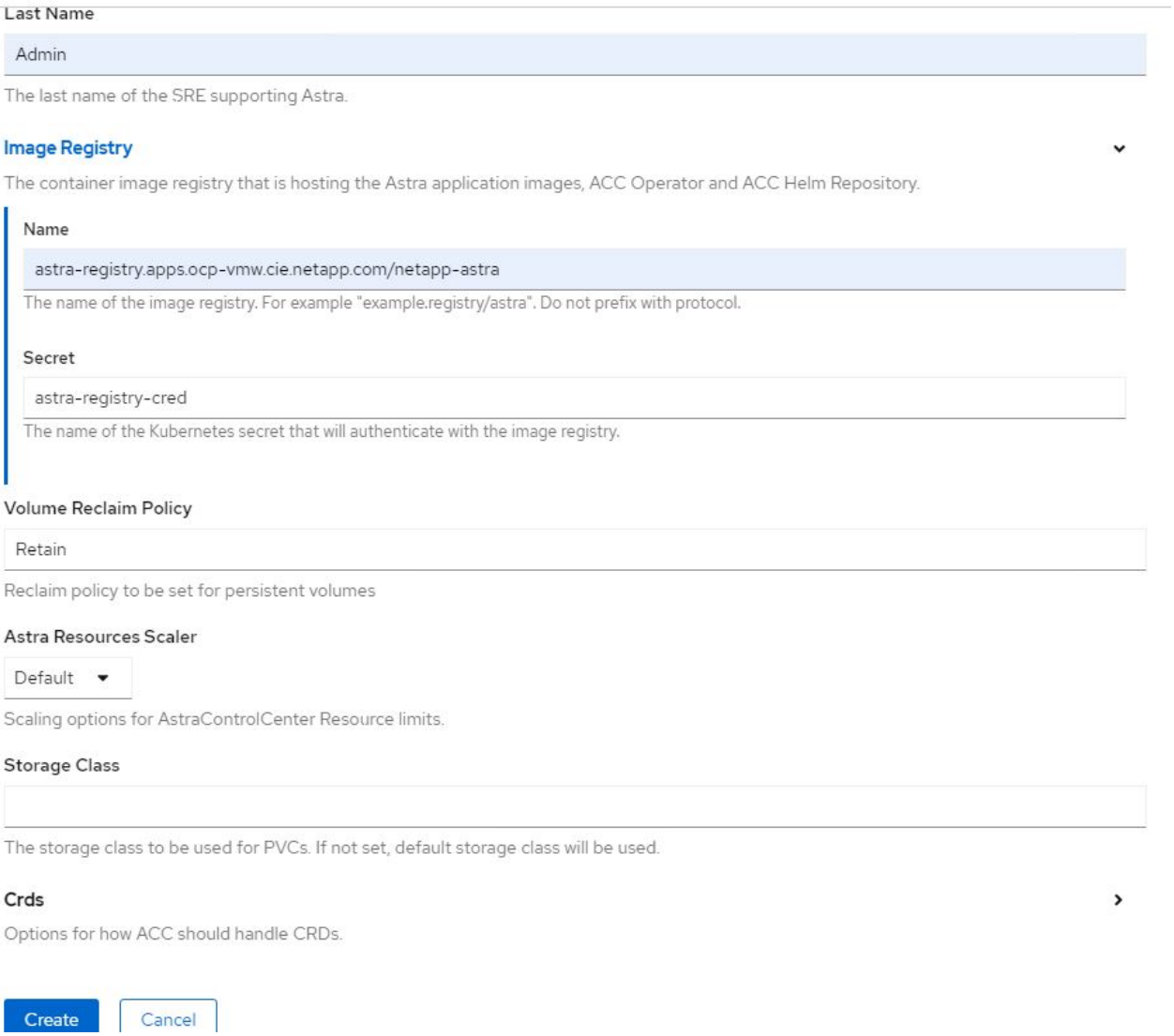
h. Se si utilizza un registro che richiede l'autenticazione, inserire il nome segreto nella sezione Registro immagini.

i. Configurare le opzioni di scalabilità per i limiti delle risorse di Astra Control Center.

j. Inserire il nome della classe di storage se si desidera inserire PVC in una classe di storage non predefinita.

k. Definire le preferenze di gestione CRD.





Automatizzato [Ansible]

1. Per utilizzare i playbook Ansible per implementare Astra Control Center, è necessaria una macchina Ubuntu/RHEL con Ansible installato. Seguire le procedure "qui" Per Ubuntu e RHEL.
2. Clonare il repository GitHub che ospita il contenuto Ansible.

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

3. Accedi al sito NetApp Support e scarica l'ultima versione di NetApp Astra Control Center. Per farlo, è necessaria una licenza allegata al tuo account NetApp. Dopo aver scaricato il tarball, trasferirlo sulla workstation.

> Per iniziare a utilizzare una licenza di prova per Astra Control, visitare il sito "Sito di registrazione Astra".

4. Creare o ottenere il file kubeconfig con accesso amministratore al cluster OpenShift su cui deve essere installato Astra Control Center.

5. Modificare la directory in na_astra_control_suite.

```
cd na_astra_control_suite
```

6. Modificare il `vars/vars.yml` e inserire le variabili con le informazioni richieste.

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

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

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

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

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

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

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

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

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

```
storageclass_reclaim_policy: Retain

#Private Registry Details
astra_registry_name: "docker.io"

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

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

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

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

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

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

7. Esegui il manuale per implementare Astra Control Center. Il playbook richiede privilegi root per alcune configurazioni.

   Se l'utente che esegue il playbook è root o ha configurato sudo senza password, eseguire il seguente comando per eseguire il playbook.

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

   Se l'utente ha configurato l'accesso sudo basato su password, eseguire il seguente comando per eseguire il manuale, quindi inserire la password sudo.

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

Fasi successive all'installazione

1. Il completamento dell'installazione potrebbe richiedere alcuni minuti. Verificare che tutti i pod e i servizi in `netapp-astra-cc` namespace in esecuzione.

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

2. Controllare `acc-operator-controller-manager` registri per garantire che l'installazione sia completata.

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

> Il seguente messaggio indica la corretta installazione di Astra Control Center.

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

3. Il nome utente per l'accesso ad Astra Control Center è l'indirizzo e-mail dell'amministratore fornito nel file CRD e la password è una stringa `ACC-` Aggiunto all'UUID di Astra Control Center. Eseguire il seguente comando:

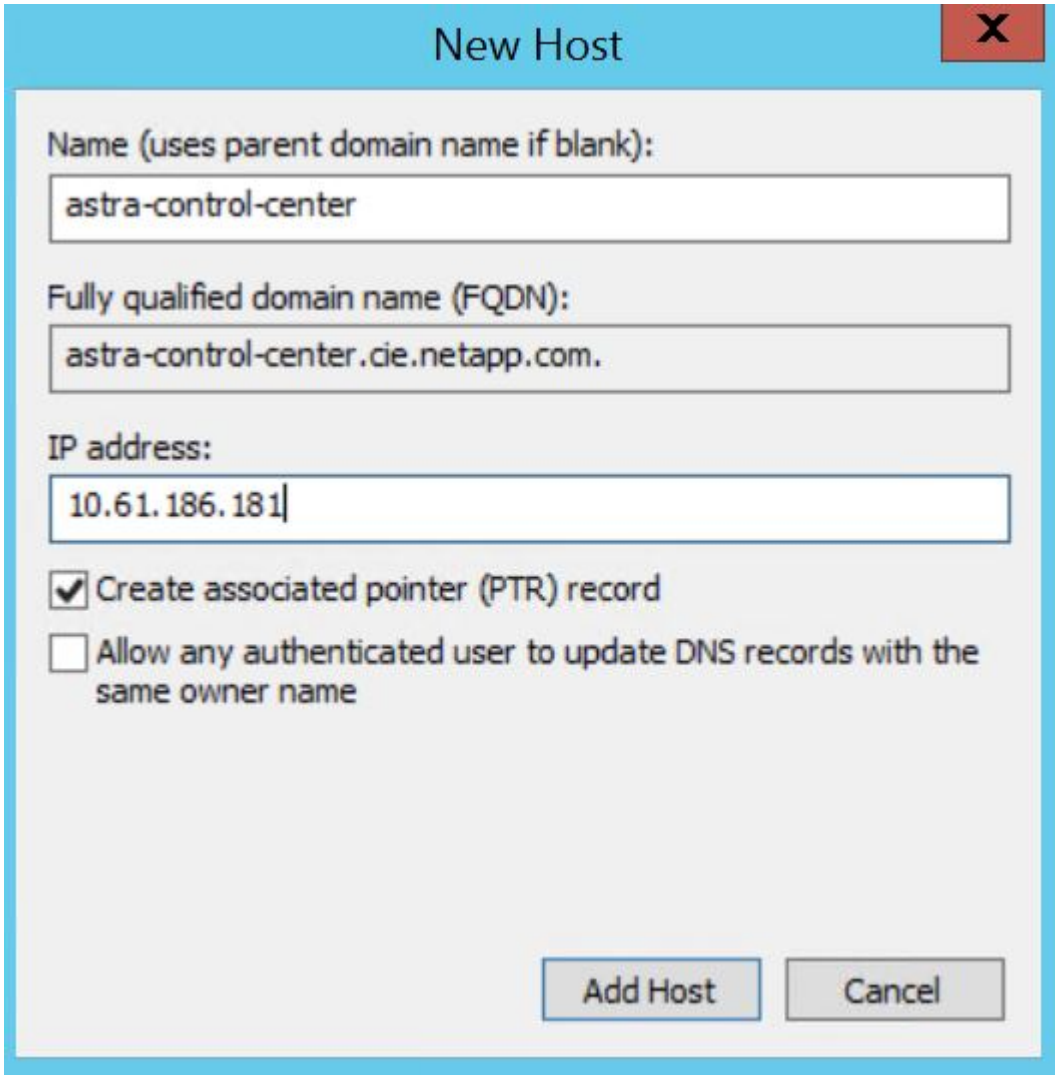
```
[netapp-user@rhel7 ~]$ oc get astracontrolcenters -n netapp-astra-cc
NAME    UUID
astra   345c55a5-bf2e-21f0-84b8-b6f2bce5e95f
```

> In questo esempio, la password è `ACC-345c55a5-bf2e-21f0-84b8-b6f2bce5e95f`.

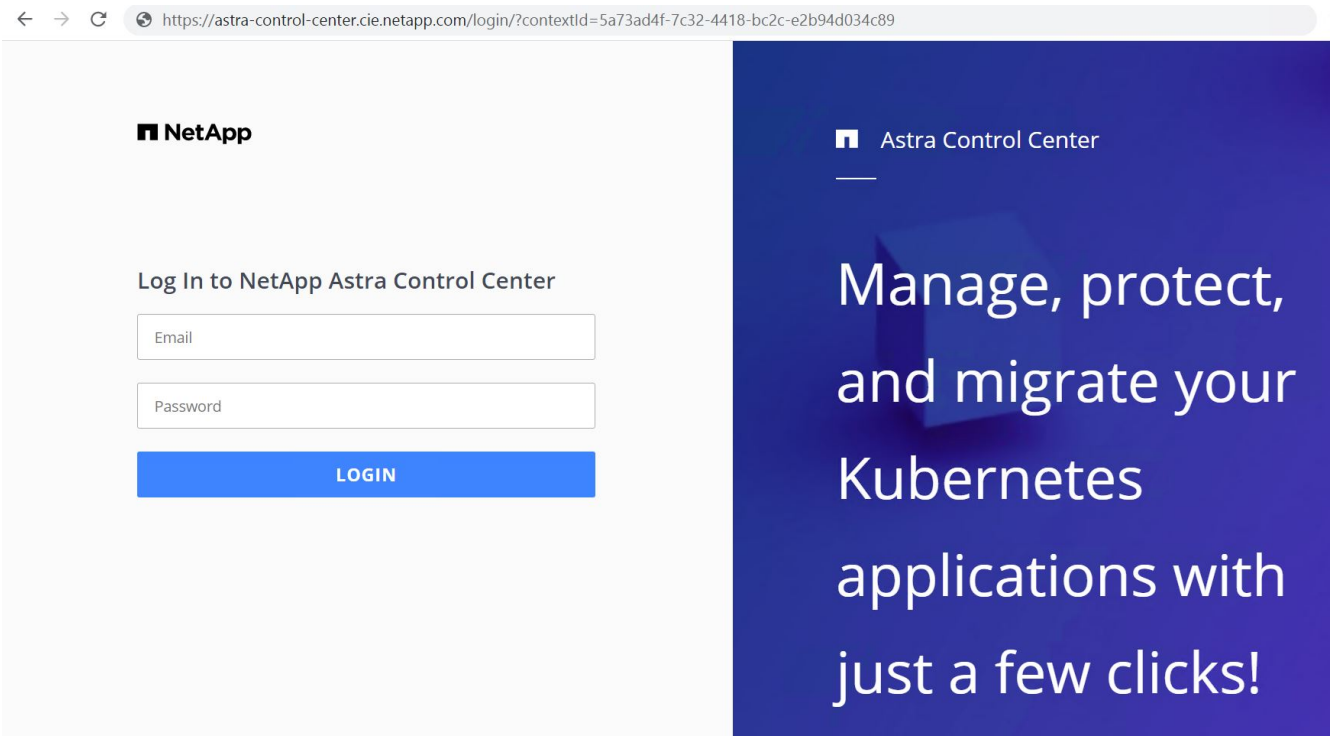
4. Ottieni l'IP del bilanciamento del carico del servizio traefik.

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

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

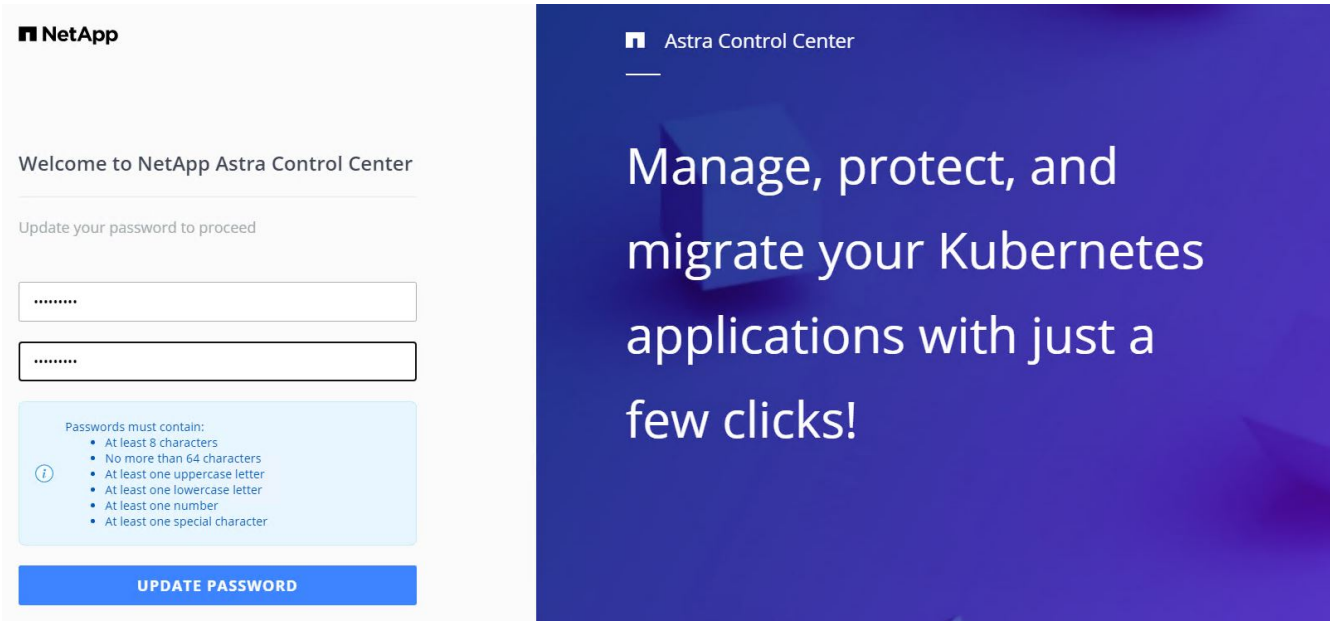
5. Aggiungere una voce nel server DNS che punta all'FQDN fornito nel file CRD di Astra Control Center `EXTERNAL-IP` del servizio traefik.



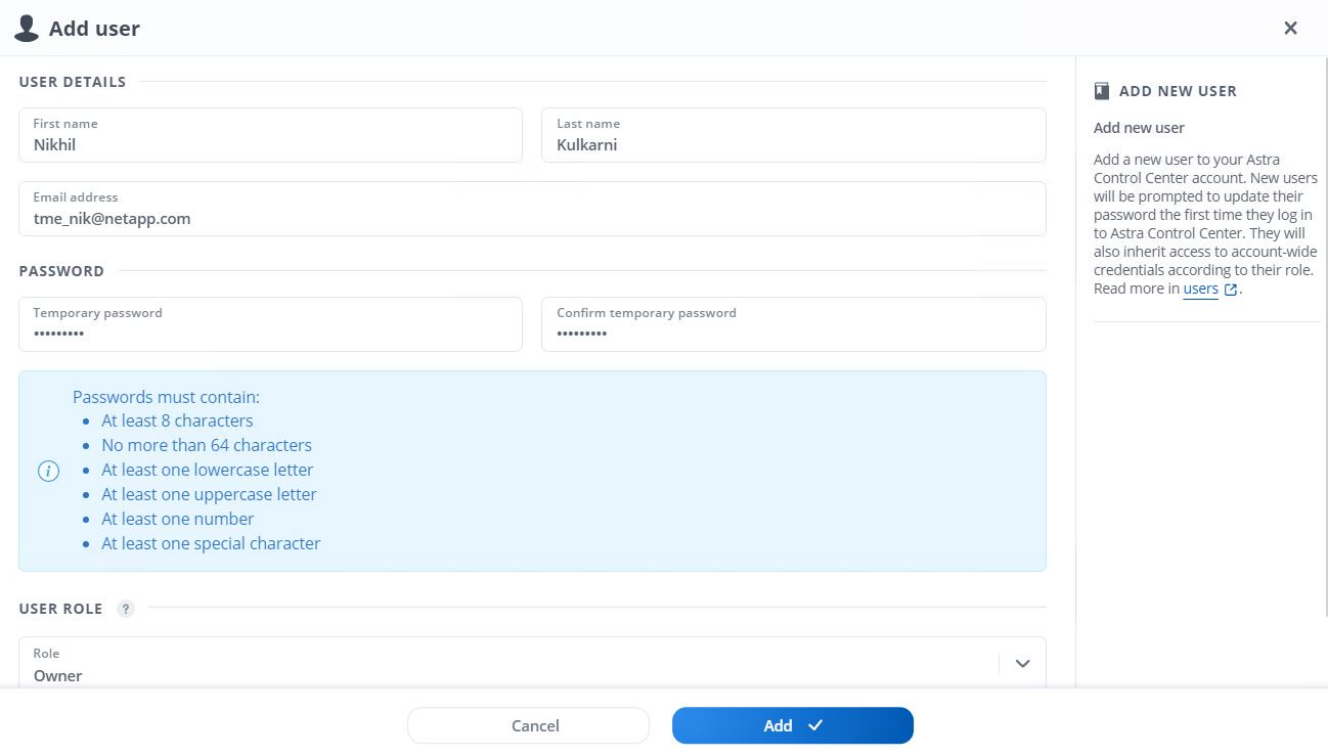
6. Accedere alla GUI di Astra Control Center esplorando il relativo FQDN.



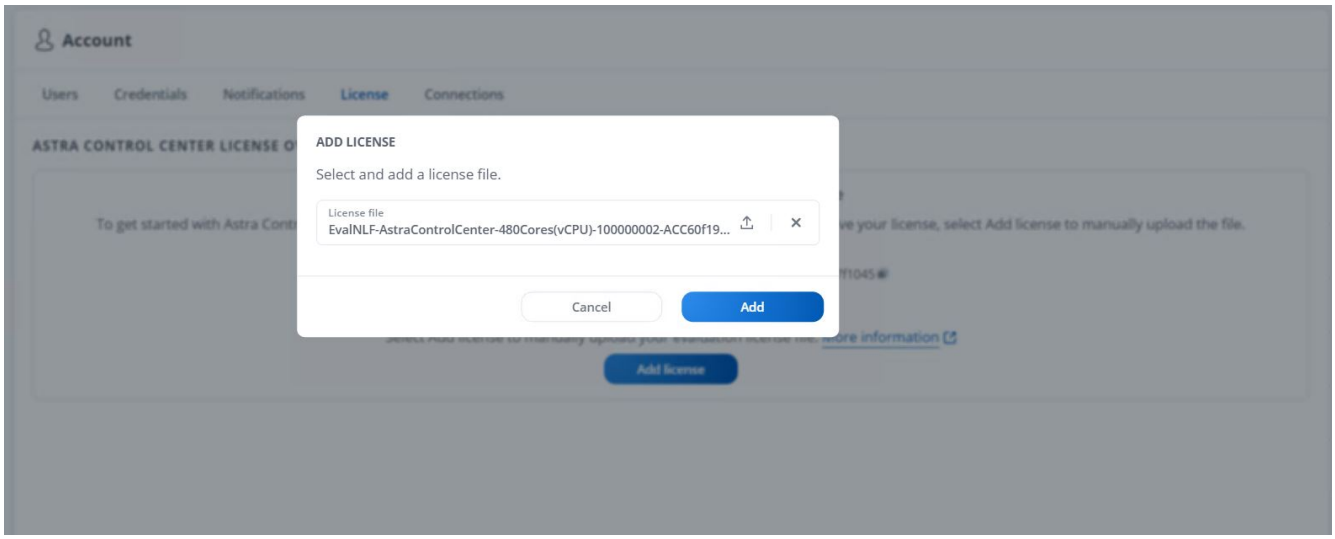
7. Quando si accede all'interfaccia grafica di Astra Control Center per la prima volta utilizzando l'indirizzo email admin fornito in CRD, è necessario modificare la password.



8. Se si desidera aggiungere un utente ad Astra Control Center, accedere a account > Users (account > utenti), fare clic su Add (Aggiungi), inserire i dettagli dell'utente e fare clic su Add (Aggiungi).



9. Astra Control Center richiede una licenza per il funzionamento di tutte le funzionalità IT. Per aggiungere una licenza, accedere a account > License (account > licenza), fare clic su Add License (Aggiungi licenza) e caricare il file di licenza.



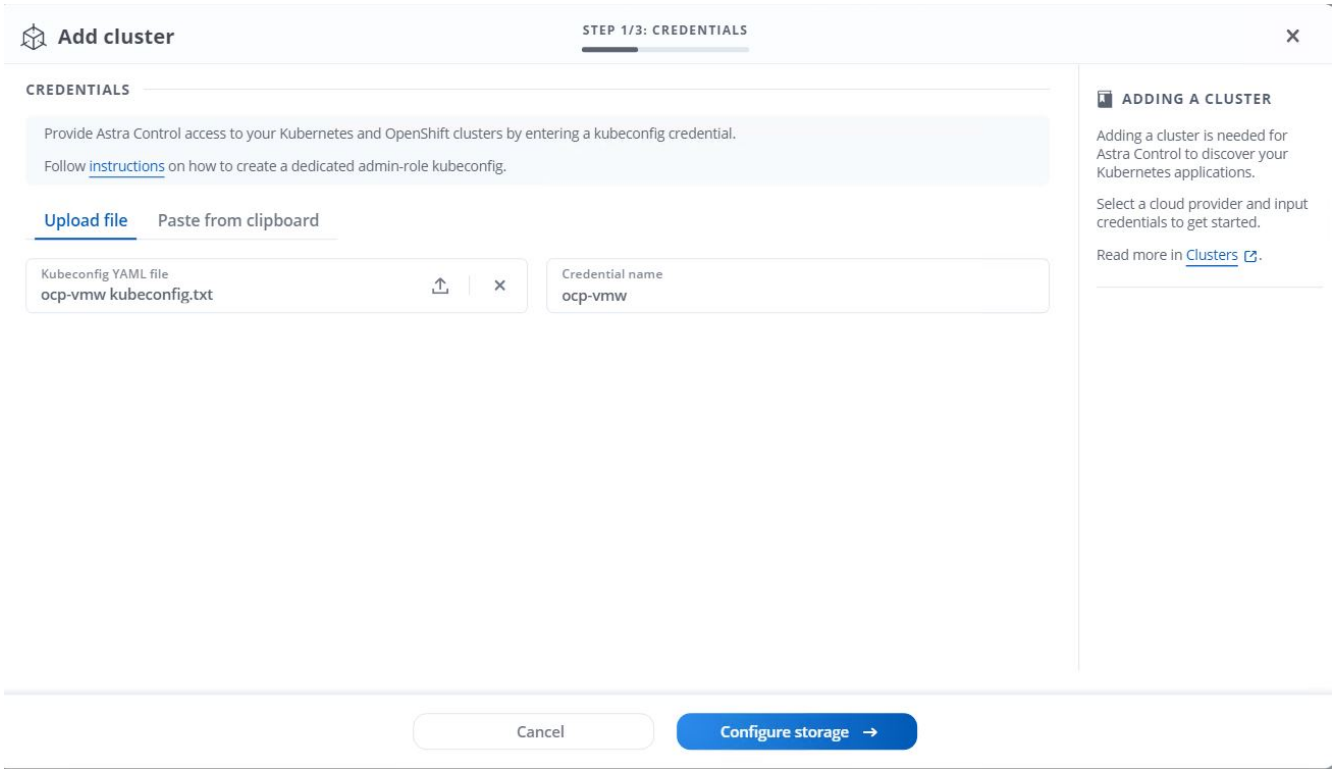
In caso di problemi con l'installazione o la configurazione di NetApp Astra Control Center, è disponibile la knowledge base dei problemi noti "qui".

Registra i tuoi Red Hat OpenShift Clusters con Astra Control Center

Per consentire ad Astra Control Center di gestire i carichi di lavoro, devi prima registrare il cluster Red Hat OpenShift.

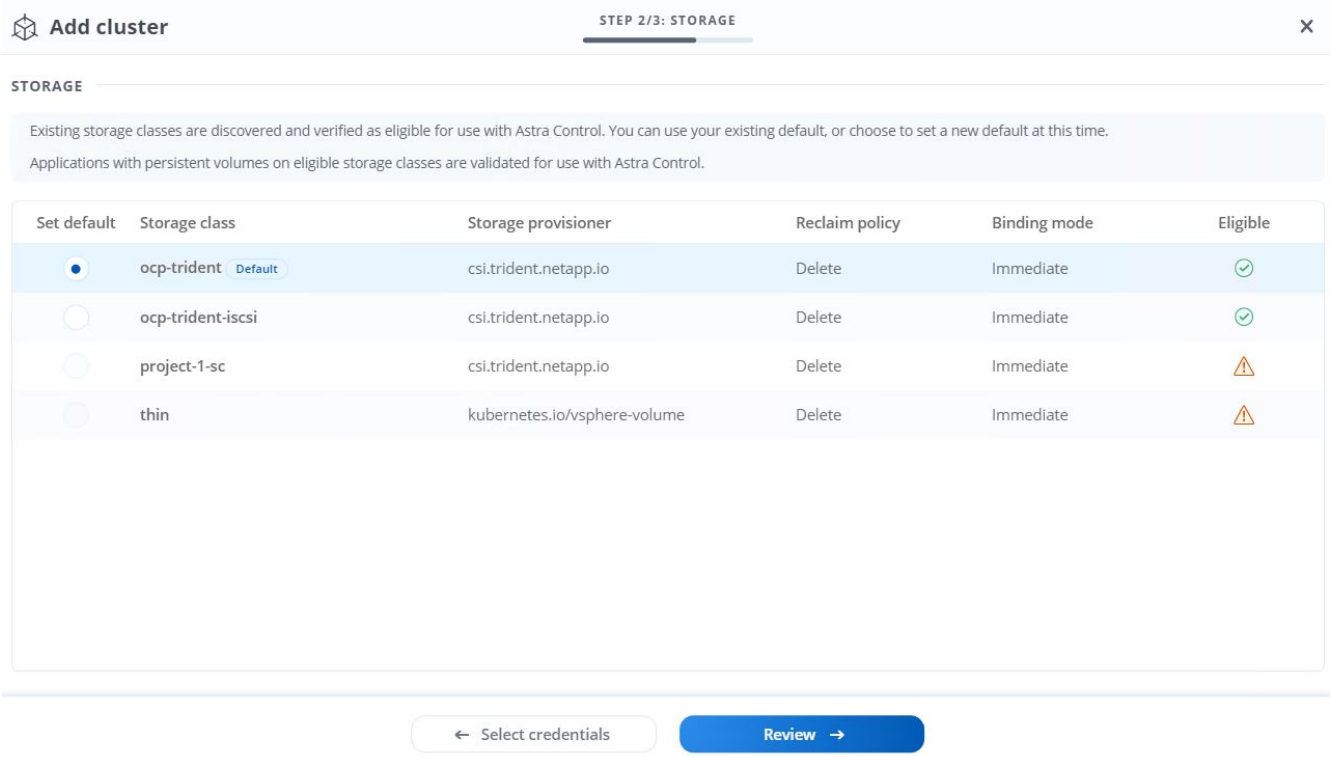
Registra i cluster Red Hat OpenShift

1. Il primo passo consiste nell'aggiungere i cluster OpenShift all'Astra Control Center e gestirli. Accedere a Clusters e fare clic su Add a Cluster (Aggiungi cluster), caricare il file kubeconfig per il cluster OpenShift e fare clic su Select Storage (Seleziona storage).

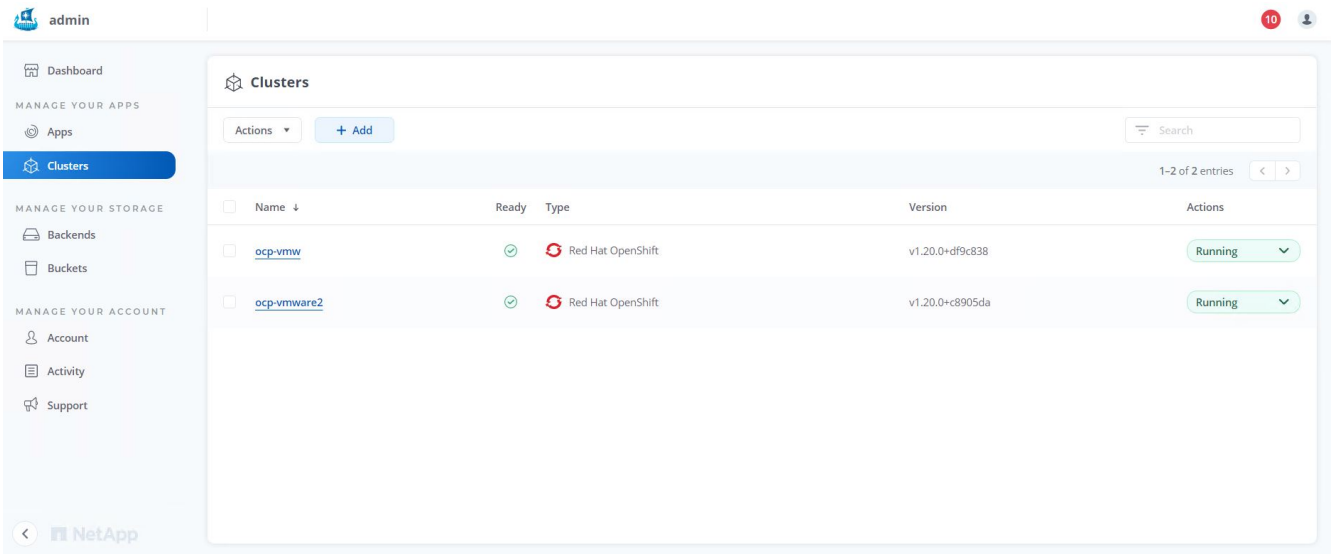


> Il file kubeconfig può essere generato per l'autenticazione con un nome utente e una password o un token. I token scadono dopo un periodo di tempo limitato e potrebbero non essere raggiungibili dal cluster registrato. NetApp consiglia di utilizzare un file kubeconfig con nome utente e password per registrare i cluster OpenShift su Astra Control Center.

2. Astra Control Center rileva le classi di storage idonee. Selezionare ora il modo in cui lo storageclass effettua il provisioning dei volumi utilizzando Trident supportato da una SVM su NetApp ONTAP e fare clic su Review (esamina). Nel riquadro successivo, verificare i dettagli e fare clic su Add Cluster (Aggiungi cluster).

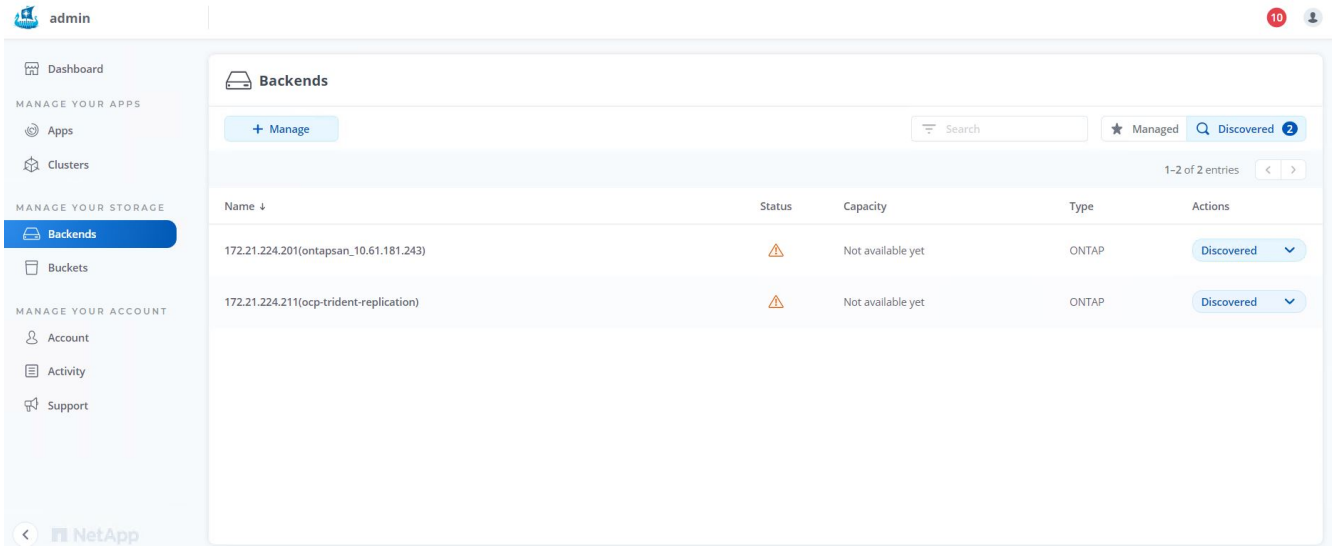


3. Registrare entrambi i cluster OpenShift come descritto al punto 1. Una volta aggiunti, i cluster passano allo stato di rilevamento mentre Astra Control Center li ispeziona e installa gli agenti necessari. Lo stato del cluster diventa in esecuzione dopo che sono stati registrati correttamente.

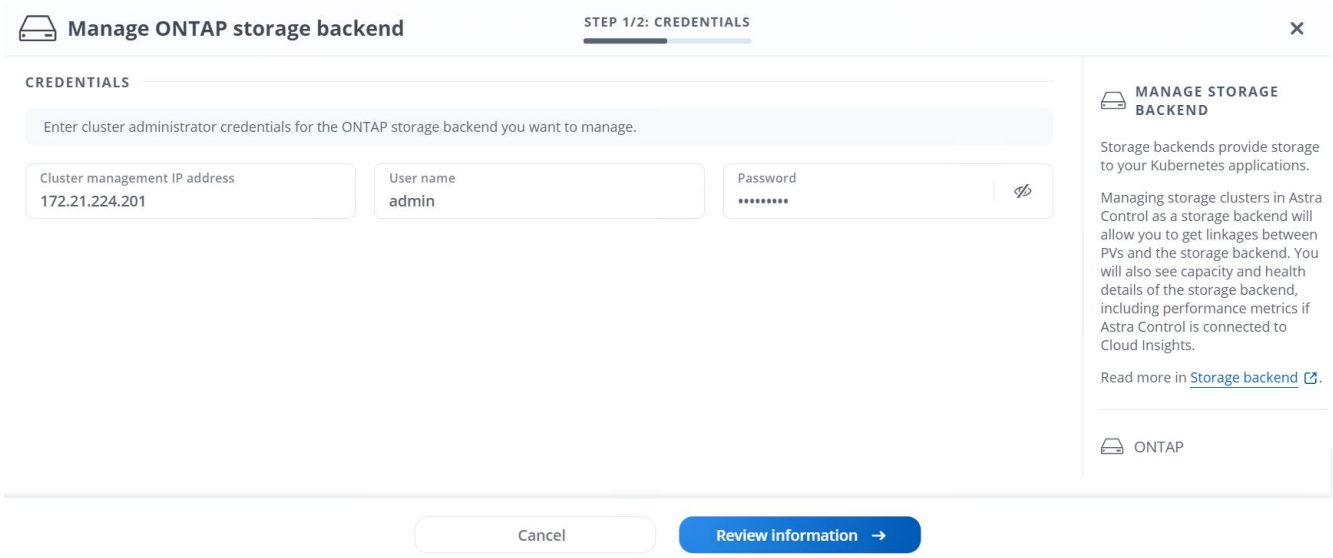


> Tutti i cluster Red Hat OpenShift che devono essere gestiti da Astra Control Center devono avere accesso al registro delle immagini utilizzato per l'installazione, poiché gli agenti installati sui cluster gestiti estraggono le immagini da tale registro.

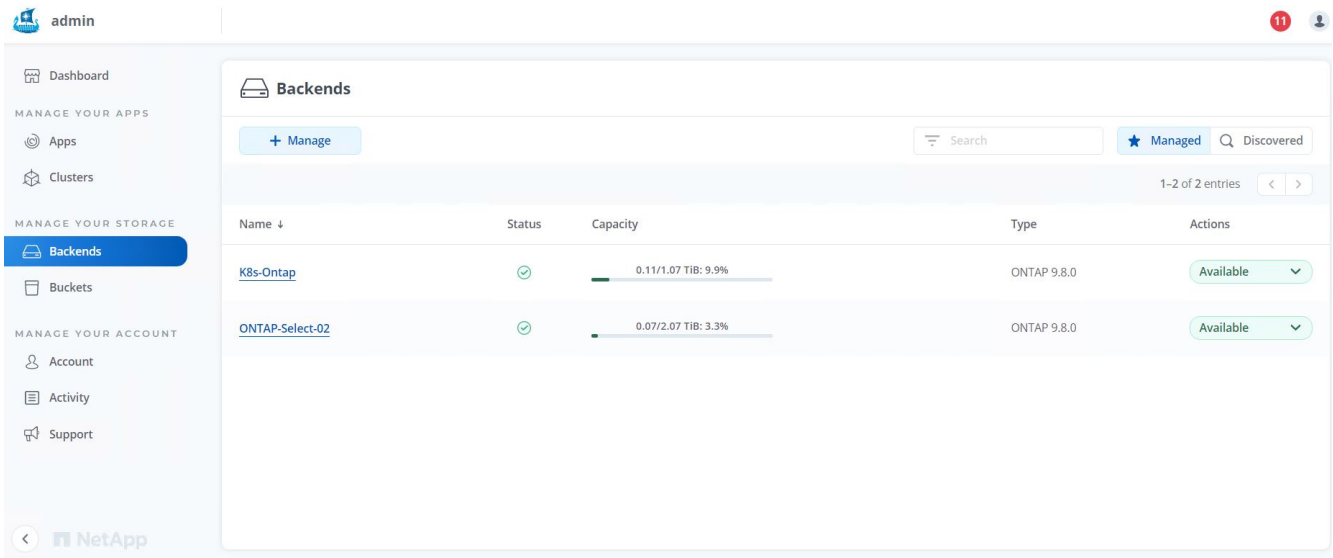
4. Importa i cluster ONTAP come risorse storage da gestire come back-end dal centro di controllo Astra. Quando i cluster OpenShift vengono aggiunti ad Astra e viene configurato uno storageclass, il cluster ONTAP viene automaticamente ispezionato e ispezionato per il backup dello storageclass, ma non viene importato nel centro di controllo Astra da gestire.



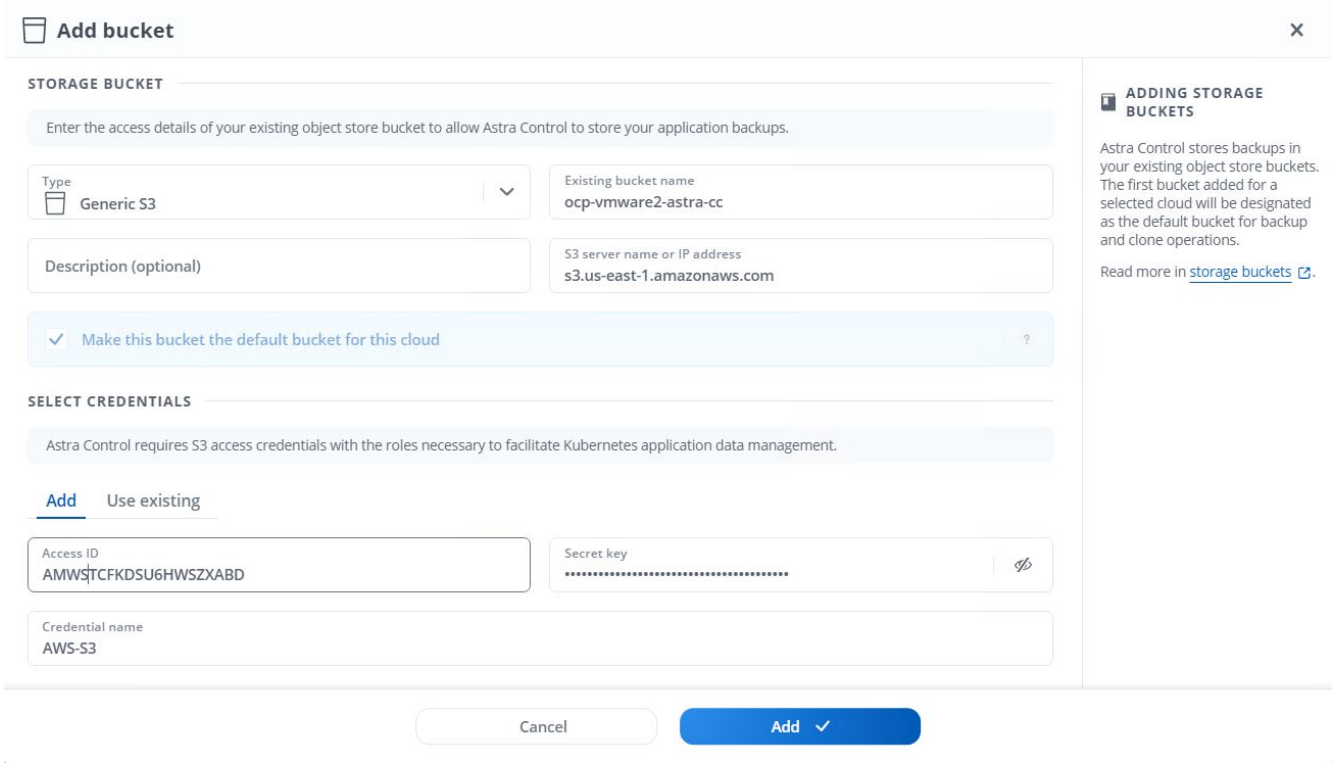
5. Per importare i cluster ONTAP, accedere a Backend, fare clic sul menu a discesa e selezionare Manage (Gestisci) accanto al cluster ONTAP da gestire. Immettere le credenziali del cluster ONTAP, fare clic su informazioni di revisione, quindi fare clic su Importa backend storage.



6. Una volta aggiunti i backend, lo stato diventa disponibile. Questi backend ora dispongono delle informazioni sui volumi persistenti nel cluster OpenShift e sui volumi corrispondenti nel sistema ONTAP.



7. Per il backup e il ripristino tra cluster OpenShift utilizzando Astra Control Center, è necessario eseguire il provisioning di un bucket di storage a oggetti che supporti il protocollo S3. Le opzioni attualmente supportate sono ONTAP S3, StorageGRID e AWS S3. Ai fini di questa installazione, configureremo un bucket AWS S3. Accedere a Bucket, fare clic su Add bucket (Aggiungi bucket) e selezionare Generic S3. Inserisci i dettagli sul bucket S3 e le credenziali per accedervi, fai clic sulla casella di controllo "Rendi questo bucket il bucket predefinito per il cloud", quindi fai clic su Aggiungi.

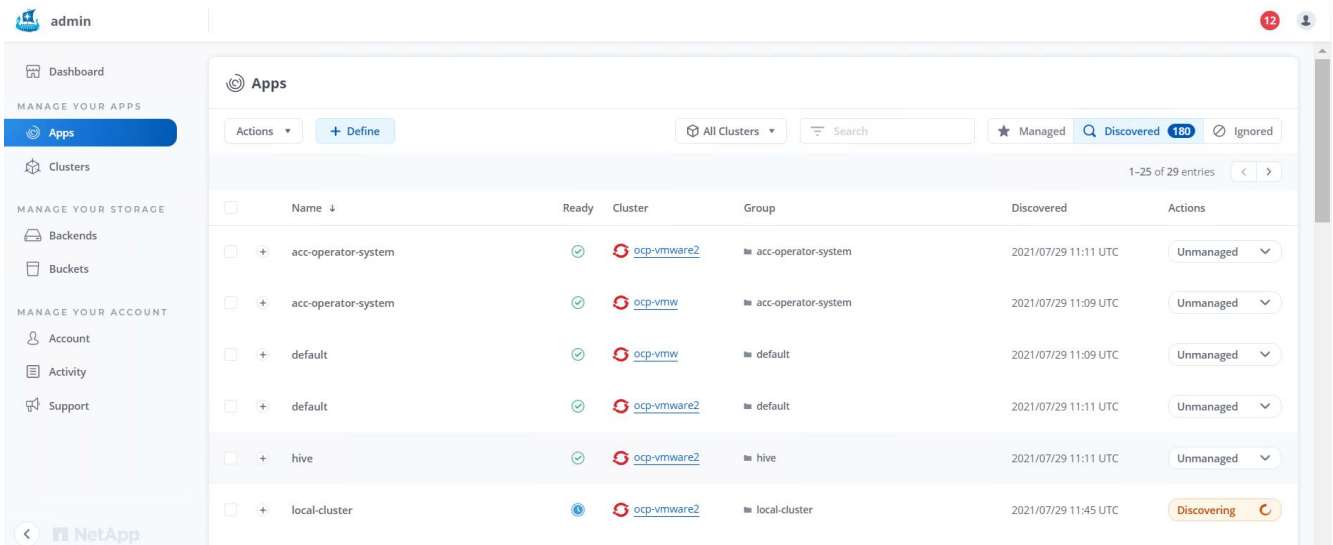


Scegliere le applicazioni da proteggere

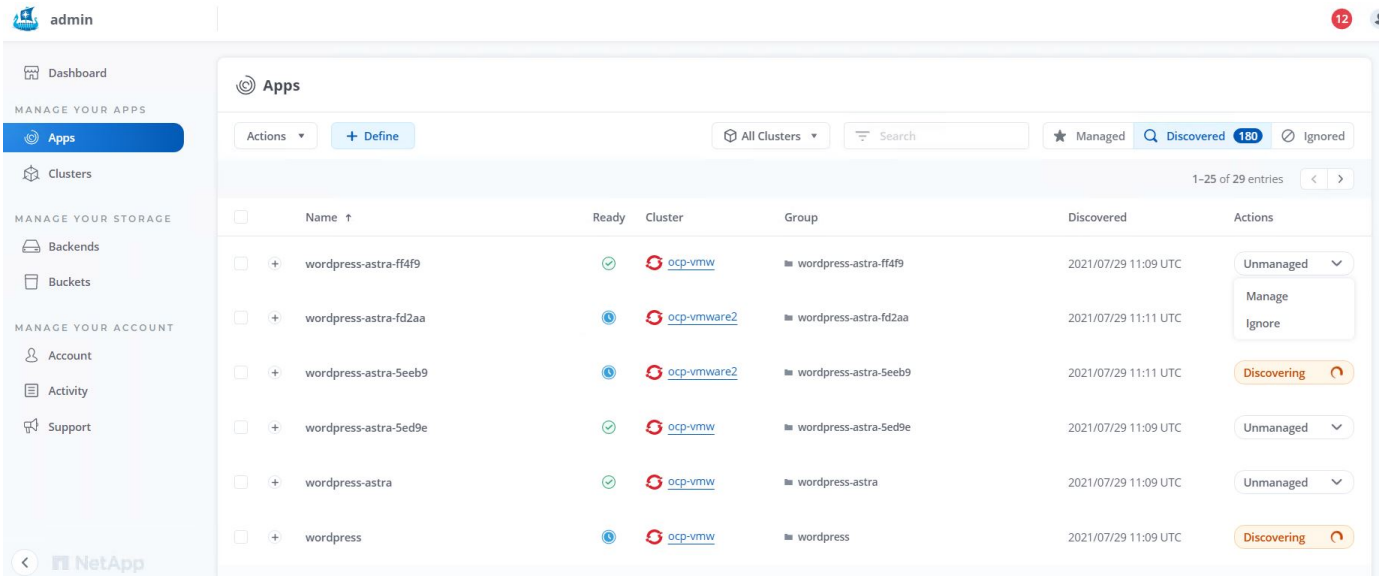
Dopo aver registrato i cluster Red Hat OpenShift, è possibile individuare le applicazioni implementate e gestirle tramite Astra Control Center.

Gestire le applicazioni

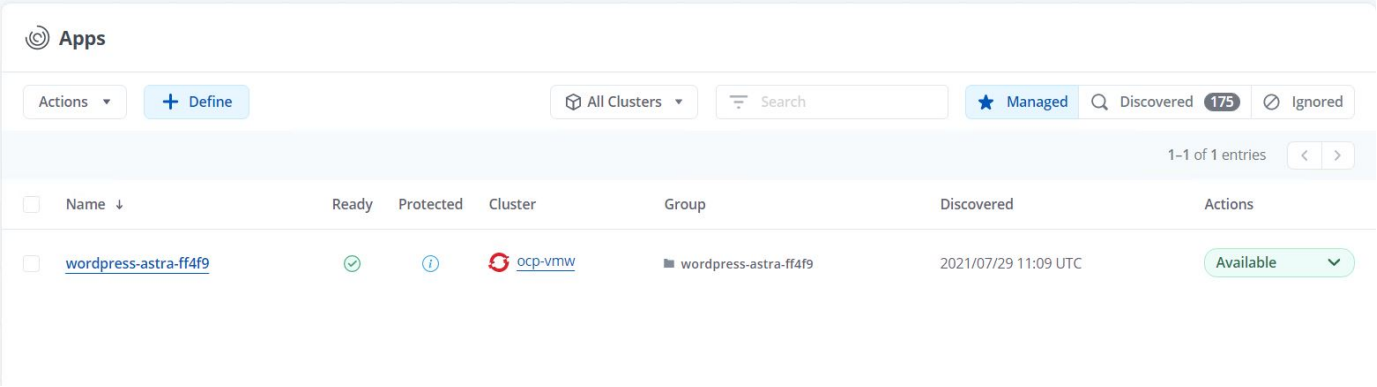
1. Una volta registrati i cluster OpenShift e i backend ONTAP con il centro di controllo Astra, il centro di controllo inizia automaticamente a rilevare le applicazioni in tutti gli spazi dei nomi che utilizzano lo storageclass configurato con il backend ONTAP specificato.



2. Accedere a Apps > Dovered (applicazioni > rilevate) e fare clic sul menu a discesa accanto all'applicazione che si desidera gestire utilizzando Astra. Quindi fare clic su Manage (Gestisci)



1. L'applicazione entra nello stato Available (disponibile) e può essere visualizzata nella scheda Managed (gestito) nella sezione Apps (applicazioni).



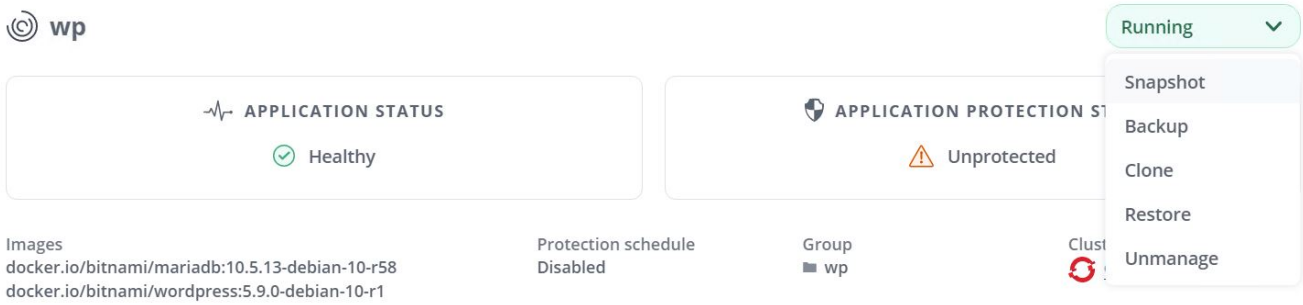
Proteggi le tue applicazioni

Una volta gestiti i carichi di lavoro delle applicazioni da Astra Control Center, è possibile configurare le impostazioni di protezione per tali carichi di lavoro.

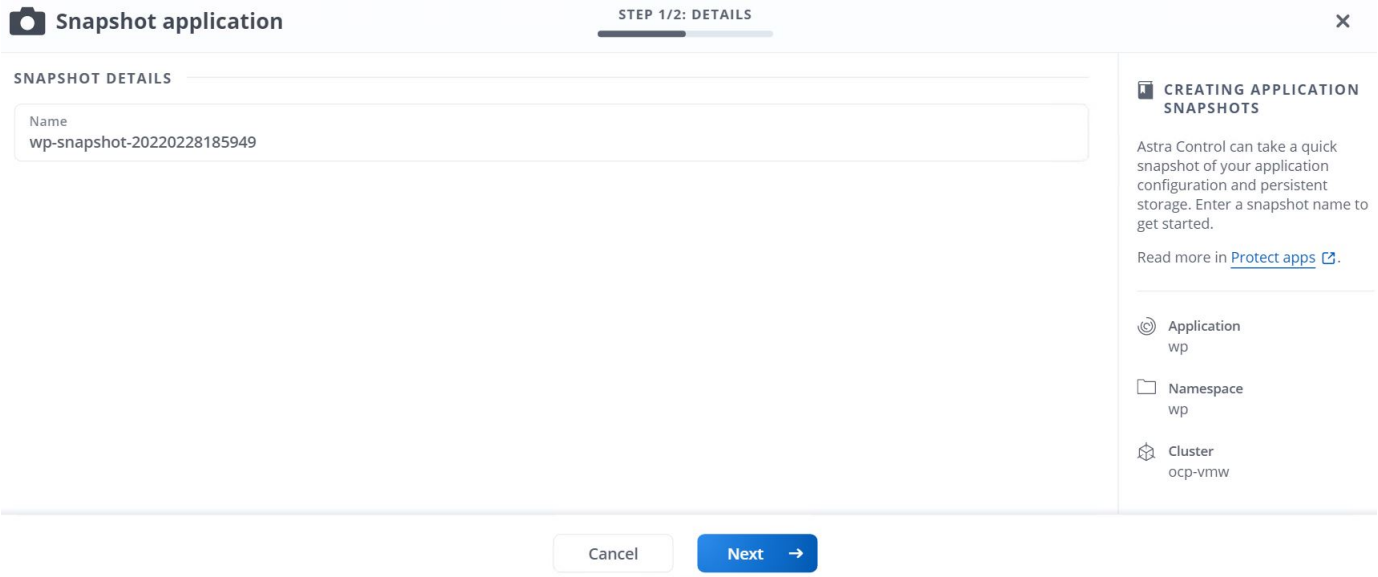
Creazione di un'istantanea dell'applicazione

Un'istantanea di un'applicazione crea una copia Snapshot di ONTAP che può essere utilizzata per ripristinare o clonare l'applicazione in un momento specifico in base a tale copia Snapshot.

1. Per creare un'istantanea dell'applicazione, accedere alla scheda applicazioni > gestite e fare clic sull'applicazione di cui si desidera creare una copia Snapshot. Fare clic sul menu a discesa accanto al nome dell'applicazione e fare clic su Snapshot.



2. Inserire i dettagli dell'istantanea, fare clic su Next (Avanti), quindi su Snapshot (istantanea). La creazione dello snapshot richiede circa un minuto e lo stato diventa disponibile dopo la creazione dello snapshot.



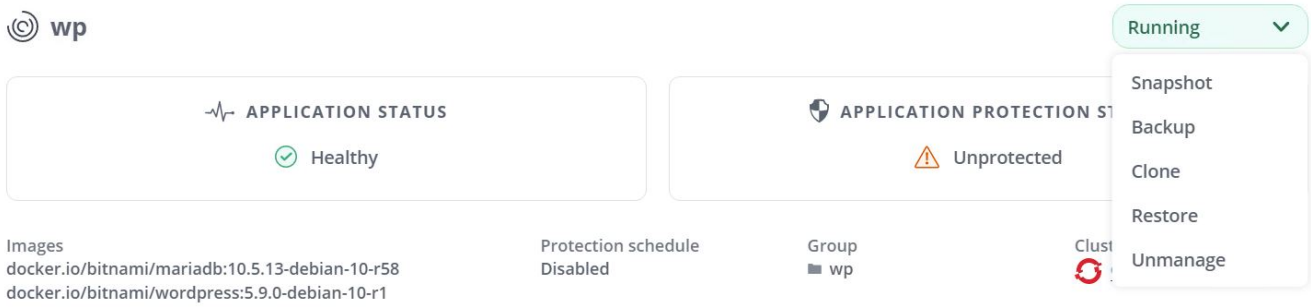
Creazione di un backup dell'applicazione

Un backup di un'applicazione acquisisce lo stato attivo dell'applicazione e la configurazione delle risorse IT, le taglia in file e le memorizza in un bucket di storage a oggetti remoto.

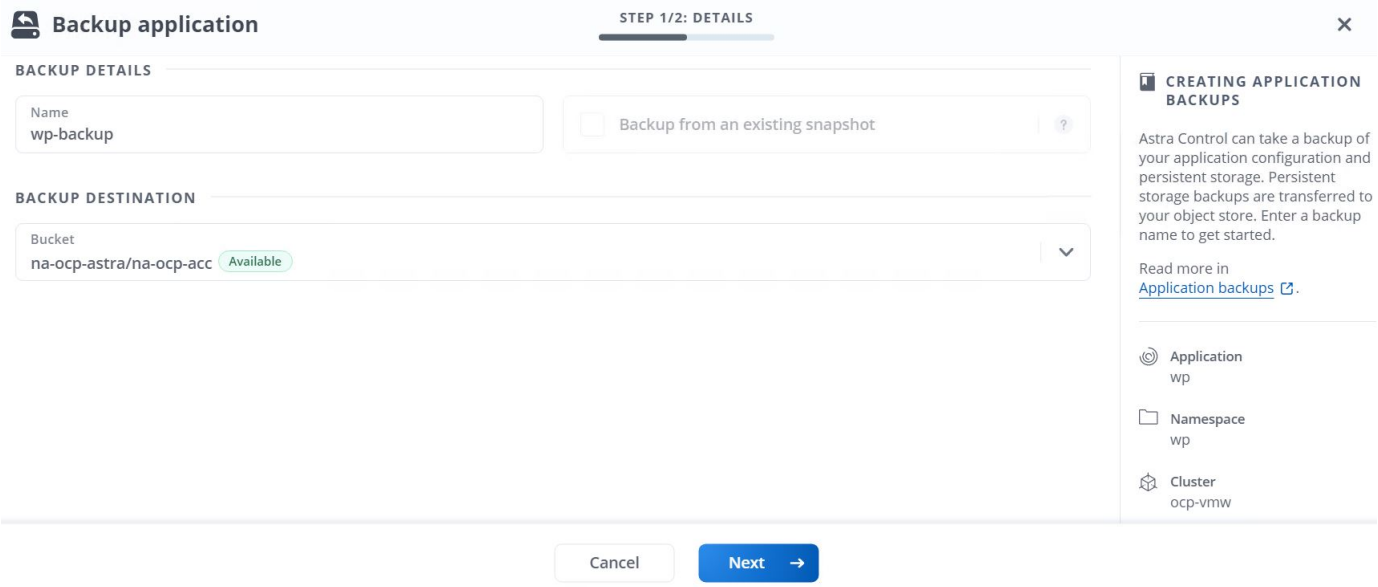
Per il backup e il ripristino delle applicazioni gestite nel centro di controllo Astra, è necessario configurare le impostazioni del superutente per i sistemi ONTAP di backup come prerequisito. A tale scopo, immettere i seguenti comandi.

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

1. Per creare un backup dell'applicazione gestita in Astra Control Center, accedere alla scheda Apps (applicazioni) > Managed (gestite) e fare clic sull'applicazione di cui si desidera eseguire il backup. Fare clic sul menu a discesa accanto al nome dell'applicazione e fare clic su Backup.



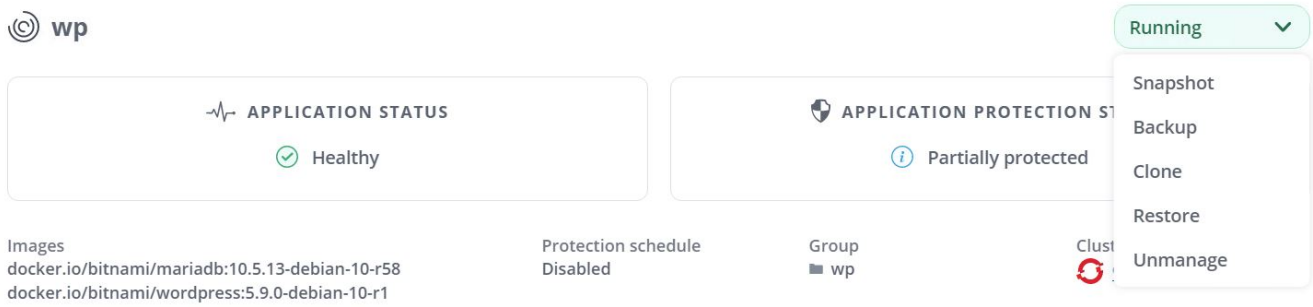
2. Inserire i dettagli del backup, selezionare il bucket di storage a oggetti in cui memorizzare i file di backup, fare clic su Next (Avanti) e, dopo aver esaminato i dettagli, fare clic su Backup (Backup). A seconda delle dimensioni dell'applicazione e dei dati, il backup può richiedere alcuni minuti e lo stato del backup diventa disponibile una volta completato correttamente il backup.



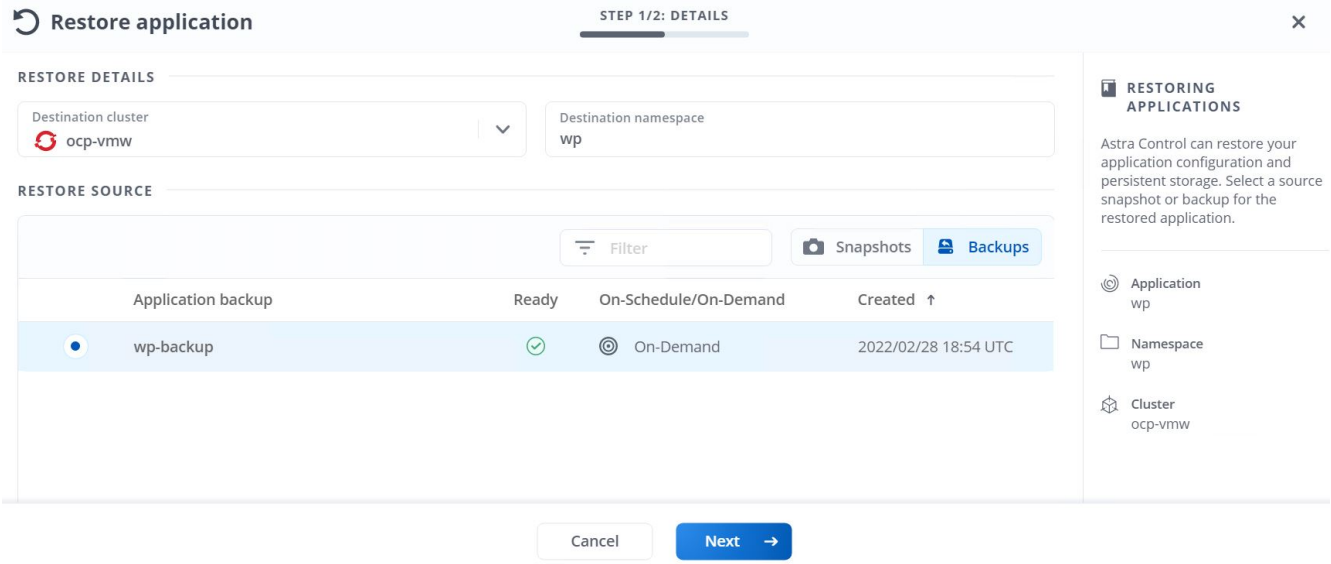
Ripristino di un'applicazione

Con la semplice pressione di un pulsante, è possibile ripristinare un'applicazione nello spazio dei nomi di origine nello stesso cluster o in un cluster remoto per la protezione delle applicazioni e il disaster recovery.

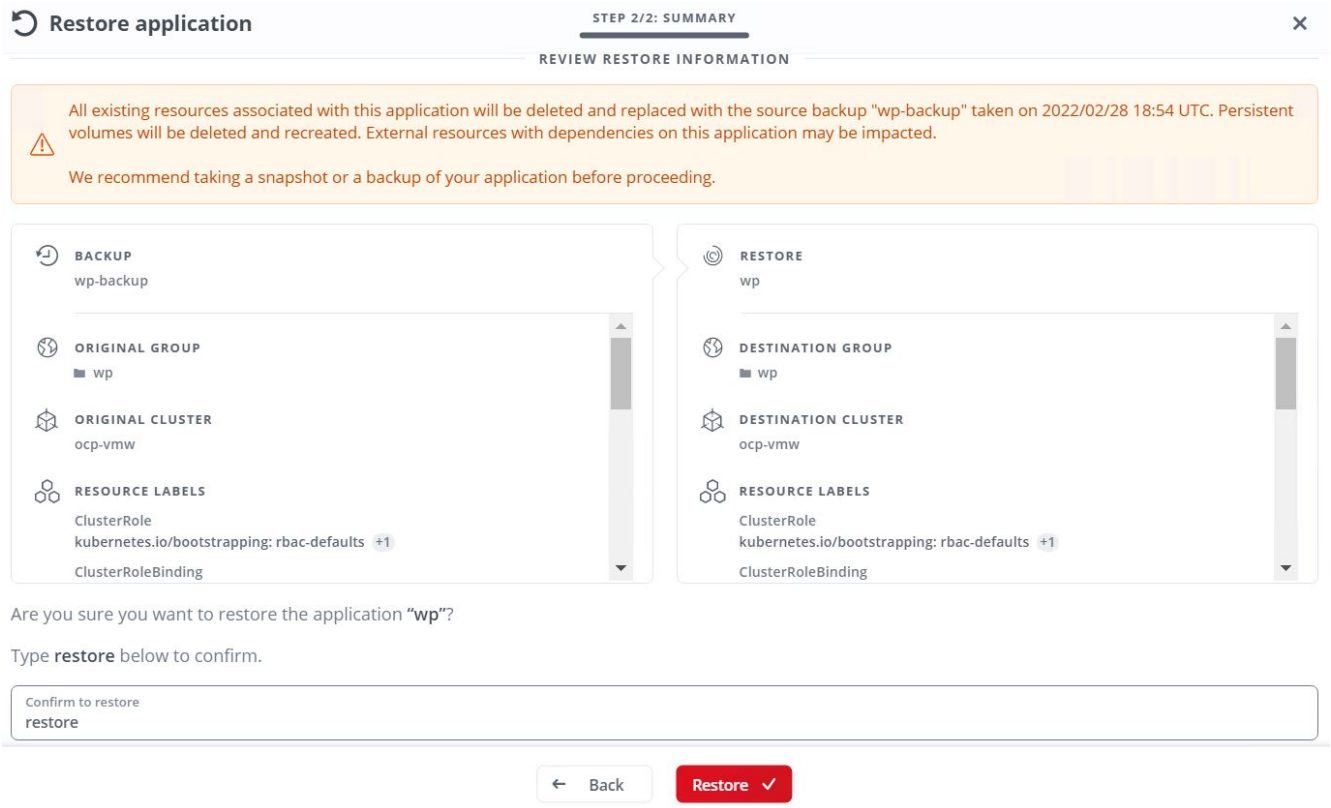
1. Per ripristinare un'applicazione, selezionare Apps (applicazioni) > Managed Tab (scheda gestita) e fare clic sull'applicazione in questione. Fare clic sul menu a discesa accanto al nome dell'applicazione e fare clic su `Restore`.



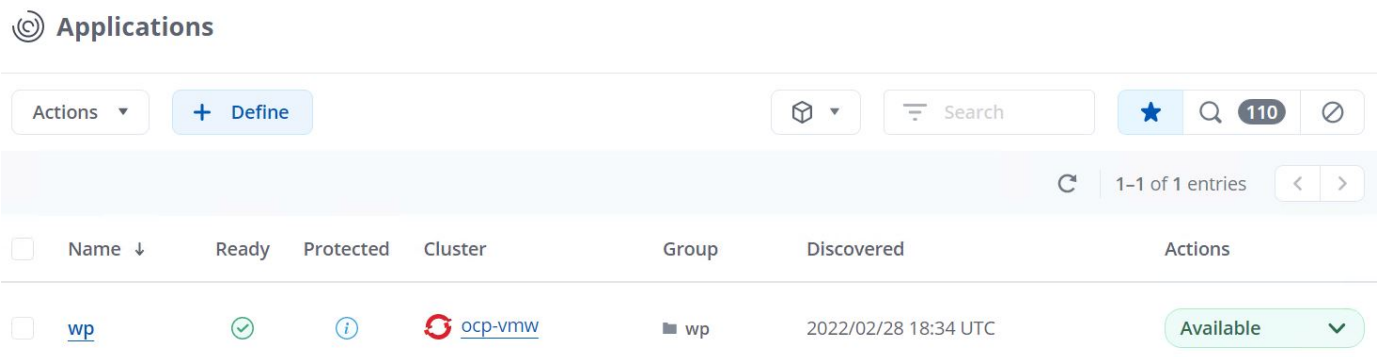
2. Immettere il nome dello spazio dei nomi di ripristino, selezionare il cluster in cui si desidera ripristinarlo e scegliere se si desidera ripristinarlo da uno snapshot esistente o da un backup dell'applicazione. Fare clic su Avanti.



3. Nel riquadro di revisione, immettere `restore` E fare clic su Restore (Ripristina) dopo aver esaminato i dettagli.



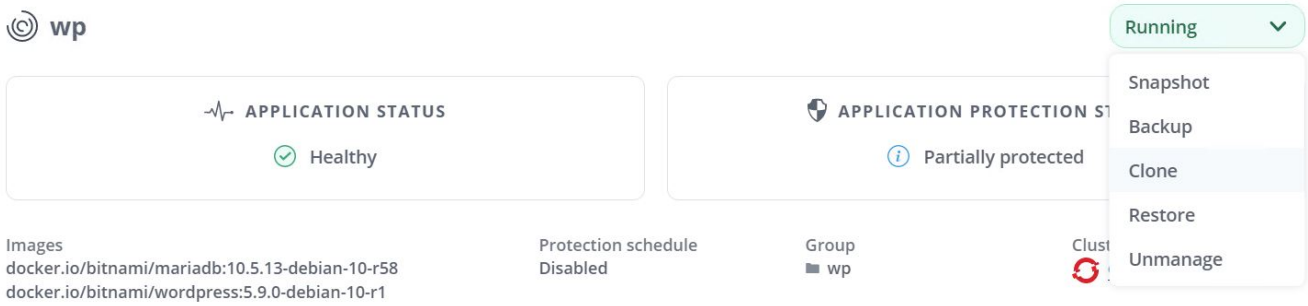
4. La nuova applicazione passa allo stato di ripristino mentre Astra Control Center ripristina l'applicazione sul cluster selezionato. Una volta installate e rilevate tutte le risorse dell'applicazione da Astra, l'applicazione passa allo stato Available (disponibile).



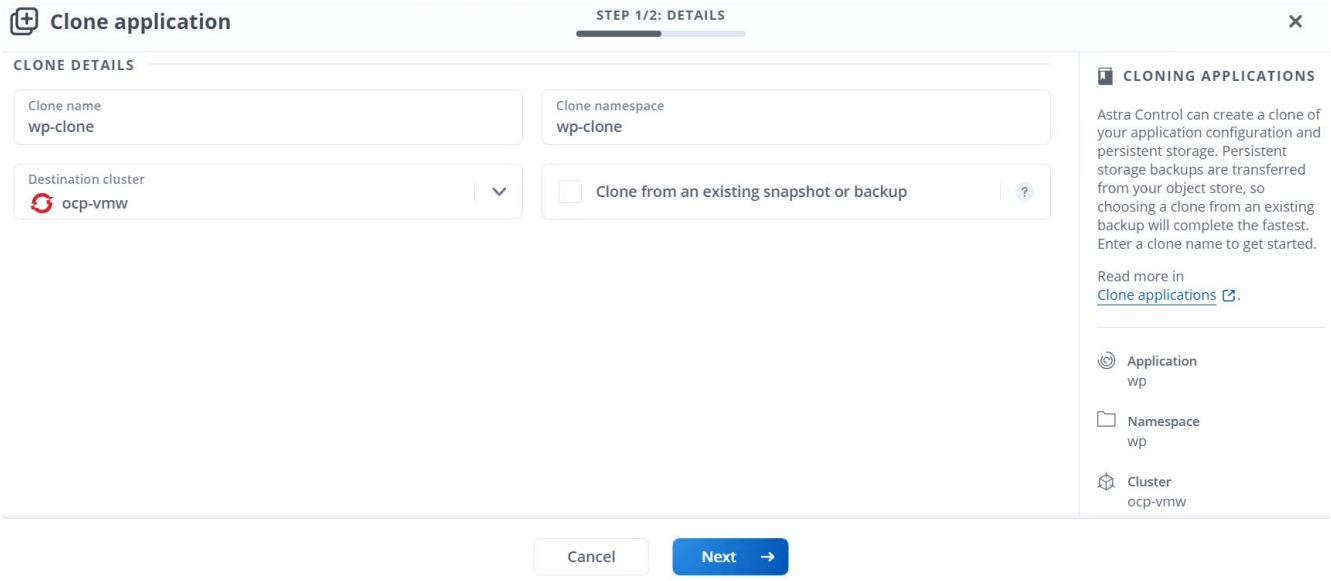
Clonare un'applicazione

È possibile clonare un'applicazione nel cluster di origine o in un cluster remoto per scopi di sviluppo/test o protezione dell'applicazione e disaster recovery. La clonazione di un'applicazione all'interno dello stesso cluster sullo stesso backend di storage utilizza la tecnologia NetApp FlexClone, che clona i PVC all'istante e consente di risparmiare spazio di storage.

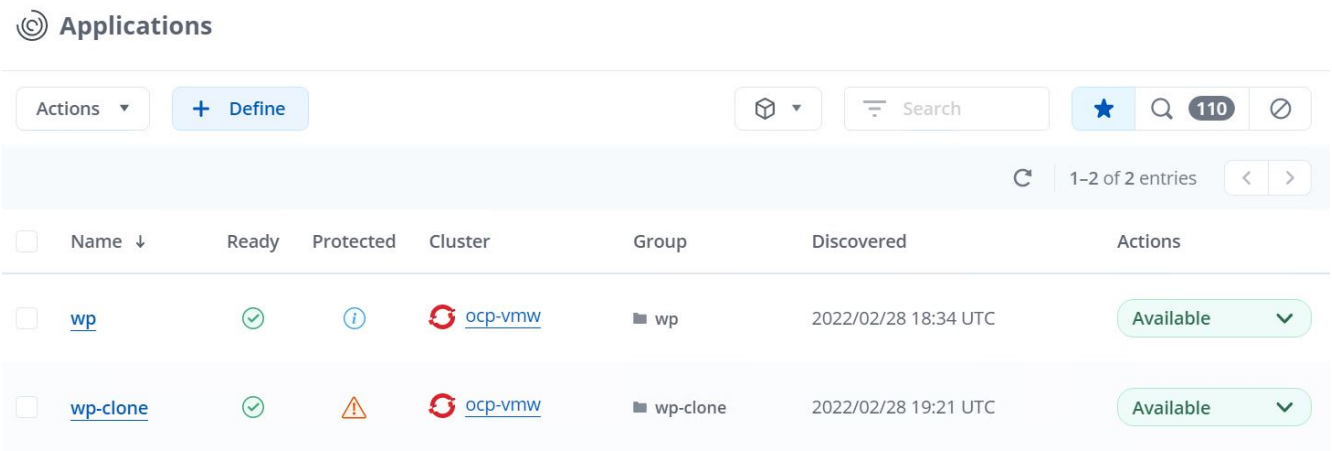
1. Per clonare un'applicazione, accedere alla scheda applicazioni > gestite e fare clic sull'applicazione in questione. Fare clic sul menu a discesa accanto al nome dell'applicazione e fare clic su Clone (Clona).



2. Immettere i dettagli del nuovo spazio dei nomi, selezionare il cluster in cui si desidera clonarlo e scegliere se clonarlo da uno snapshot esistente o da un backup o dallo stato corrente dell'applicazione. Quindi, fare clic su Next (Avanti) e su Clone on review pane (Clona sul pannello di revisione) dopo aver esaminato i dettagli.



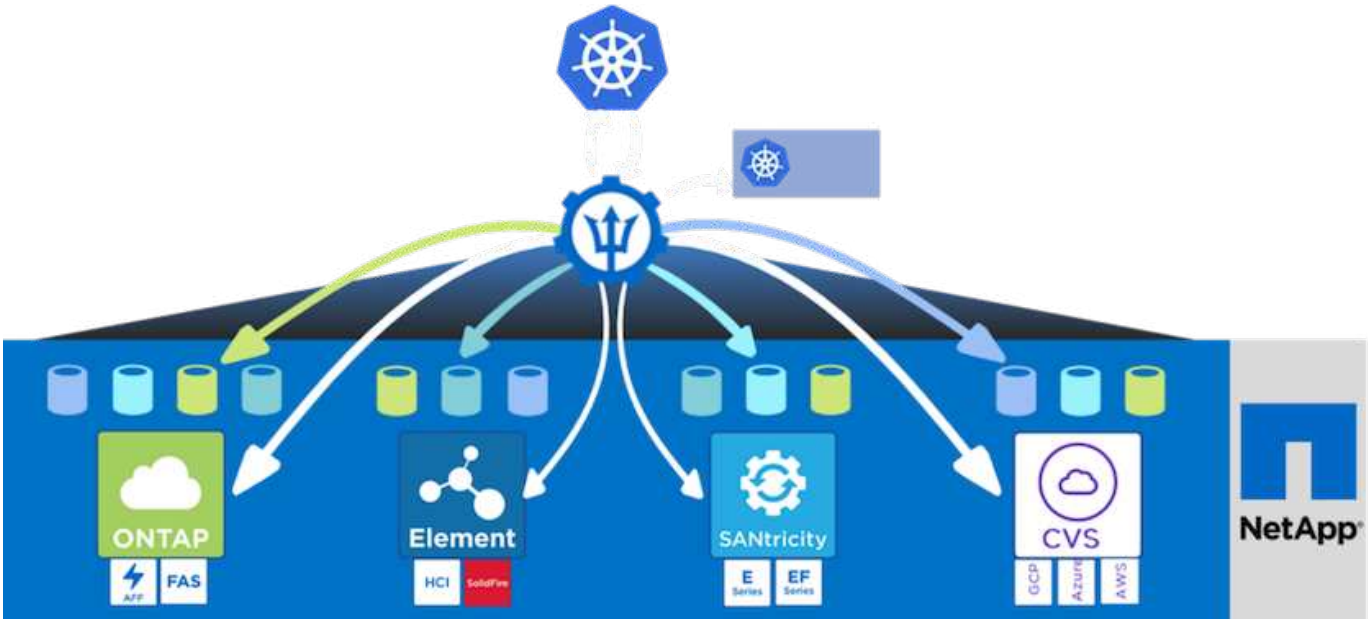
3. La nuova applicazione passa allo stato di rilevamento mentre Astra Control Center crea l'applicazione sul cluster selezionato. Una volta installate e rilevate tutte le risorse dell'applicazione da Astra, l'applicazione passa allo stato Available (disponibile).



Panoramica di Astra Trident

Astra Trident è un orchestrator di storage open-source e completamente supportato per container e distribuzioni Kubernetes, incluso Red Hat OpenShift. Trident lavora con l'intero portfolio di storage NetApp, inclusi i sistemi storage NetApp ONTAP ed Element, e supporta anche connessioni NFS e iSCSI. Trident accelera il workflow DevOps consentendo agli utenti finali di eseguire il provisioning e gestire lo storage dai sistemi storage NetApp senza richiedere l'intervento di un amministratore dello storage.

Un amministratore può configurare una serie di backend di storage in base alle esigenze di progetto e ai modelli di sistemi di storage che consentono funzionalità di storage avanzate, tra cui compressione, tipi di dischi specifici o livelli di QoS che garantiscono un certo livello di performance. Una volta definiti, questi backend possono essere utilizzati dagli sviluppatori nei loro progetti per creare dichiarazioni di volume persistenti (PVC) e per collegare storage persistente ai propri container on-demand.



Astra Trident ha un rapido ciclo di sviluppo e, proprio come Kubernetes, viene rilasciato quattro volte all'anno.

L'ultima versione di Astra Trident è la 22.01 rilasciata a gennaio 2022. Matrice di supporto per quale versione

di Trident è stata testata con la quale è possibile trovare la distribuzione Kubernetes "qui".

A partire dalla versione 20.04, l'impostazione di Trident viene eseguita dall'operatore Trident. L'operatore semplifica le implementazioni su larga scala e fornisce supporto aggiuntivo, inclusa la riparazione automatica dei pod implementati nell'installazione di Trident.

Con la versione 21.01, è stato reso disponibile un grafico Helm per facilitare l'installazione dell'operatore Trident.

Scarica Astra Trident

Per installare Trident sul cluster di utenti implementato ed eseguire il provisioning di un volume persistente, attenersi alla seguente procedura:

1. Scaricare l'archivio di installazione sulla workstation di amministrazione ed estrarre il contenuto. La versione corrente di Trident è la 22.01, che può essere scaricata "qui".

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

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

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

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

2. Estrarre l'installazione di Trident dal bundle scaricato.

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

Installare l'operatore Trident con Helm

1. Innanzitutto, impostare la posizione del cluster utente `kubeconfig` File come variabile di ambiente in modo da non doverla fare riferimento, perché Trident non ha alcuna opzione per passare questo file.

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

2. Eseguire il comando Helm per installare l'operatore Trident dal tarball nella directory helm durante la creazione dello spazio dei nomi Trident nel cluster di utenti.

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

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

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

To learn more about the release, try:

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

3. È possibile verificare che Trident sia installato correttamente controllando i pod in esecuzione nello spazio dei nomi o utilizzando il binario tridentctl per controllare la versione installata.

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

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

In alcuni casi, gli ambienti dei clienti potrebbero richiedere la personalizzazione dell'implementazione di Trident. In questi casi, è anche possibile installare manualmente l'operatore Trident e aggiornare i manifesti inclusi per personalizzare l'implementazione.

Installare manualmente l'operatore Trident

1. Innanzitutto, impostare la posizione del cluster utente `kubeconfig` File come variabile di ambiente in modo da non doverla fare riferimento, perché Trident non ha alcuna opzione per passare questo file.

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

2. Il `trident-installer` la directory contiene i manifesti per la definizione di tutte le risorse richieste. Utilizzando i manifesti appropriati, creare `TridentOrchestrator` definizione personalizzata delle risorse.

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

3. Se non ne esiste uno, creare uno spazio dei nomi Trident nel cluster utilizzando il manifesto fornito.

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

4. Creare le risorse necessarie per l'implementazione dell'operatore Trident, ad esempio un `ServiceAccount` per l'operatore, un `ClusterRole` e. `ClusterRoleBinding` al `ServiceAccount`, un

dedicato `PodSecurityPolicy`o l'operatore stesso.

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

5. È possibile controllare lo stato dell'operatore dopo l'implementazione con i seguenti comandi:

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

6. Con l'implementazione dell'operatore, ora possiamo utilizzarlo per installare Trident. Per eseguire questa operazione, è necessario creare un `TridentOrchestrator`.

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

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

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

7. È possibile verificare che Trident sia installato correttamente controllando i pod in esecuzione nello spazio dei nomi o utilizzando il binario tridentctl per controllare la versione installata.

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

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

Preparare i nodi di lavoro per lo storage

NFS

La maggior parte delle distribuzioni Kubernetes viene fornita con i pacchetti e le utility per montare i backend NFS installati di default, incluso Red Hat OpenShift.

Tuttavia, per NFSv3, non esiste alcun meccanismo per negoziare la concorrenza tra il client e il server. Pertanto, il numero massimo di voci della tabella degli slot sunrpc lato client deve essere sincronizzato manualmente con il valore supportato sul server per garantire le migliori prestazioni per la connessione NFS

senza che il server debba ridurre le dimensioni della finestra della connessione.

Per ONTAP, il numero massimo supportato di voci della tabella degli slot sunrpc è 128, ovvero ONTAP può gestire 128 richieste NFS simultanee alla volta. Tuttavia, per impostazione predefinita, Red Hat CoreOS/Red Hat Enterprise Linux ha un massimo di 65,536 voci della tabella degli slot sunrpc per connessione. È necessario impostare questo valore su 128 e questo può essere fatto usando Machine Config Operator (MCO) in OpenShift.

Per modificare il numero massimo di voci della tabella degli slot sunrpc nei nodi di lavoro OpenShift, attenersi alla seguente procedura:

1. Accedere alla console Web di OCP e selezionare Compute > Machine Configs (calcolo > configurazioni macchina). Fare clic su Create Machine Config. Copiare e incollare il file YAML e fare clic su Create (Crea).

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

2. Dopo aver creato l'MCO, la configurazione deve essere applicata a tutti i nodi di lavoro e riavviata uno alla volta. L'intero processo richiede da 20 a 30 minuti circa. Verificare se la configurazione del computer viene applicata utilizzando `oc get mcp` e assicurarsi che il pool di configurazione del computer per i lavoratori sia aggiornato.

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

ISCSI

Per preparare i nodi di lavoro per consentire la mappatura dei volumi di storage a blocchi tramite il protocollo iSCSI, è necessario installare i pacchetti necessari per supportare tale funzionalità.

In Red Hat OpenShift, questo viene gestito applicando un MCO (Machine Config Operator) al cluster dopo averlo implementato.

Per configurare i nodi di lavoro per l'esecuzione dei servizi iSCSI, attenersi alla seguente procedura:

1. Accedere alla console Web di OCP e selezionare Compute > Machine Configs (calcolo > configurazioni macchina). Fare clic su Create Machine Config. Copiare e incollare il file YAML e fare clic su Create (Crea).

   Quando non si utilizza il multipathing:

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

   Quando si utilizza il multipathing:

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

2. Una volta creata la configurazione, sono necessari circa 20 - 30 minuti per applicarla ai nodi di lavoro e ricaricarla. Verificare se la configurazione del computer viene applicata utilizzando `oc get mcp` e assicurarsi che il pool di configurazione del computer per i lavoratori sia aggiornato. È inoltre possibile accedere ai nodi di lavoro per confermare che il servizio iscsid è in esecuzione (e il servizio multipath è in esecuzione se si utilizza il multipathing).

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

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

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

È inoltre possibile confermare che MachineConfig sia stato applicato correttamente e che i servizi siano stati avviati come previsto eseguendo il `oc debug` con i flag appropriati.

Creazione di backend per il sistema storage

Dopo aver completato l'installazione di Astra Trident Operator, è necessario configurare il backend per la piattaforma di storage NetApp specifica in uso. Seguire i collegamenti riportati di seguito per continuare

l'installazione e la configurazione di Astra Trident.

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

Configurazione NFS di NetApp ONTAP

Per consentire l'integrazione di Trident con il sistema storage NetApp ONTAP, è necessario creare un backend che consenta la comunicazione con il sistema storage.

1. Nell'archivio di installazione scaricato in sono disponibili file backend di esempio `sample-input` gerarchia di cartelle. Per i sistemi NetApp ONTAP che servono NFS, copiare il `backend-ontap-nas.json` nella directory di lavoro e modificare il file.

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

2. Modificare il backendName, managementLIF, dataLIF, svm, nome utente, e password in questo file.

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

> È consigliabile definire il valore backendName personalizzato come combinazione di storageDriverName e dataLIF che fornisce NFS per una facile identificazione.

3. Una volta creato questo file di back-end, eseguire il comando seguente per creare il primo backend.

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

4. Una volta creato il backend, è necessario creare una classe di storage. Come per il backend, esiste un file di esempio della classe di storage che può essere modificato per l'ambiente disponibile nella cartella di input di esempio. Copiarlo nella directory di lavoro e apportare le modifiche necessarie per riflettere il backend creato.

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

5. L'unica modifica che deve essere apportata a questo file è definire `backendType` valore al nome del driver di storage dal backend appena creato. Annotare anche il valore del campo nome, a cui si deve fare riferimento in un passaggio successivo.

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

> Esiste un campo opzionale chiamato `fsType` definito in questo file. Questa riga può essere eliminata nei backend NFS.

6. Eseguire `oc` per creare la classe di storage.

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

7. Una volta creata la classe di storage, è necessario creare la prima dichiarazione di volume persistente (PVC). C'è un esempio `pvc-basic.yaml` file che può essere utilizzato per eseguire questa azione, disponibile anche in input di esempio.

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

8. L'unica modifica che deve essere apportata a questo file è garantire che il `storageClassName` il campo corrisponde a quello appena creato. La definizione PVC può essere ulteriormente personalizzata in base alle esigenze del carico di lavoro da fornire.

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

9. Creare il PVC emettendo il `oc` comando. La creazione può richiedere del tempo a seconda delle dimensioni del volume di backup da creare, in modo da poter guardare il processo mentre viene completato.

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

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

Configurazione iSCSI di NetApp ONTAP

Per consentire l'integrazione di Trident con il sistema storage NetApp ONTAP, è necessario creare un backend che consenta la comunicazione con il sistema storage.

1. Nell'archivio di installazione scaricato in sono disponibili file backend di esempio `sample-input` gerarchia di cartelle. Per i sistemi NetApp ONTAP che utilizzano iSCSI, copiare il `backend-ontap-san.json` nella directory di lavoro e modificare il file.

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

2. Modificare i valori di gestione LIF, dataLIF, svm, nome utente e password in questo file.

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

3. Una volta creato questo file di back-end, eseguire il comando seguente per creare il primo backend.

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

4. Una volta creato il backend, è necessario creare una classe di storage. Come per il backend, esiste un file di esempio della classe di storage che può essere modificato per l'ambiente disponibile nella cartella di input di esempio. Copiarlo nella directory di lavoro e apportare le modifiche necessarie per riflettere il backend creato.

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

5. L'unica modifica che deve essere apportata a questo file è definire `backendType` valore al nome del driver di storage dal backend appena creato. Annotare anche il valore del campo nome, a cui si deve fare riferimento in un passaggio successivo.

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

> Esiste un campo opzionale chiamato `fsType` definito in questo file. Nei backend iSCSI, questo valore può essere impostato su un tipo di filesystem Linux specifico (XFS, ext4, ecc.) o può essere cancellato per consentire a OpenShift di decidere quale filesystem usare.

6. Eseguire `oc` per creare la classe di storage.

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

7. Una volta creata la classe di storage, è necessario creare la prima dichiarazione di volume persistente (PVC). C'è un esempio `pvc-basic.yaml` file che può essere utilizzato per eseguire questa azione, disponibile anche in input di esempio.

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

8. L'unica modifica che deve essere apportata a questo file è garantire che il `storageClassName` il campo corrisponde a quello appena creato. La definizione PVC può essere ulteriormente personalizzata in base alle esigenze del carico di lavoro da fornire.

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

9. Creare il PVC emettendo il `oc` comando. La creazione può richiedere del tempo a seconda delle

dimensioni del volume di backup da creare, in modo da poter guardare il processo mentre viene completato.

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

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

Configurazione iSCSI NetApp Element

Per abilitare l'integrazione di Trident con il sistema storage NetApp Element, è necessario creare un backend che consenta la comunicazione con il sistema storage utilizzando il protocollo iSCSI.

1. Nell'archivio di installazione scaricato in sono disponibili file backend di esempio `sample-input` gerarchia di cartelle. Per i sistemi NetApp Element che utilizzano iSCSI, copiare `backend-solidfire.json` nella directory di lavoro e modificare il file.

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

   a. Modificare i valori di utente, password e MVIP su `EndPoint` linea.
   b. Modificare il `SVIP` valore.

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

2. Una volta creato questo file back-end, eseguire il seguente comando per creare il primo backend.

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

3. Una volta creato il backend, è necessario creare una classe di storage. Come per il backend, esiste un file di esempio della classe di storage che può essere modificato per l'ambiente disponibile nella cartella di input di esempio. Copiarlo nella directory di lavoro e apportare le modifiche necessarie per riflettere il backend creato.

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

4. L'unica modifica che deve essere apportata a questo file è definire `backendType` valore al nome del driver di storage dal backend appena creato. Annotare anche il valore del campo nome, a cui si deve fare riferimento in un passaggio successivo.

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

> Esiste un campo opzionale chiamato `fsType` definito in questo file. Nei backend iSCSI, questo valore può essere impostato su un tipo di filesystem Linux specifico (XFS, ext4 e così via), oppure può essere cancellato per consentire a OpenShift di decidere quale filesystem usare.

5. Eseguire `oc` per creare la classe di storage.

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

6. Una volta creata la classe di storage, è necessario creare la prima dichiarazione di volume persistente (PVC). C'è un esempio `pvc-basic.yaml` file che può essere utilizzato per eseguire questa azione, disponibile anche in input di esempio.

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

7. L'unica modifica che deve essere apportata a questo file è garantire che il `storageClassName` il campo corrisponde a quello appena creato. La definizione PVC può essere ulteriormente personalizzata in base alle esigenze del carico di lavoro da fornire.

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

8. Creare il PVC emettendo il `oc` comando. La creazione può richiedere del tempo a seconda delle dimensioni del volume di backup da creare, in modo da poter guardare il processo mentre viene completato.

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

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

Opzioni di configurazione avanzate

Analisi delle opzioni di bilanciamento del carico: Red Hat OpenShift con NetApp

Nella maggior parte dei casi, Red Hat OpenShift rende le applicazioni disponibili al mondo esterno attraverso i percorsi. Un servizio viene esposto assegnandogli un nome host raggiungibile esternamente. Il percorso definito e gli endpoint identificati dal servizio possono essere utilizzati da un router OpenShift per fornire questa connettività denominata ai client esterni.

Tuttavia, in alcuni casi, le applicazioni richiedono l'implementazione e la configurazione di bilanciatori di carico personalizzati per esporre i servizi appropriati. Un esempio è NetApp Astra Control Center. Per soddisfare questa esigenza, abbiamo valutato diverse opzioni di bilanciamento del carico personalizzate. L'installazione e la configurazione sono descritte in questa sezione.

Le seguenti pagine contengono informazioni aggiuntive sulle opzioni di bilanciamento del carico validate nella soluzione Red Hat OpenShift con NetApp:

- "MetalLB"
- "F5 BIG-IP"

Installazione di bilanciatori di carico MetalLB: Red Hat OpenShift con NetApp

Questa pagina elenca le istruzioni di installazione e configurazione per il bilanciamento del carico MetalLB.

MetalLB è un bilanciamento del carico di rete self-hosting installato sul cluster OpenShift che consente la creazione di servizi OpenShift di bilanciamento del carico di tipo in cluster che non vengono eseguiti su un provider cloud. Le due funzionalità principali di MetalLB che lavorano insieme per supportare i servizi LoadBalancer sono l'allocazione degli indirizzi e l'annuncio esterno.

Opzioni di configurazione di MetalLB

In base al modo in cui MetalLB annuncia l'indirizzo IP assegnato ai servizi LoadBalancer all'esterno del cluster OpenShift, opera in due modalità:

- **Layer 2 mode.** in questa modalità, un nodo del cluster OpenShift assume la proprietà del servizio e risponde alle richieste ARP per quell'IP per renderlo raggiungibile all'esterno del cluster OpenShift. Poiché solo il nodo annuncia l'IP, presenta un collo di bottiglia nella larghezza di banda e limitazioni di failover lente. Per ulteriori informazioni, consultare la documentazione "qui".
- **Modalità BGP.** in questa modalità, tutti i nodi del cluster OpenShift stabiliscono sessioni di peering BGP con un router e pubblicizzano i route per inoltrare il traffico agli IP del servizio. Il prerequisito per questa operazione è l'integrazione di MetalLB con un router in tale rete. A causa del meccanismo di hashing in BGP, il mapping IP-to-Node per un servizio presenta una certa limitazione. Per ulteriori informazioni, consultare la documentazione "qui".

> Ai fini di questo documento, stiamo configurando MetalLB in modalità Layer-2.

Installazione del bilanciamento del carico MetalLB

1. Scarica le risorse di MetalLB.

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

2. Modificare il file `metallb.yaml` e rimuovere `spec.template.spec.securityContext` Da Controller Deployment e dal DemonSet dell'oratore.

   Righe da eliminare:

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

3. Creare il `metallb-system` namespace.

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

4. Creare il CR MetalLB.

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

5. Prima di configurare l'altoparlante MetalLB, concedere al relatore i privilegi elevati DemonSet in modo che possa eseguire la configurazione di rete richiesta per far funzionare i bilanciatori di carico.

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

6. Configurare MetalLB creando un `ConfigMap` in `metallb-system` namespace.

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

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

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

7. Ora, quando vengono creati i servizi loadbalancer, MetalLB assegna un IP esterno ai servizi e annuncia l'indirizzo IP rispondendo alle richieste ARP.

> Se si desidera configurare MetalLB in modalità BGP, saltare il punto 6 e seguire la procedura nella documentazione di MetalLB "qui".

Installazione di F5 BIG-IP Load Balancer

F5 BIG-IP è un Application Delivery Controller (ADC) che offre un'ampia gamma di servizi avanzati di gestione del traffico e sicurezza di livello produttivo come il bilanciamento del carico L4-L7, l'offload SSL/TLS, DNS, firewall e molto altro ancora. Questi servizi aumentano drasticamente la disponibilità, la sicurezza e le performance delle tue applicazioni.

F5 BIG-IP può essere implementato e utilizzato in vari modi, su hardware dedicato, nel cloud o come appliance virtuale on-premise. Fare riferimento alla documentazione qui per esplorare e implementare F5 BIG-IP in base ai requisiti.

Per un'integrazione efficiente dei servizi Big-IP di F5 con Red Hat OpenShift, F5 offre IL BIG-IP Container Ingress Service (CIS). CIS viene installato come controller pod che controlla l'API OpenShift per alcune

definizioni di risorse personalizzate (CRD) e gestisce la configurazione del sistema F5 BIG-IP. F5 BIG-IP CIS può essere configurato per controllare i tipi di servizio LoadBalancer e route in OpenShift.

Inoltre, per l'allocazione automatica dell'indirizzo IP al servizio del tipo LoadBalancer, è possibile utilizzare il controller F5 IPAM. Il controller F5 IPAM viene installato come controller pod che controlla i servizi di OpenShift API per LoadBalancer con un'annotazione ipamLabel per allocare l'indirizzo IP da un pool preconfigurato.

Questa pagina elenca le istruzioni di installazione e configurazione per i controller F5 BIG-IP CIS e IPAM. Come prerequisito, è necessario disporre di un sistema F5 BIG-IP distribuito e concesso in licenza. Deve inoltre essere concesso in licenza per i servizi SDN, inclusi per impostazione predefinita con LA licenza base BIG-IP VE.

> F5 BIG-IP può essere implementato in modalità standalone o cluster. Ai fini di questa convalida, F5 BIG-IP è stato implementato in modalità standalone, ma per scopi di produzione, è preferibile disporre di un cluster di big-IP per evitare un singolo punto di errore.

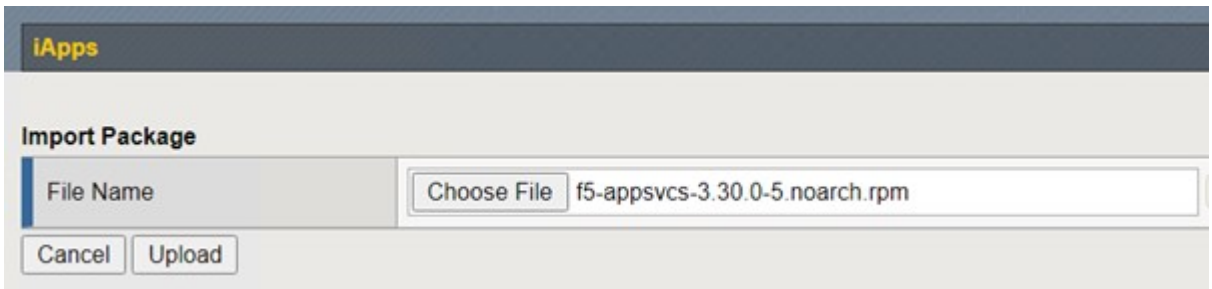
> Un sistema F5 BIG-IP può essere implementato su hardware dedicato, nel cloud o come appliance virtuale on-premise con versioni superiori alla 12.x per l'integrazione con F5 CIS. Ai fini di questo documento, il sistema F5 BIG-IP è stato validato come appliance virtuale, ad esempio utilizzando L'edizione BIG-IP VE.

Release validate

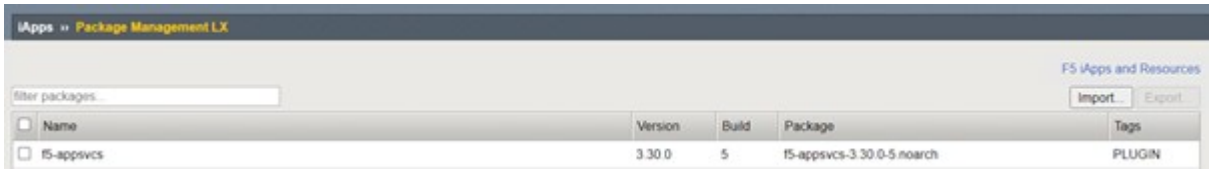
Tecnologia	Versione del software
Red Hat OpenShift	4.6 EUS, 4.7
F5 BIG-IP VE EDITION	16.1.0
F5 Container Ingress Service	2.5.1
F5 Controller IPAM	0.1.4
F5 AS3	3.30.0

Installazione

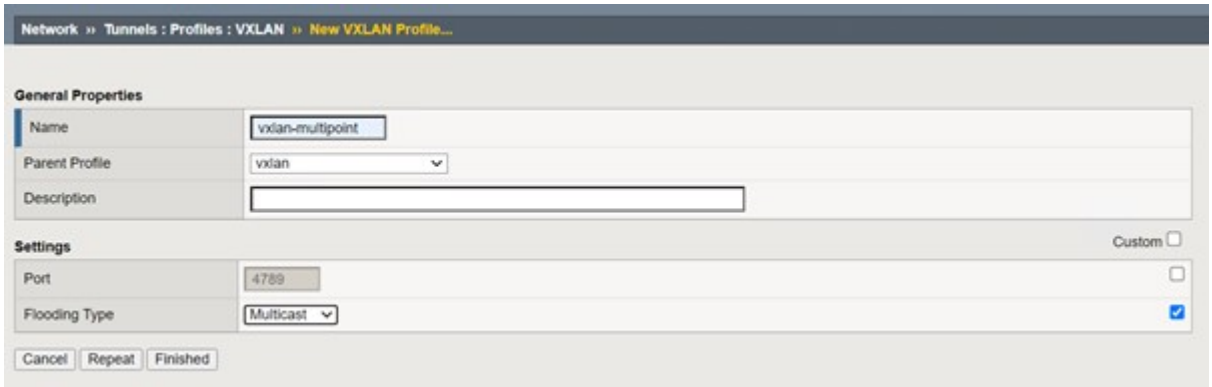
1. Installare l'estensione F5 Application Services 3 per consentire ai sistemi BIG-IP di accettare configurazioni in JSON invece di comandi imperativi. Passare a. "F5 repository AS3 GitHub"E scaricare il file RPM più recente.
2. Accedere al sistema F5 BIG-IP, accedere a iApps > Package Management LX e fare clic su Import (Importa).
3. Fare clic su Choose file (Scegli file) e selezionare il file RPM AS3 scaricato, fare clic su OK, quindi su Upload (carica).



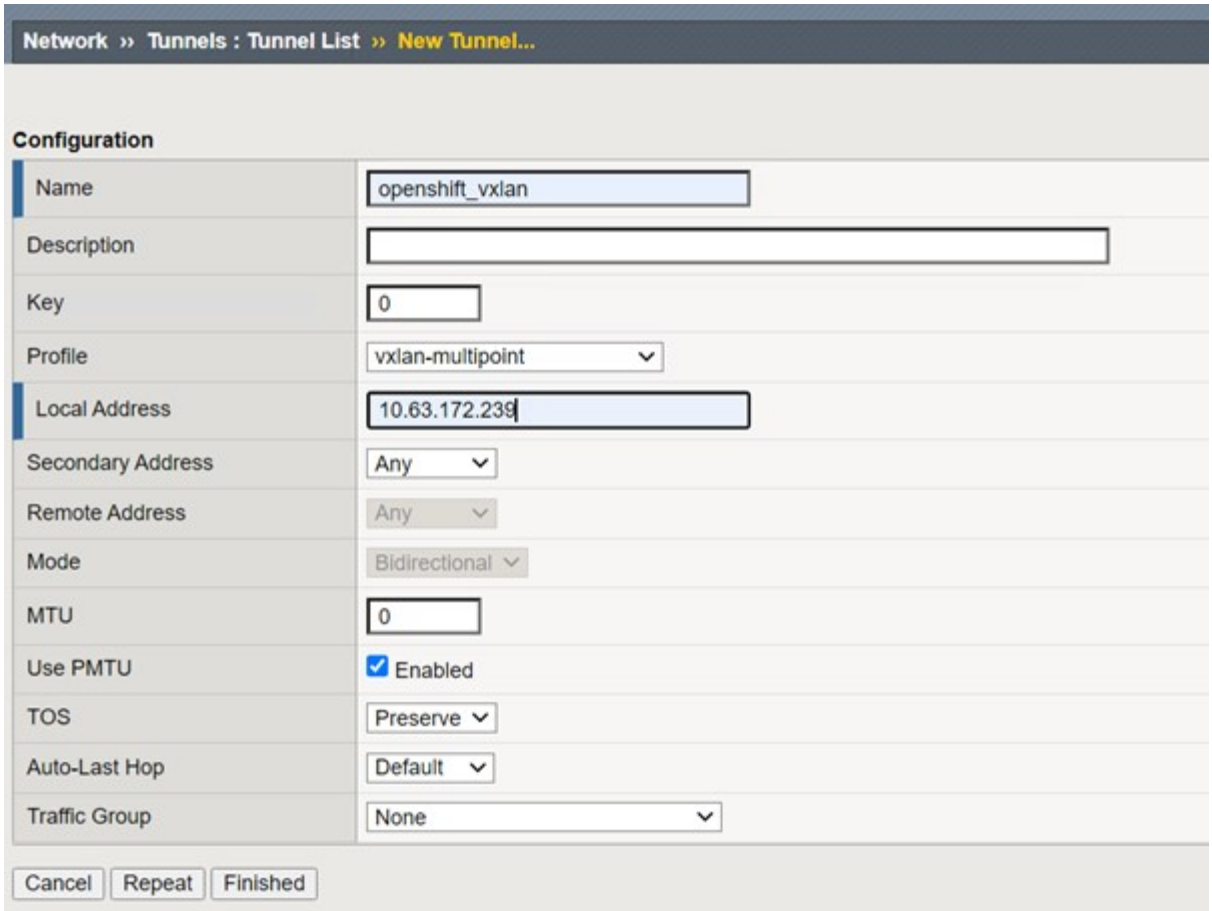
4. Verificare che l’estensione AS3 sia installata correttamente.



5. Quindi, configurare le risorse necessarie per la comunicazione tra OpenShift e I sistemi BIG-IP. Creare innanzitutto un tunnel tra OpenShift e IL SERVER BIG-IP creando un’interfaccia di tunnel VXLAN sul sistema BIG-IP per OpenShift SDN. Accedere a Network > Tunnels > Profiles (rete > tunnel > profili), fare clic su Create (Crea) e impostare il profilo principale su vxlan e il tipo di flooding su Multicast. Inserire un nome per il profilo e fare clic su fine.



6. Accedere a Network (rete) > Tunnels (tunnel) > Tunnel List (elenco tunnel), fare clic su Create (Crea) e immettere il nome e l’indirizzo IP locale per il tunnel. Selezionare il profilo di tunnel creato nel passaggio precedente e fare clic su fine.



7. Accedi al cluster Red Hat OpenShift con privilegi di amministratore del cluster.
8. Creare una subnet host su OpenShift per il server F5 BIG-IP, che estende la subnet dal cluster OpenShift al server F5 BIG-IP. Scaricare la definizione YAML della subnet host.

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

9. Modificare il file di sottorete host e aggiungere l'IP BIG-IP VTEP (tunnel VXLAN) per OpenShift SDN.

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

Modificare l’indirizzo e altri dettagli in base all’ambiente in uso.

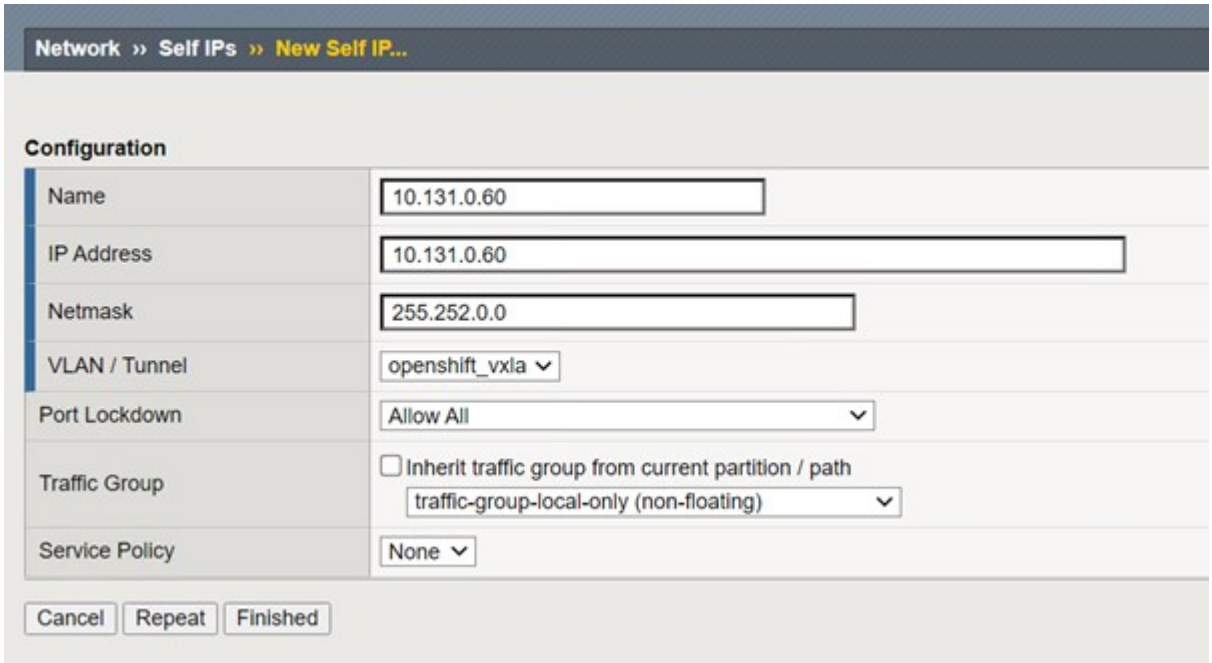
10. Creare la risorsa HostSubnet.

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

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

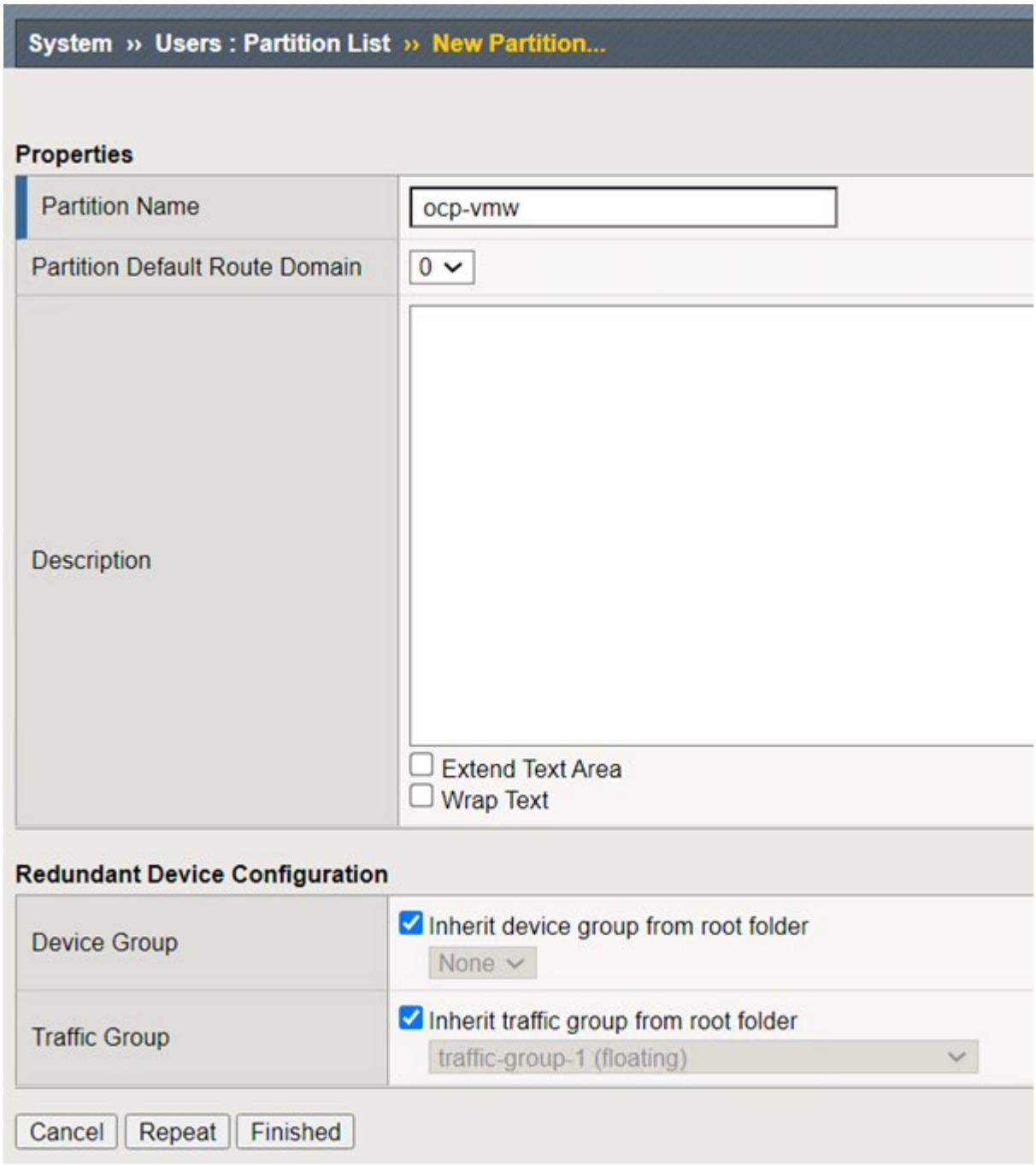
11. Ottenere l’intervallo di subnet IP del cluster per la subnet host creata per il server Big-IP F5.

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

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

12. Creare un IP self su OpenShift VXLAN con un IP nell'intervallo di subnet host di OpenShift corrispondente al server F5 BIG-IP. Accedere al sistema F5 BIG-IP, selezionare Network > Self IPs (rete > IP automatici) e fare clic su Create (Crea). Inserire un IP dalla subnet IP del cluster creata per la subnet host F5 BIG-IP, selezionare il tunnel VXLAN e immettere gli altri dettagli. Quindi fare clic su fine.



13. Creare una partizione nel sistema F5 BIG-IP da configurare e utilizzare con CIS. Accedere a sistema > utenti > elenco partizioni, fare clic su Crea e immettere i dettagli. Quindi fare clic su fine.



F5 consiglia di non eseguire alcuna configurazione manuale sulla partizione gestita da CIS.

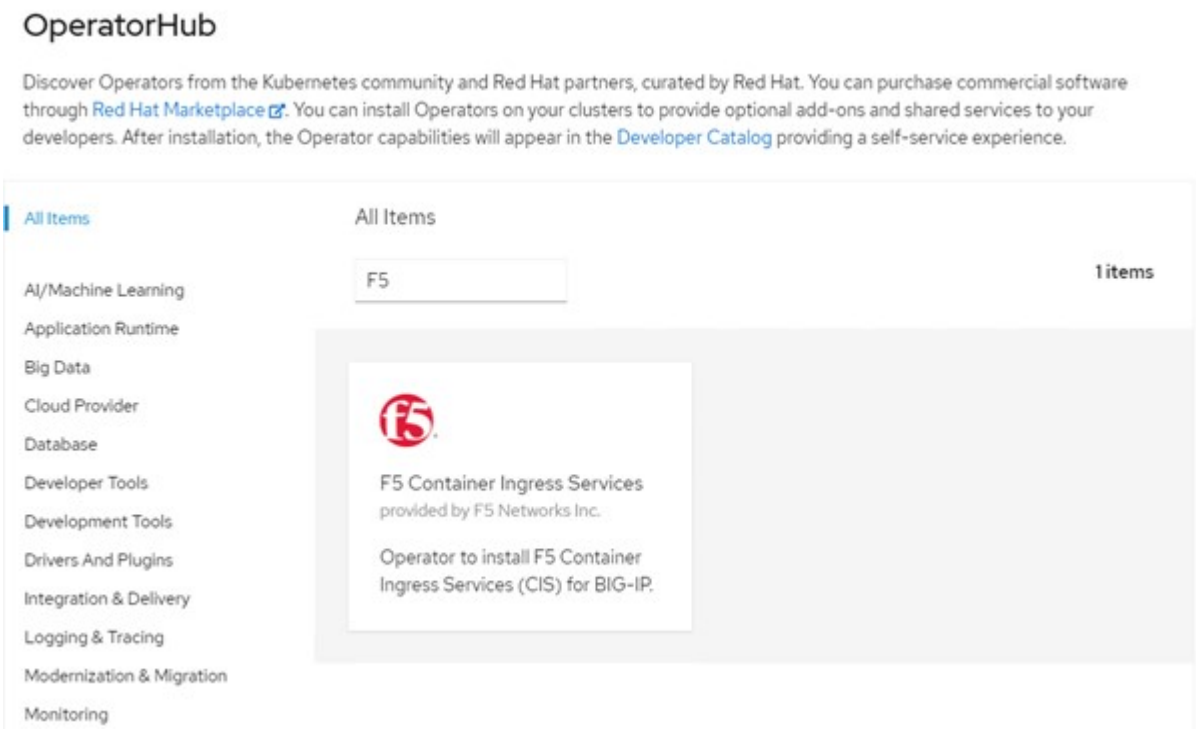
14. Installare F5 BIG-IP CIS utilizzando l'operatore di OperatorHub. Accedi al cluster Red Hat OpenShift con privilegi di amministrazione del cluster e crea un segreto con le credenziali di accesso del sistema F5 BIG-IP, un prerequisito per l'operatore.

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

secret/bigip-login created
```

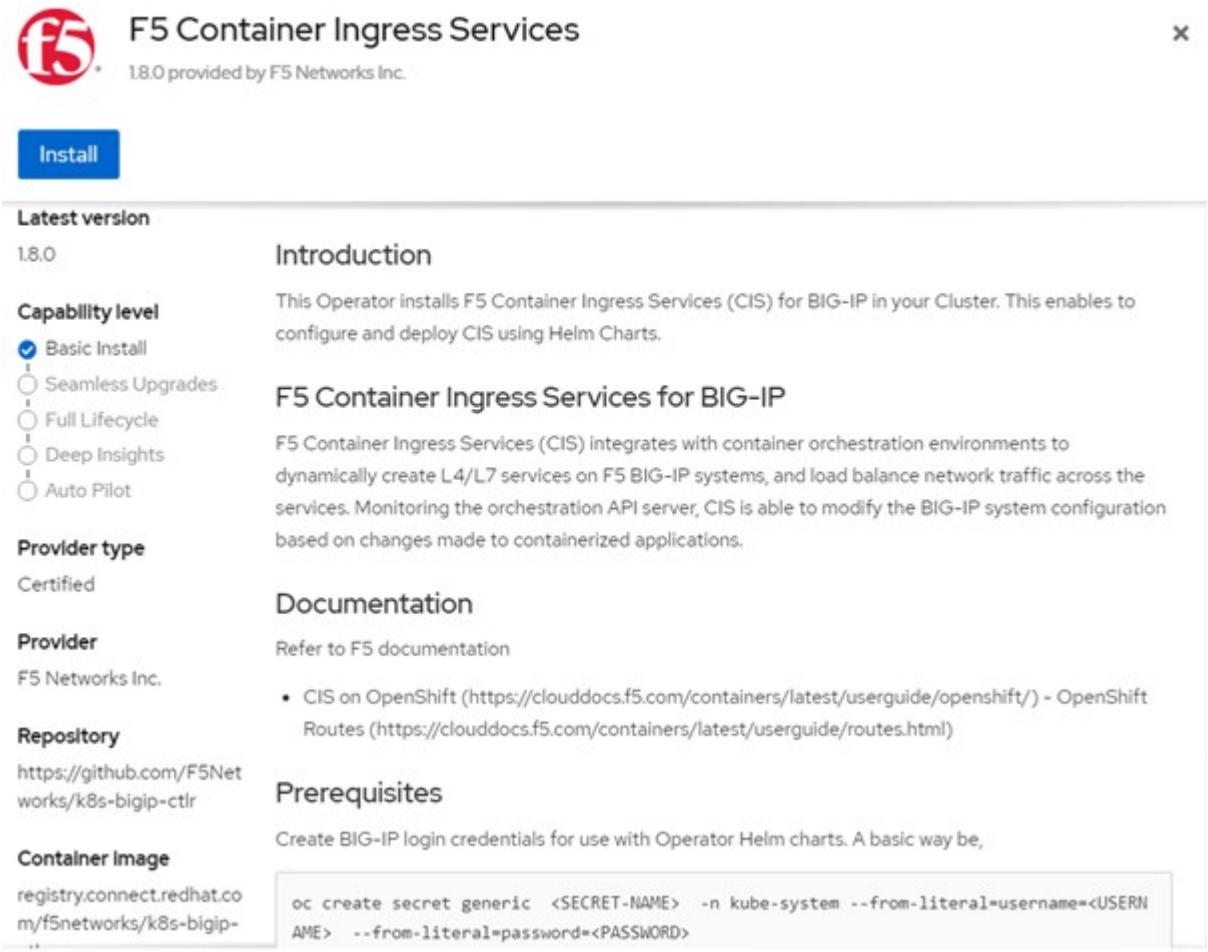
15. Installare F5 CIS CRD.

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

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

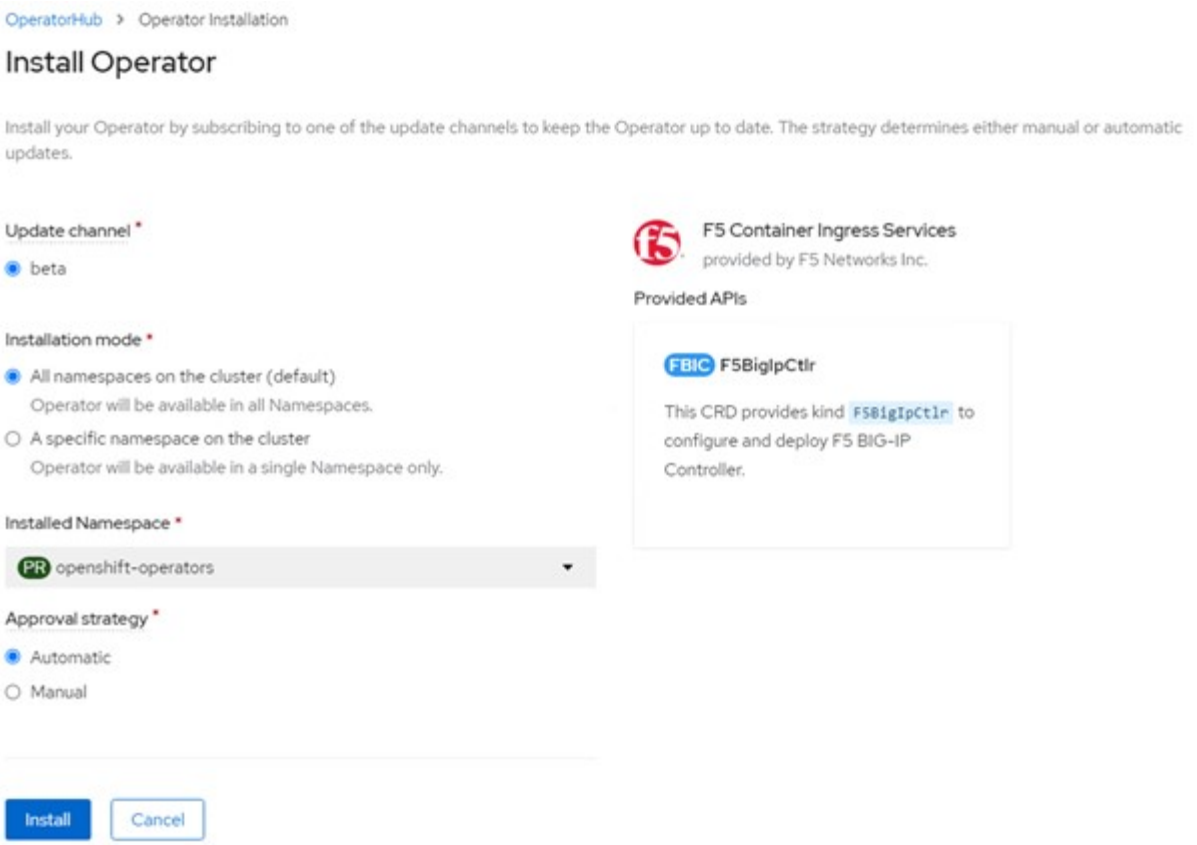
16. Accedere a Operator > OperatorHub, cercare la parola chiave F5 e fare clic sul riquadro F5 Container Ingress Service.



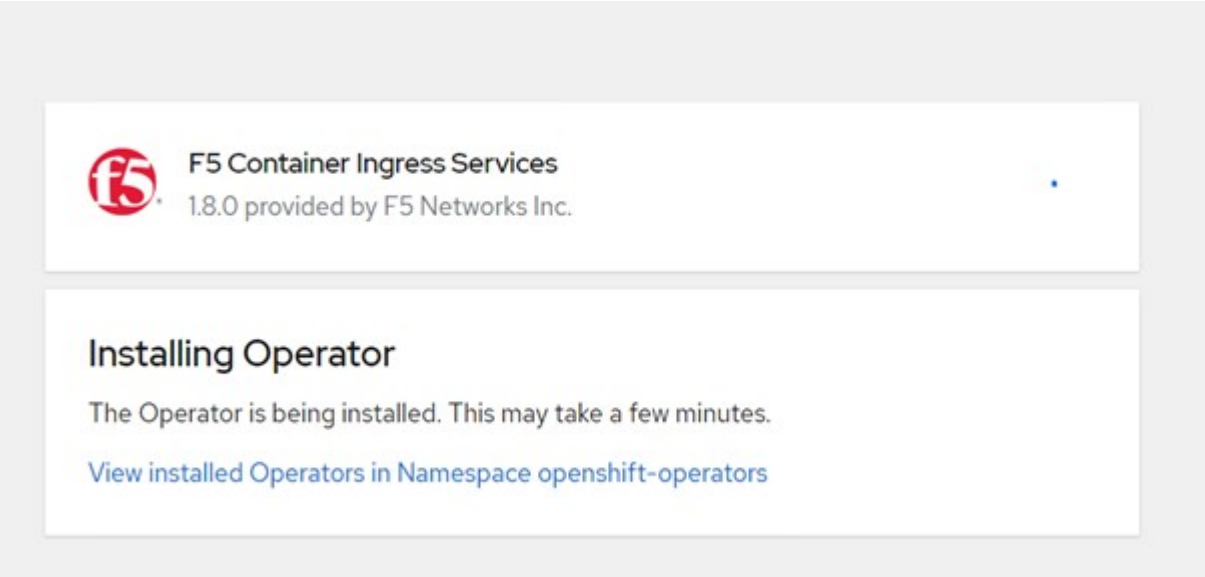
17. Leggere le informazioni dell’operatore e fare clic su Install (Installa).



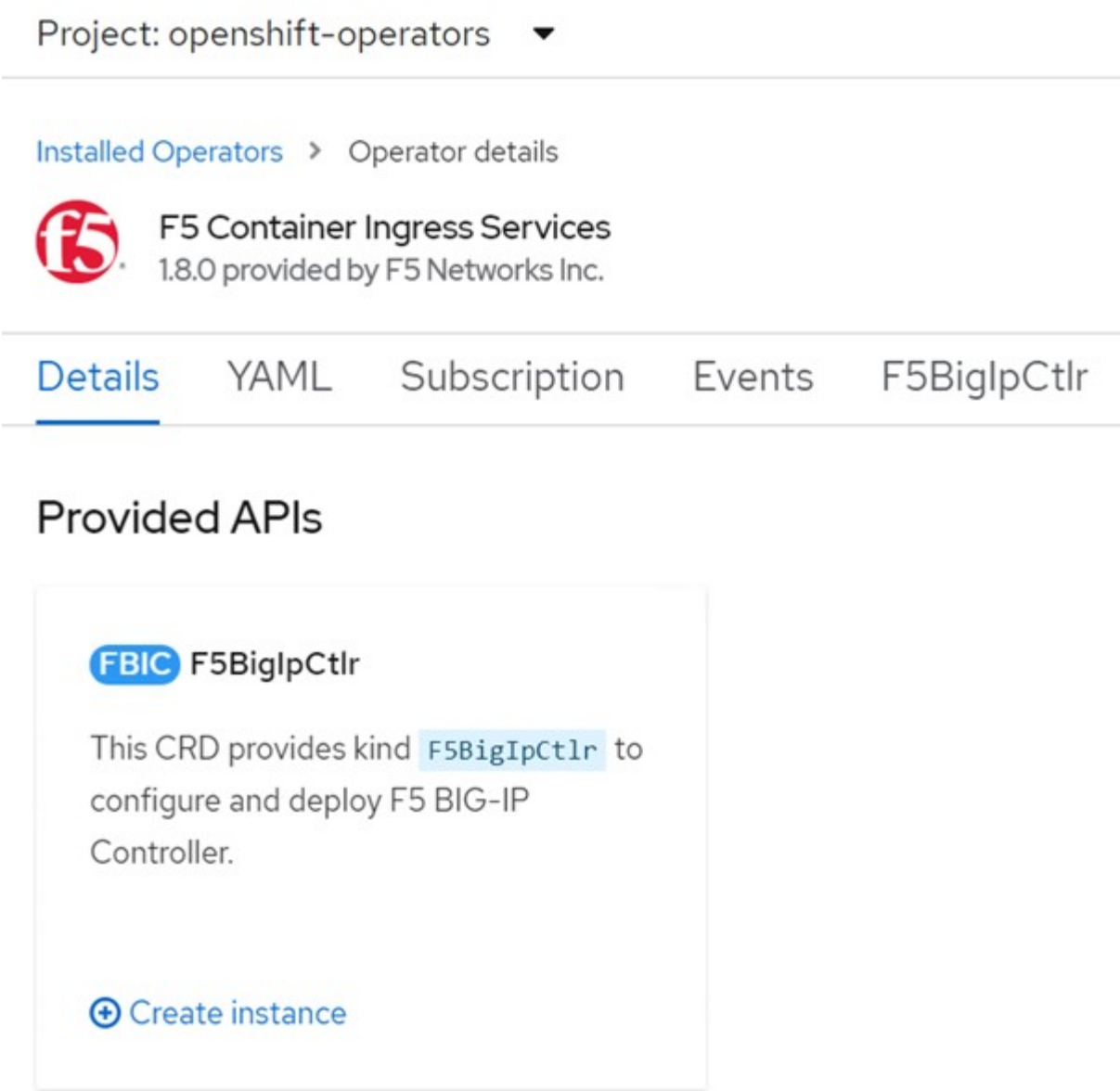
18. Nella schermata Install operator (Installa operatore), lasciare tutti i parametri predefiniti e fare clic su Install (Installa).



19. L’installazione dell’operatore richiede un po' di tempo.



20. Una volta installato l’operatore, viene visualizzato il messaggio Installazione completata.
21. Accedere a Operators > Installed Operators (operatori > operatori installati), fare clic su F5 Container Ingress Service (F5 Container Ingress Service), quindi fare clic su Create Instance (Crea istanza) nella sezione F5BigIpCtlr.

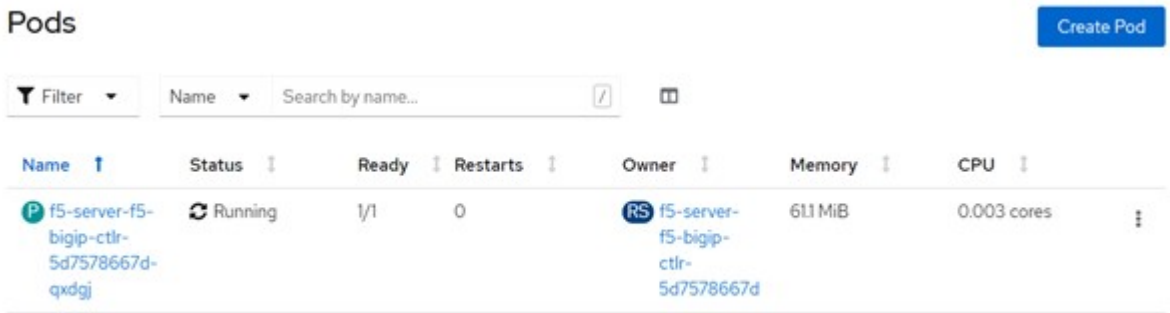


22. Fare clic su YAML View (Visualizza YAML) e incollare il seguente contenuto dopo aver aggiornato i parametri necessari.

> Aggiornare i parametri `bigip_partition`, ` openshift_sdn_name`, `bigip_url` e. `bigip_login_secret` di seguito per riflettere i valori per la configurazione prima di copiare il contenuto.

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

23. Dopo aver incollato questo contenuto, fare clic su Create (Crea). In questo modo vengono installati i pod CIS nello spazio dei nomi del sistema kube.



Red Hat OpenShift, per impostazione predefinita, fornisce un modo per esporre i servizi tramite route per il bilanciamento del carico L7. Un router OpenShift integrato è responsabile della pubblicità e della gestione del traffico per questi percorsi. Tuttavia, è anche possibile configurare F5 CIS per supportare i percorsi attraverso un sistema esterno F5 BIG-IP, che può essere eseguito come router ausiliario o come sostituto del router OpenShift self-hosting. CIS crea un server virtuale nel sistema BIG-IP che funge da router per i route OpenShift, mentre BIG-IP gestisce il routing degli annunci pubblicitari e del traffico. Fare riferimento alla documentazione qui per informazioni sui parametri per attivare questa funzione. Si noti che questi parametri sono definiti per la risorsa di implementazione OpenShift nell'API apps/v1. Pertanto, quando si utilizzano questi dati con l'API cis.f5.com/v1 della risorsa F5BigIpCtlr, sostituire i trattini (-) con i trattini (_) per i nomi dei parametri.

24. Gli argomenti passati alla creazione delle risorse CIS includono `ipam: true` e. `custom_resource_mode: true`. Questi parametri sono necessari per abilitare l'integrazione CIS con un controller IPAM. Verificare che il CIS abbia attivato l'integrazione IPAM creando la risorsa F5 IPAM.

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

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

25. Creare l'account del servizio, il ruolo e il rolebinding richiesti per il controller F5 IPAM. Creare un file YAML e incollare il seguente contenuto.

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

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

26. Creare le risorse.

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

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

27. Creare un file YAML e incollare la definizione di implementazione F5 IPAM fornita di seguito.

Aggiornare il parametro ip-range in spec.template.spec.containers[0].args di seguito per riflettere gli intervalli di indirizzi IP e ipamLabels corrispondenti alla configurazione.

IpamLabels [`range1` e. `range2` Nell'esempio seguente] devono essere annotati per i servizi di tipo LoadBalancer affinché il controller IPAM rilevi e assegni un indirizzo IP dall'intervallo definito.

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

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

28. Creare l’implementazione del controller F5 IPAM.

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

deployment/f5-ipam-controller created
```

29. Verificare che i controller pod F5 IPAM siano in esecuzione.

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

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

30. Creare lo schema F5 IPAM.

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

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

Verifica

1. Creare un servizio di tipo LoadBalancer

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

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

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

service/f5-demo-test created
```

2. Controllare se il controller IPAM assegna un indirizzo IP esterno.

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

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

3. Creare un’implementazione e utilizzare il servizio LoadBalancer creato.

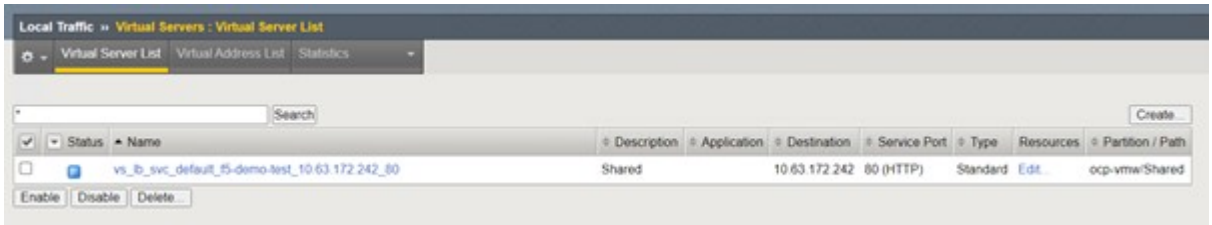
```
[admin@rhel-7 ~]$ vi example_deployment.yaml

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

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

deployment/f5-demo-test created
```

4. Verificare che i pod siano in funzione.

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

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

5. Controllare se il server virtuale corrispondente viene creato nel sistema BIG-IP per il servizio di tipo LoadBalancer in OpenShift. Accedere a traffico locale > Server virtuali > elenco server virtuali.



Creazione di registri immagine privati

Per la maggior parte delle implementazioni di Red Hat OpenShift, utilizzando un registro pubblico come "Quay.io" oppure "DockerHub" soddisfa la maggior parte delle esigenze dei clienti. Tuttavia, in alcuni casi un cliente potrebbe voler ospitare le proprie immagini private o personalizzate.

Questa procedura documenta la creazione di un registro di immagini privato supportato da un volume persistente fornito da Astra Trident e NetApp ONTAP.

> Astra Control Center richiede un registro per ospitare le immagini richieste dai container Astra. La sezione seguente descrive i passaggi per configurare un registro privato sul cluster Red Hat OpenShift e per inviare le immagini necessarie per supportare l'installazione di Astra Control Center.

Creazione Di un registro di immagine privato

1. Rimuovere l'annotazione predefinita dalla classe di storage predefinita corrente e annotare la classe di storage supportata da Trident come predefinita per il cluster OpenShift.

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

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

2. Modificare l'operatore di imageregistry immettendo i seguenti parametri di storage in `spec` sezione.

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

storage:
  pvc:
    claim:
```

3. Inserire i seguenti parametri in `spec` Sezione per la creazione di un percorso OpenShift con un nome host personalizzato. Salvare e uscire.

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

> La suddetta configurazione del percorso viene utilizzata quando si desidera un nome host personalizzato per il percorso. Se si desidera che OpenShift crei un percorso con un nome host predefinito, è possibile aggiungere i seguenti parametri a `spec` sezione: `defaultRoute: true`.

Certificati TLS personalizzati

Quando si utilizza un nome host personalizzato per il percorso, per impostazione predefinita, utilizza la configurazione TLS predefinita dell'operatore OpenShift Ingress. Tuttavia, è possibile aggiungere una configurazione TLS personalizzata al percorso. A tale scopo, attenersi alla seguente procedura.

a. Creare un segreto con i certificati TLS e la chiave del percorso.

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

b. Modificare l'operatore di imageregistry e aggiungere i seguenti parametri a `spec` sezione.

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

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

4. Modificare nuovamente l'operatore di imageregistry e modificare lo stato di gestione dell'operatore in `Managed` stato. Salvare e uscire.

```
oc edit configs.imageregistry/cluster

managementState: Managed
```

5. Se tutti i prerequisiti sono soddisfatti, PVC, POD e servizi vengono creati per il registro delle immagini private. In pochi minuti, il registro dovrebbe essere attivo.

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

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

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

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

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

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

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

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

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

6. Se si utilizzano i certificati TLS predefiniti per il percorso del Registro di sistema OpenShift dell'operatore di ingresso, è possibile recuperare i certificati TLS utilizzando il seguente comando.

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

7. Per consentire ai nodi OpenShift di accedere e estrarre le immagini dal Registro di sistema, aggiungere i certificati al client del docker sui nodi OpenShift. Creare una mappa di configurazione in `openshift-config` Namespace che utilizza i certificati TLS e lo patch alla configurazione dell'immagine del cluster per rendere attendibile il certificato.

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

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

8. Il registro interno di OpenShift è controllato dall'autenticazione. Tutti gli utenti di OpenShift possono accedere al registro di OpenShift, ma le operazioni che l'utente connesso può eseguire dipendono dalle autorizzazioni dell'utente.
   a. Per consentire a un utente o a un gruppo di utenti di estrarre immagini dal registro, agli utenti deve essere assegnato il ruolo di visualizzatore del registro.

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

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

   b. Per consentire a un utente o a un gruppo di utenti di scrivere o inviare immagini, agli utenti deve essere assegnato il ruolo di editor del Registro di sistema.

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

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

9. Per consentire ai nodi OpenShift di accedere al Registro di sistema e di eseguire il push o il pull delle immagini, è necessario configurare un pull secret.

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

10. Questo segreto pull può quindi essere patchato agli account di servizio o può essere referenziato nella definizione del pod corrispondente.
    a. Per applicare la patch agli account di servizio, eseguire il seguente comando.

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

b. Per fare riferimento al segreto pull nella definizione del pod, aggiungere il seguente parametro a `spec` sezione.

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

11. Per trasferire o estrarre un'immagine dalle workstation a parte il nodo OpenShift, attenersi alla seguente procedura.

a. Aggiungere i certificati TLS al client docker.

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

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

b. Accedere a OpenShift usando il comando oc login.

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

c. Accedere al registro utilizzando le credenziali utente di OpenShift con il comando podman/docker.

podman

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

+ NOTA: Se si utilizza `kubeadmin` per accedere al registro di sistema privato, quindi utilizzare il token invece della password.

docker

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

+ NOTA: Se si utilizza `kubeadmin` per accedere al registro di sistema privato, quindi utilizzare il token invece della password.

d. Premere o tirare le immagini.

podman

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

docker

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

Convalida della soluzione e casi d'utilizzo: Red Hat OpenShift con NetApp

Gli esempi forniti in questa pagina sono validazioni di soluzioni e casi di utilizzo per Red Hat OpenShift con NetApp.

- "Implementare una pipeline ci/CD Jenkins con storage persistente"
- "Configura la multitenancy su Red Hat OpenShift con NetApp"
- "Virtualizzazione Red Hat OpenShift con NetApp ONTAP"
- "Gestione avanzata dei cluster per Kubernetes su Red Hat OpenShift con NetApp"

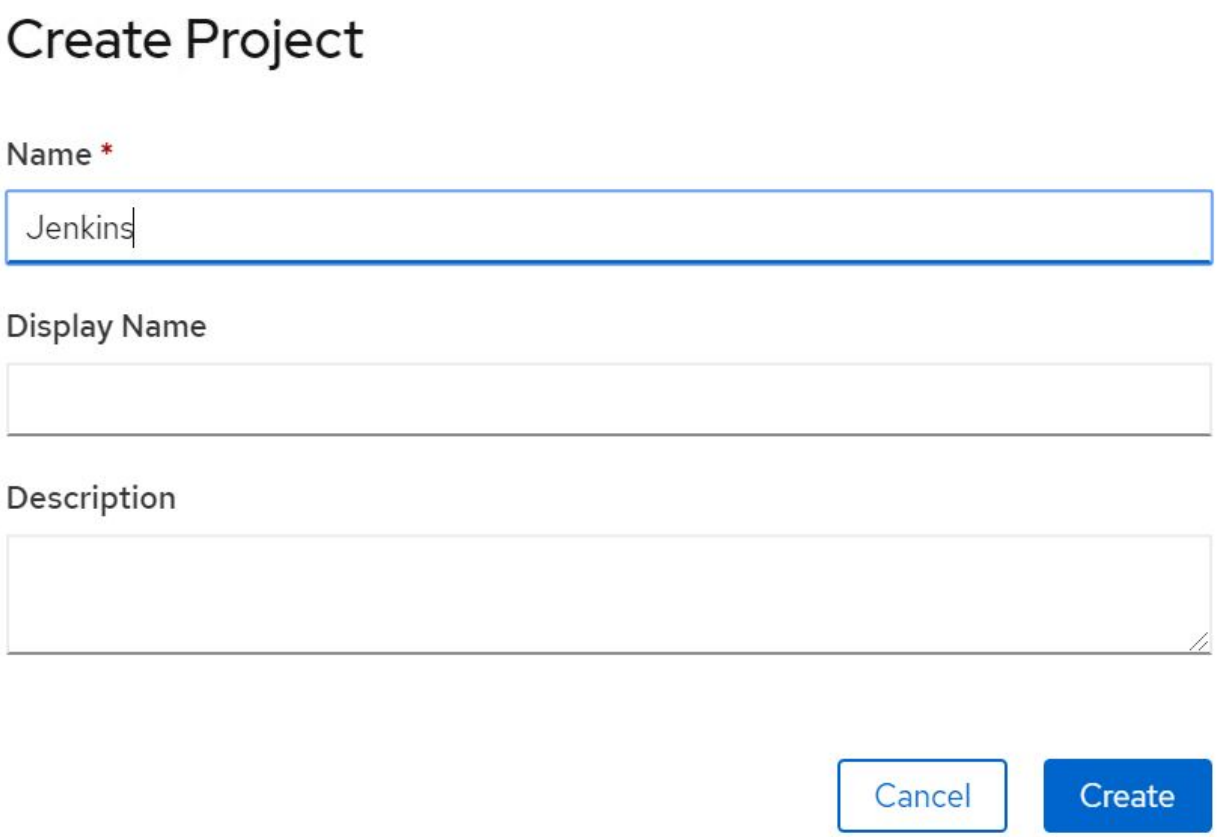
Implementa una pipeline ci/CD Jenkins con storage persistente: Red Hat OpenShift con NetApp

In questa sezione vengono fornite le fasi necessarie per implementare una pipeline ci/CD (Continuous Integration/Continuous Delivery or Deployment) con Jenkins per convalidare il funzionamento della soluzione.

Creare le risorse necessarie per l'implementazione di Jenkins

Per creare le risorse necessarie per l'implementazione dell'applicazione Jenkins, attenersi alla seguente procedura:

1. Crea un nuovo progetto chiamato Jenkins.



2. In questo esempio, abbiamo implementato Jenkins con storage persistente. Per supportare la build Jenkins, creare il PVC. Selezionare Storage > Persistent Volume Claims (Storage > Reclami volumi persistenti) e fare clic su Create Persistent Selezionare la classe di storage creata, assicurarsi che il nome della richiesta di rimborso del volume persistente sia jenkins, selezionare la dimensione e la modalità di accesso appropriate, quindi fare clic su Create (Crea).

Project: jenkins

Create Persistent Volume Claim

Edit YAML

Storage Class

SC basic

Storage class for the new claim.

Persistent Volume Claim Name *

jenkins

A unique name for the storage claim within the project.

Access Mode *

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

Permissions to the mounted drive.

Size *

100 GiB

Desired storage capacity.

Use label selectors to request storage

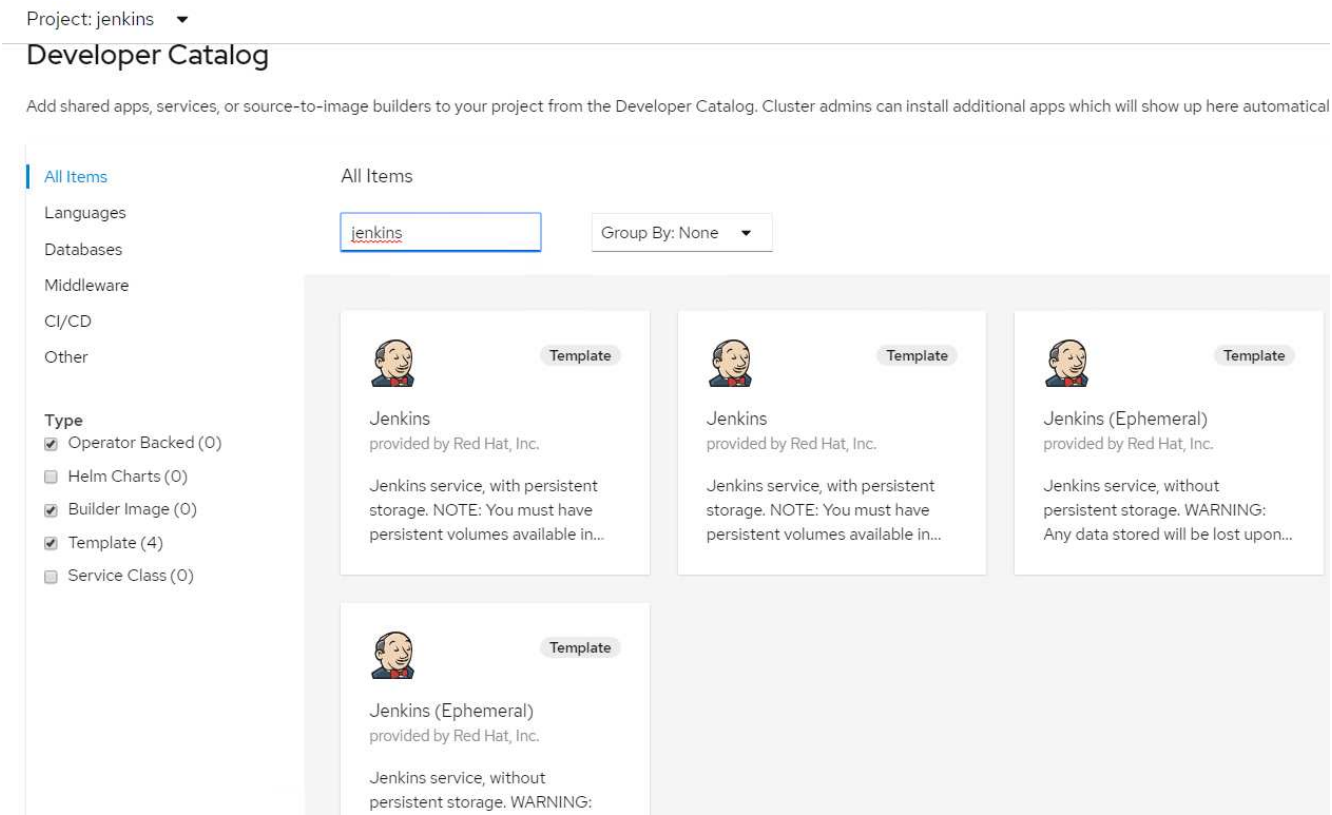
Use label selectors to define how storage is created.

Create Cancel

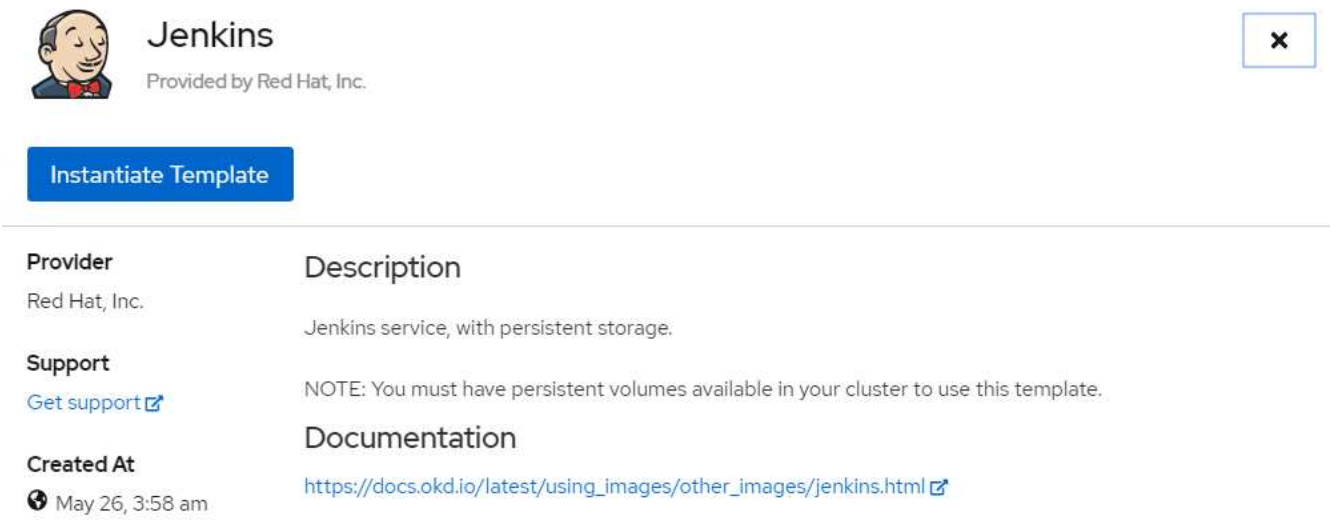
Implementare Jenkins con lo storage persistente

Per implementare Jenkins con lo storage persistente, attenersi alla seguente procedura:

1. Nell'angolo in alto a sinistra, modificare il ruolo da Amministratore a sviluppatore. Fare clic su +Add (Aggiungi) e selezionare From Catalog (dal catalogo) Nella barra Filtra per parola chiave, cercare jenkins. Selezionare Servizio Jenkins con storage persistente.

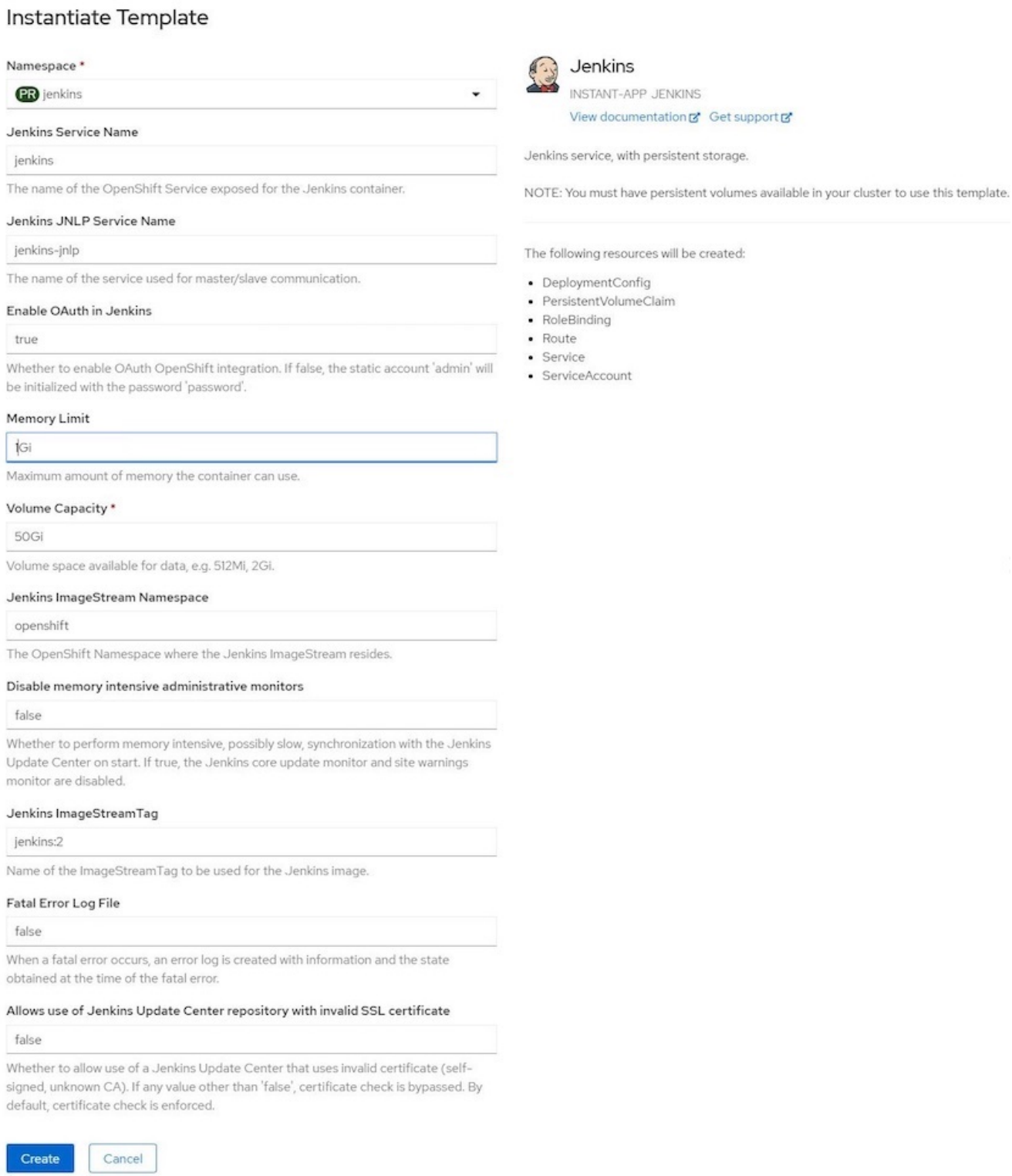


2. Fare clic su `Instantiate Template`.



3. Per impostazione predefinita, i dettagli dell'applicazione Jenkins vengono popolati. In base alle proprie esigenze, modificare i parametri e fare clic su Create (Crea). Questo processo crea tutte le risorse

necessarie per supportare Jenkins su OpenShift.



4. I pod Jenkins impiegano circa 10 - 12 minuti per entrare nello stato Pronta.

Project: jenkins

Pods

Create Pod

Filter by name...

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

Select all filters

1 of 2 Items

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

5. Una volta creata l'istanza dei pod, accedere a Networking > routes (rete > percorsi). Per aprire la pagina Web di Jenkins, fare clic sull'URL fornito per il percorso jenkins.

Project: jenkins

Routes

Create Route

Filter by name...

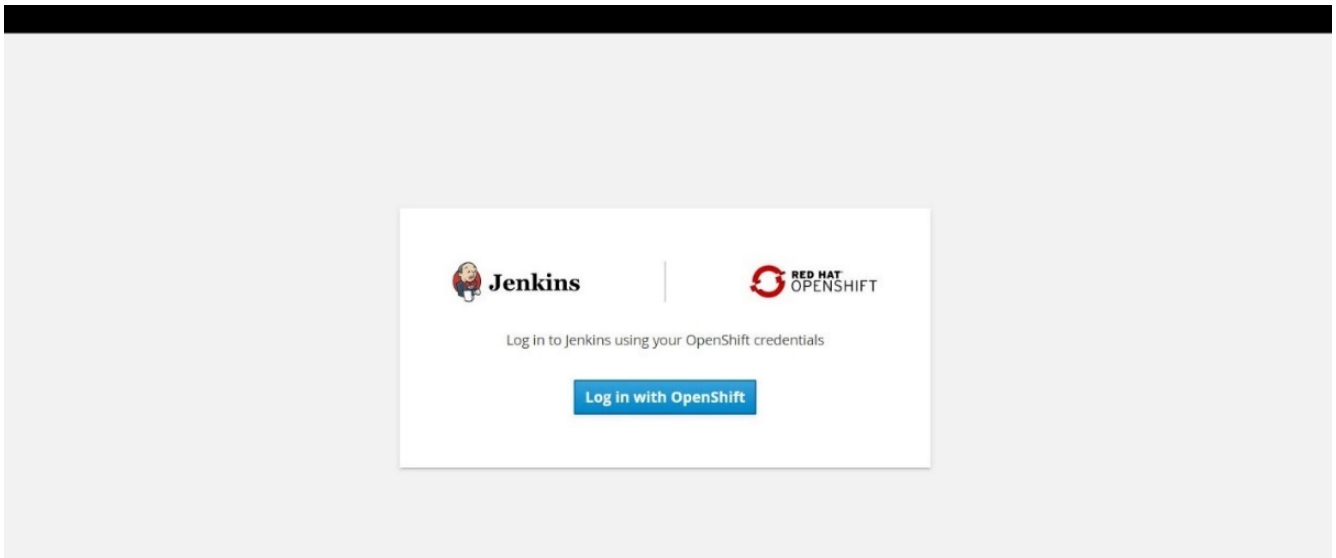
1 Accepted | 0 Rejected | 0 Pending

Select all filters

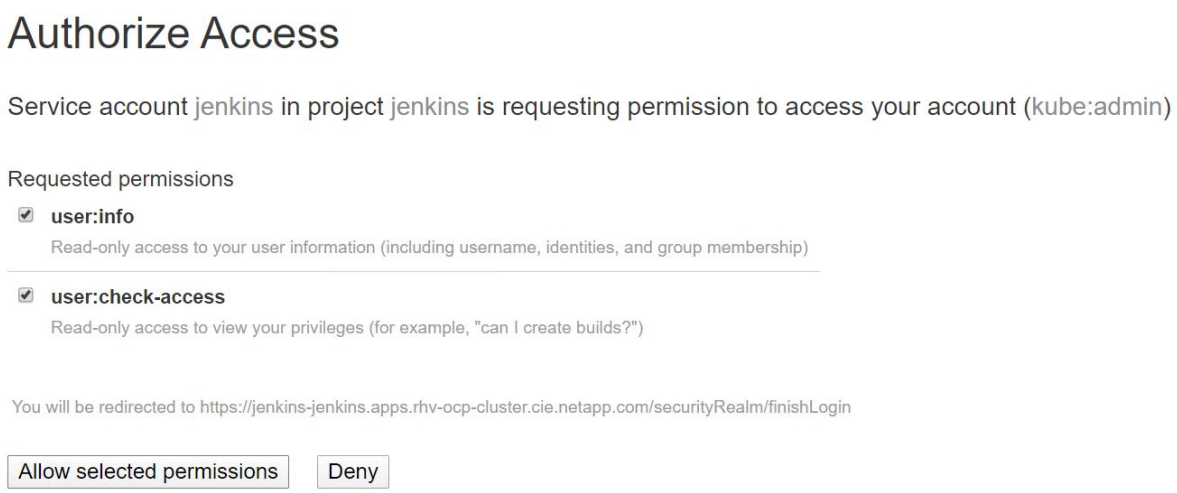
1 Item

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

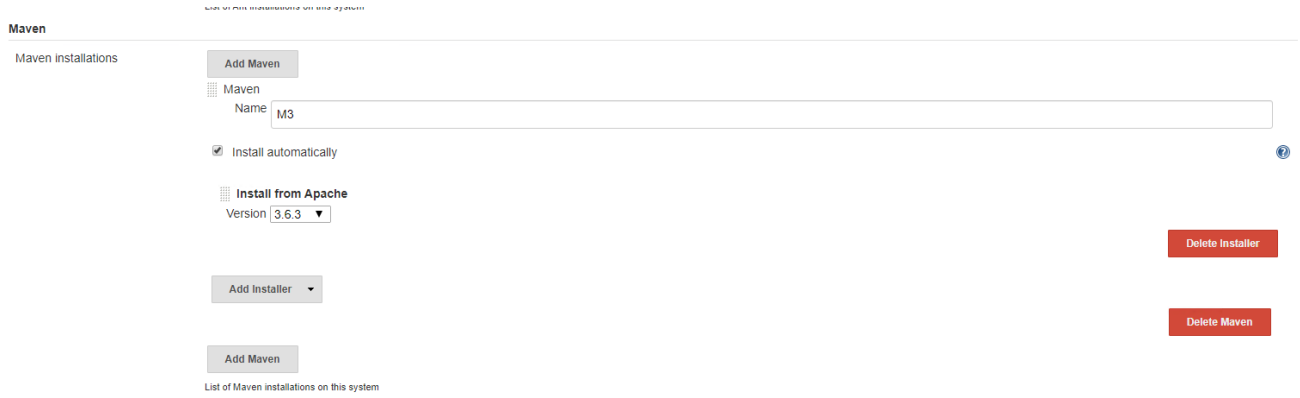
6. Poiché OpenShift OAuth è stato utilizzato durante la creazione dell'applicazione Jenkins, fare clic su Accedi con OpenShift.



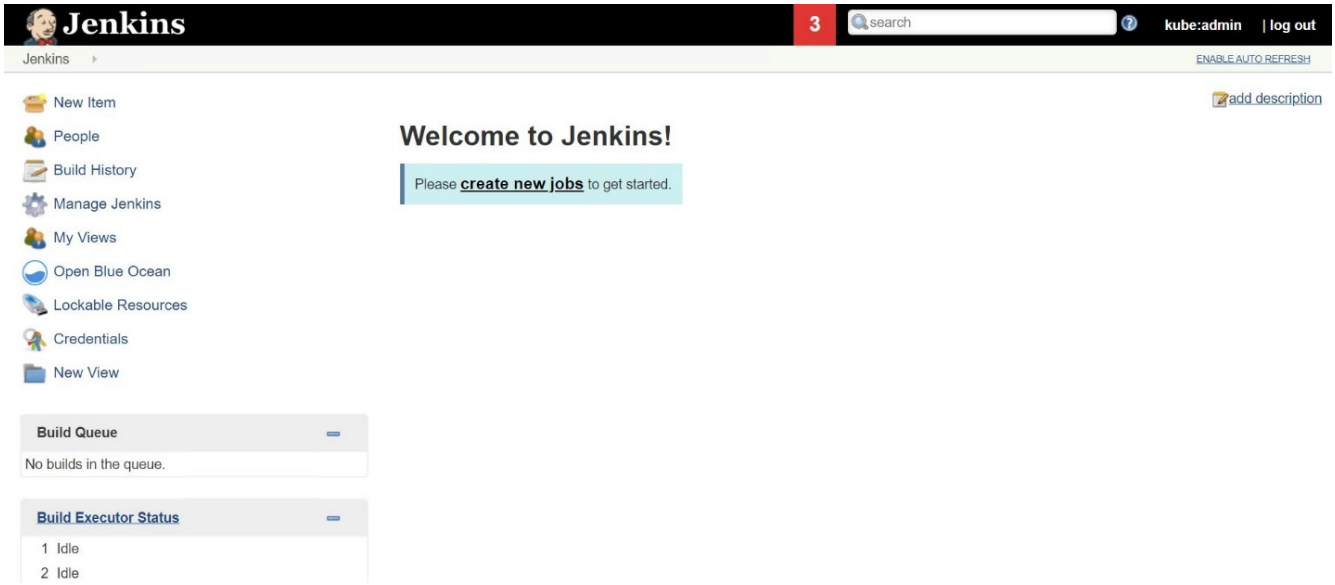
7. Autorizzare l’account del servizio Jenkins ad accedere agli utenti OpenShift.



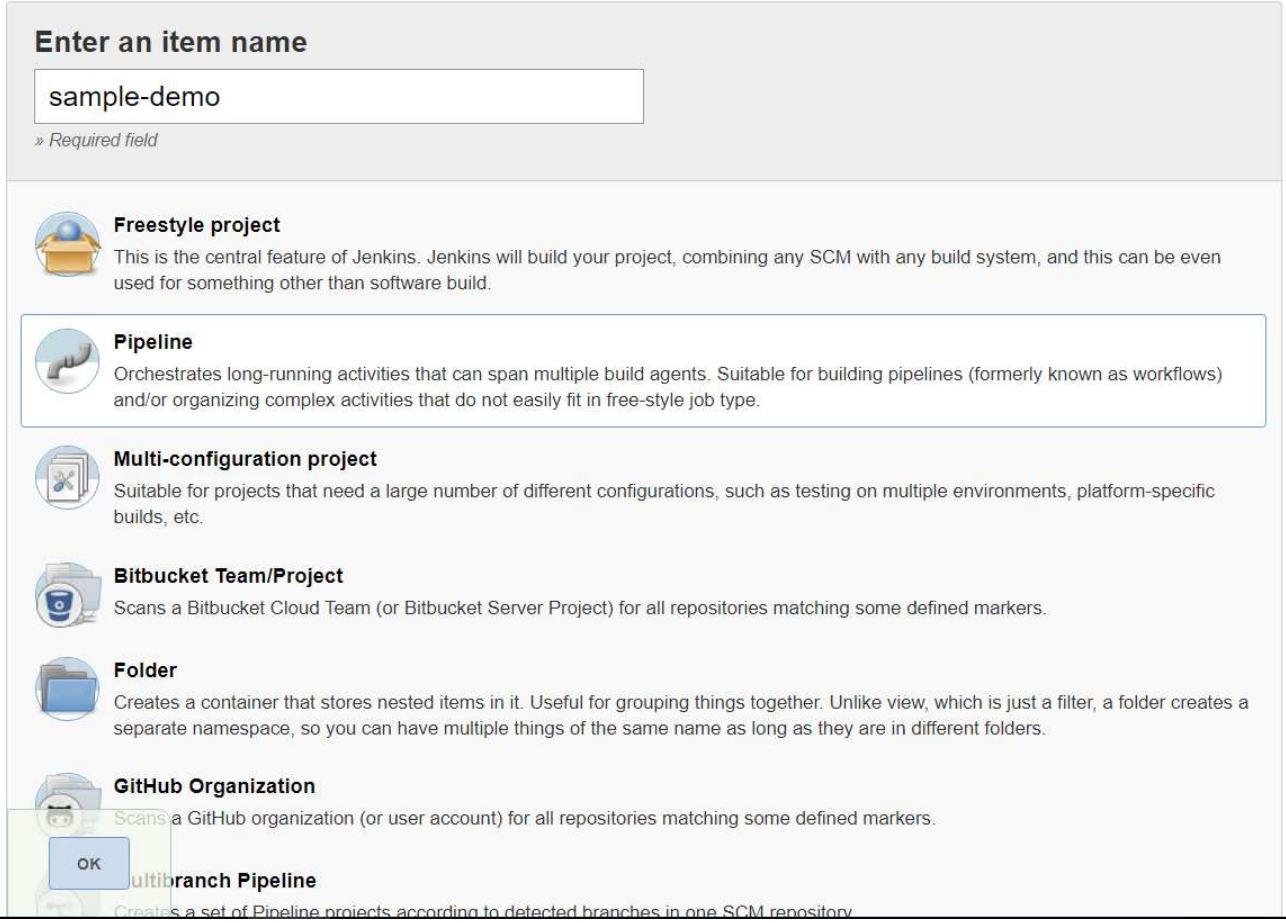
8. Viene visualizzata la pagina di benvenuto di Jenkins. Poiché stiamo utilizzando una build Maven, completare prima l’installazione di Maven. Accedere a Manage Jenkins > Global Tool Configuration (Gestisci Jenkins > Configurazione globale strumenti), quindi fare clic su Add Maven (Aggiungi Maven) nella sottotesta di Maven. Immettere il nome desiderato e assicurarsi che l’opzione Installa automaticamente sia selezionata. Fare clic su Salva.



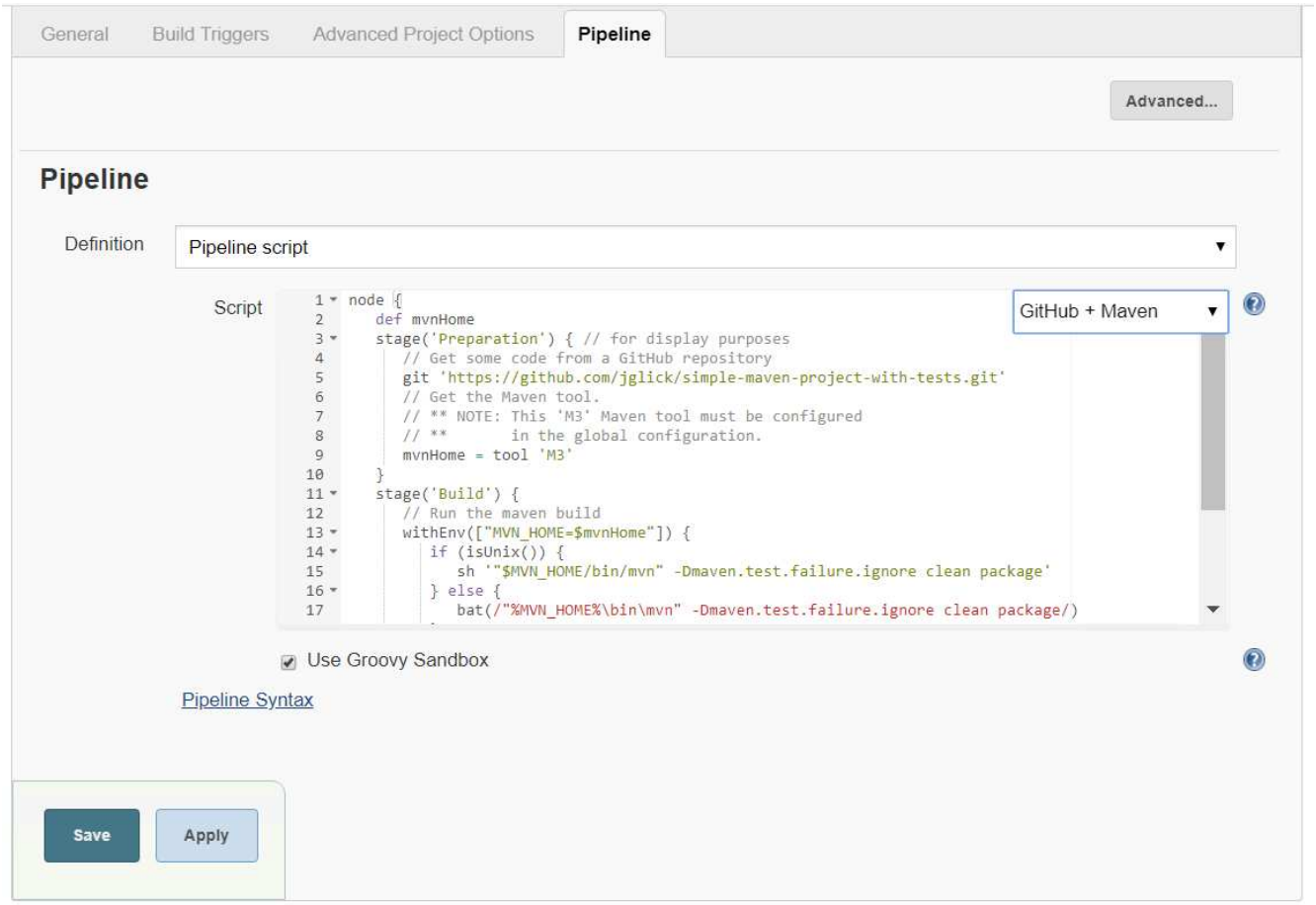
9. È ora possibile creare una pipeline per dimostrare il flusso di lavoro ci/CD. Nella home page, fare clic su Create New Jobs (Crea nuovi lavori) o New Item (nuovo elemento) dal menu a sinistra.



10. Nella pagina Create Item (Crea elemento), immettere il nome desiderato, selezionare Pipeline e fare clic su OK.

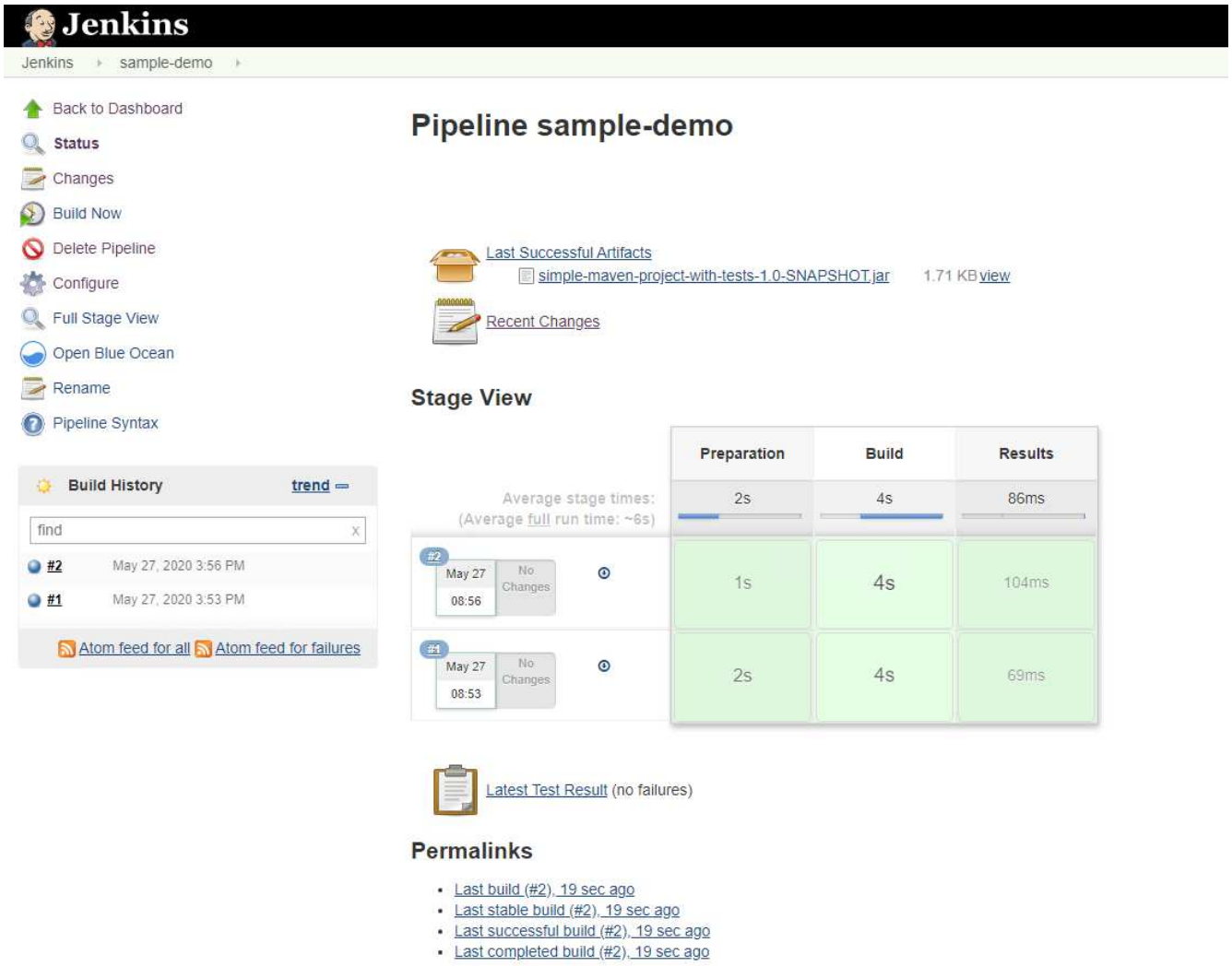


11. Selezionare la scheda pipeline. Dal menu a discesa Try Sample Pipeline, selezionare Github + Maven. Il codice viene compilato automaticamente. Fare clic su Salva.



12. Fare clic su Build Now (Crea ora) per avviare lo sviluppo attraverso la fase di preparazione, creazione e test. Il completamento dell'intero processo di creazione e la visualizzazione dei risultati della creazione possono richiedere alcuni minuti.

13. Ogni volta che si verifica una modifica del codice, la pipeline può essere ricostruita per applicare patch alla nuova versione del software, consentendo un'integrazione continua e un'erogazione continua. Fare clic su Recent Changes (modifiche recenti) per tenere traccia delle modifiche rispetto alla versione precedente.



Configura la multi-tenancy su Red Hat OpenShift con NetApp ONTAP

Configurazione della multi-tenancy su Red Hat OpenShift con NetApp

Molte organizzazioni che eseguono più applicazioni o carichi di lavoro su container tendono a implementare un cluster Red Hat OpenShift per applicazione o carico di lavoro. Ciò consente loro di implementare un rigoroso isolamento per l'applicazione o il carico di lavoro, ottimizzare le performance e ridurre le vulnerabilità della sicurezza. Tuttavia, l'implementazione di un cluster Red Hat OpenShift separato per ciascuna applicazione pone un proprio insieme di problemi. Aumenta l'overhead operativo dovendo monitorare e gestire ciascun cluster da solo, aumenta i costi grazie alle risorse dedicate per le diverse applicazioni e ostacola l'efficienza della scalabilità.

Per risolvere questi problemi, si può prendere in considerazione l'esecuzione di tutte le applicazioni o i carichi di lavoro in un singolo cluster Red Hat OpenShift. Tuttavia, in un'architettura di questo tipo, le vulnerabilità legate all'isolamento delle risorse e alla sicurezza delle applicazioni sono una delle sfide principali. Qualsiasi vulnerabilità di sicurezza in un workload potrebbe naturalmente ricadersi in un altro workload, aumentando così la zona di impatto. Inoltre, qualsiasi utilizzo improvviso e non controllato delle risorse da parte di un'applicazione può influire sulle prestazioni di un'altra applicazione, poiché non esiste un criterio di allocazione delle risorse per impostazione predefinita.

Pertanto, le organizzazioni cercano soluzioni in grado di ottenere il meglio in entrambi i mondi, ad esempio, consentendo loro di eseguire tutti i propri carichi di lavoro in un singolo cluster e offrendo al contempo i

vantaggi di un cluster dedicato per ogni carico di lavoro.

Una di queste soluzioni efficaci consiste nel configurare la multi-tenancy su Red Hat OpenShift. La multi-tenancy è un'architettura che consente a più tenant di coesistere sullo stesso cluster con un corretto isolamento delle risorse, della sicurezza e così via. In questo contesto, un tenant può essere visualizzato come un sottoinsieme delle risorse del cluster configurate per essere utilizzate da un particolare gruppo di utenti a scopo esclusivo. La configurazione della multi-tenancy su un cluster Red Hat OpenShift offre i seguenti vantaggi:

- Riduzione di CapEx e OpEx grazie alla condivisione delle risorse del cluster
- Riduzione dell'overhead operativo e di gestione
- Proteggere i carichi di lavoro dalla contaminazione incrociata delle violazioni della sicurezza
- Protezione dei carichi di lavoro da un peggioramento inatteso delle performance dovuto a conflitti di risorse

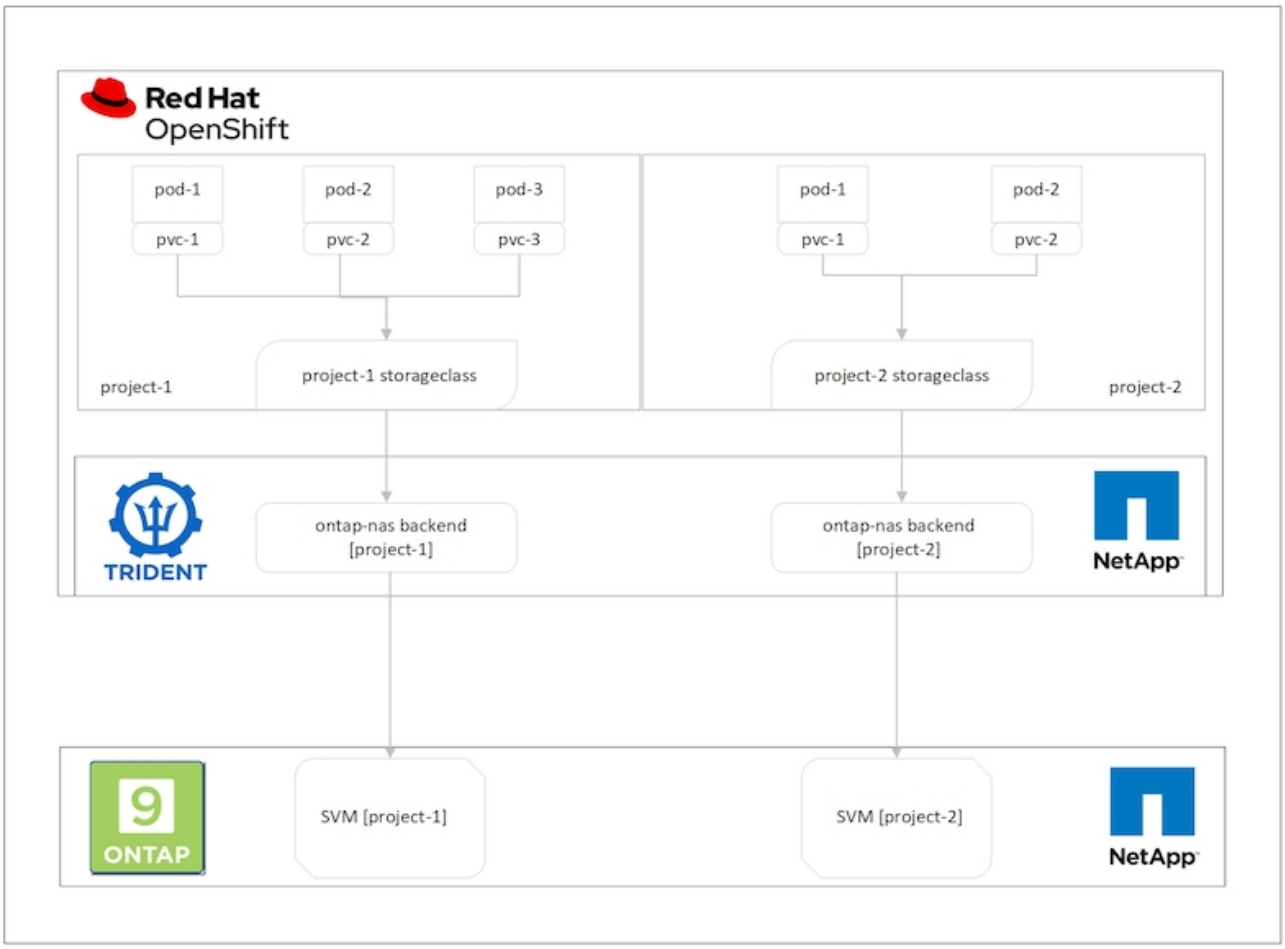
Per un cluster OpenShift multitenant completamente realizzato, è necessario configurare le quote e le restrizioni per le risorse cluster appartenenti a diversi bucket di risorse: Calcolo, storage, networking, sicurezza e così via. Anche se vengono trattati alcuni aspetti di tutti i bucket di risorse di questa soluzione, Ci concentriamo sulle Best practice per isolare e proteggere i dati serviti o consumati da più carichi di lavoro sullo stesso cluster Red Hat OpenShift configurando la multi-tenancy sulle risorse storage allocate dinamicamente da Astra Trident con il supporto di NetApp ONTAP.

Architettura

Sebbene Red Hat OpenShift e Astra Trident supportati da NetApp ONTAP non forniscano l'isolamento tra i carichi di lavoro per impostazione predefinita, offrono un'ampia gamma di funzionalità che possono essere utilizzate per configurare la multi-tenancy. Per comprendere meglio la progettazione di una soluzione multi-tenant su un cluster Red Hat OpenShift con Astra Trident supportato da NetApp ONTAP, prendiamo in considerazione un esempio con una serie di requisiti e descriviamo la configurazione che lo circonda.

Supponiamo che un'organizzazione esegue due dei propri workload su un cluster Red Hat OpenShift nell'ambito di due progetti su cui lavorano due team diversi. I dati per questi carichi di lavoro risiedono su PVC che vengono forniti dinamicamente da Astra Trident su un backend NAS NetApp ONTAP. L'organizzazione deve progettare una soluzione multi-tenant per questi due carichi di lavoro e isolare le risorse utilizzate per questi progetti per garantire il mantenimento della sicurezza e delle performance, concentrandosi principalmente sui dati che servono tali applicazioni.

La seguente figura illustra la soluzione multi-tenant su un cluster Red Hat OpenShift con Astra Trident supportato da NetApp ONTAP.



Requisiti tecnologici

1. Cluster di storage NetApp ONTAP
2. Cluster Red Hat OpenShift
3. Astra Trident

Red Hat OpenShift – risorse cluster

Dal punto di vista del cluster Red Hat OpenShift, la risorsa di primo livello da iniziare è il progetto. Un progetto OpenShift può essere visto come una risorsa di cluster che divide l'intero cluster OpenShift in più cluster virtuali. Pertanto, l'isolamento a livello di progetto fornisce una base per la configurazione della multi-tenancy.

Successivamente, configurare RBAC nel cluster. La Best practice consiste nell'avere tutti gli sviluppatori che lavorano su un singolo progetto o workload configurati in un singolo gruppo di utenti nel provider di identità (IdP). Red Hat OpenShift consente l'integrazione di IdP e la sincronizzazione dei gruppi di utenti, consentendo così l'importazione di utenti e gruppi da IdP nel cluster. Ciò consente agli amministratori del cluster di separare l'accesso delle risorse del cluster dedicate a un progetto a un gruppo di utenti o a gruppi che lavorano su tale progetto, limitando in tal modo l'accesso non autorizzato a qualsiasi risorsa del cluster. Per ulteriori informazioni sull'integrazione di IdP con Red Hat OpenShift, consulta la documentazione "qui".

NetApp ONTAP

È importante isolare lo storage condiviso che funge da provider di storage persistente per un cluster Red Hat OpenShift per assicurarsi che i volumi creati sullo storage per ogni progetto appaiano agli host come se

fossero creati su storage separato. A tale scopo, è possibile creare un numero di SVM (macchine virtuali di storage) su NetApp ONTAP pari al numero di progetti o carichi di lavoro e dedicare ogni SVM a un carico di lavoro.

Astra Trident

Dopo aver creato diverse SVM per diversi progetti su NetApp ONTAP, è necessario mappare ciascuna SVM su un backend Trident diverso. La configurazione di back-end su Trident determina l'allocazione dello storage persistente alle risorse del cluster OpenShift e richiede il mapping dei dettagli della SVM. Questo dovrebbe essere il driver del protocollo per il backend al minimo. Facoltativamente, consente di definire il provisioning dei volumi sullo storage e di impostare limiti per la dimensione dei volumi o l'utilizzo degli aggregati e così via. È possibile trovare i dettagli relativi alla definizione dei backend Trident "qui".

Red Hat OpenShift – risorse di storage

Dopo aver configurato i backend Trident, il passaggio successivo consiste nella configurazione di StorageClasses. Configura quante sono le classi di storage in cui sono presenti i backend, fornendo a ciascuna classe di storage l'accesso per eseguire lo spin up dei volumi su un solo backend. È possibile mappare StorageClass a un particolare backend Trident utilizzando il parametro storagePools durante la definizione della classe di storage. È possibile trovare i dettagli per definire una classe di storage "qui". Pertanto, esiste una mappatura uno a uno da StorageClass a Trident backend che punta a una SVM. In questo modo, tutte le attestazioni di storage tramite la StorageClass assegnata a quel progetto vengono gestite solo dalla SVM dedicata a quel progetto.

Poiché le classi di storage non sono risorse con spazio dei nomi, come possiamo garantire che le attestazioni di storage alla classe di storage di un progetto per pod in un altro namespace o progetto vengano rifiutate? La risposta è utilizzare ResourceQuotas. ResourceQuotas sono oggetti che controllano l'utilizzo totale delle risorse per progetto. Può limitare il numero e la quantità totale di risorse che possono essere utilizzate dagli oggetti nel progetto. Quasi tutte le risorse di un progetto possono essere limitate utilizzando ResourceQuotas e questo può aiutare le organizzazioni a ridurre i costi e le interruzioni dovute all'overprovisioning o all'eccessivo consumo di risorse. Consultare la documentazione "qui" per ulteriori informazioni.

In questo caso di utilizzo, dobbiamo limitare i pod di un progetto specifico al fine di richiedere storage da classi di storage non dedicate al loro progetto. A tale scopo, è necessario limitare le richieste di rimborso persistenti per volumi per altre classi di storage mediante l'impostazione `<storage-class-name>.storageclass.storage.k8s.io/persistentvolumeclaims` a 0. Inoltre, un amministratore del cluster deve garantire che gli sviluppatori di un progetto non abbiano accesso per modificare le ResourceQuotas.

Configurazione

Per qualsiasi soluzione multi-tenant, nessun utente può avere accesso a più risorse di cluster di quelle richieste. Pertanto, l'intero insieme di risorse da configurare come parte della configurazione multi-tenancy è diviso tra cluster-admin, storage-admin e sviluppatori che lavorano su ciascun progetto.

La seguente tabella descrive le diverse attività che devono essere eseguite da diversi utenti:

Ruolo	Attività
Cluster-admin	Crea progetti per applicazioni o carichi di lavoro diversi
	Creare ClusterRoles e RoleBinding per l'amministrazione dello storage
	Creazione di ruoli e associazioni per gli sviluppatori che assegnano l'accesso a progetti specifici
	[Facoltativo] configurare i progetti per pianificare i pod su nodi specifici
Storage-admin	Creare SVM su NetApp ONTAP
	Creare backend Trident
	Creare StorageClasses
	Creare ResourceQuotas di storage
Sviluppatori	Convalidare l'accesso per creare o applicare patch a PVC o pod nel progetto assegnato
	Convalida l'accesso per creare o applicare patch a PVC o pod in un altro progetto
	Convalida l'accesso per visualizzare o modificare progetti, ResourceQuotas e StorageClasses

Configurazione

Prerequisiti

- Cluster NetApp ONTAP
- Cluster Red Hat OpenShift
- Trident installato sul cluster
- Workstation di amministrazione con tool tridentctl e oc installati e aggiunti al percorso dei dollari
- Accesso amministratore a ONTAP
- Accesso cluster-admin al cluster OpenShift
- Il cluster è integrato con il provider di identità
- Il provider di identità è configurato in modo da distinguere in modo efficiente tra gli utenti di diversi team

Configurazione: Attività di amministrazione del cluster

Le seguenti attività vengono eseguite dall'amministratore del cluster Red Hat OpenShift:

1. Accedere al cluster Red Hat OpenShift come amministratore del cluster.
2. Creare due progetti corrispondenti a progetti diversi.

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

3. Creare il ruolo di sviluppatore per il progetto-1.

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

```
      - nodes
  - verbs:
      - '*'
    apiGroups:
      - trident.netapp.io
    resources:
      - tridentsnapshots
EOF
```

La definizione del ruolo fornita in questa sezione è solo un esempio. I ruoli dello sviluppatore devono essere definiti in base ai requisiti dell'utente finale.

1. Allo stesso modo, creare ruoli di sviluppatore per il progetto 2.
2. Tutte le risorse storage di OpenShift e NetApp sono generalmente gestite da un amministratore dello storage. L'accesso per gli amministratori dello storage è controllato dal ruolo di operatore trident creato al momento dell'installazione di Trident. Inoltre, l'amministratore dello storage richiede l'accesso a ResourceQuotas per controllare il modo in cui lo storage viene utilizzato.
3. Creare un ruolo per la gestione di ResourceQuotas in tutti i progetti del cluster per associarlo all'amministratore dello storage.

```
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. Assicurarsi che il cluster sia integrato con il provider di identità dell'organizzazione e che i gruppi di utenti siano sincronizzati con i gruppi di cluster. L'esempio seguente mostra che il provider di identità è stato integrato con il cluster e sincronizzato con i gruppi di utenti.

```
$ 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. Configurare ClusterRoleBinding per gli amministratori dello storage.

```
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
```

Per gli amministratori dello storage, devono essere associati due ruoli: trident-operator e Resource-quote.

1. Creare i RoleBinding per gli sviluppatori che associano il ruolo Developer-project-1 al gruppo corrispondente (ocp-project-1) nel progetto-1.

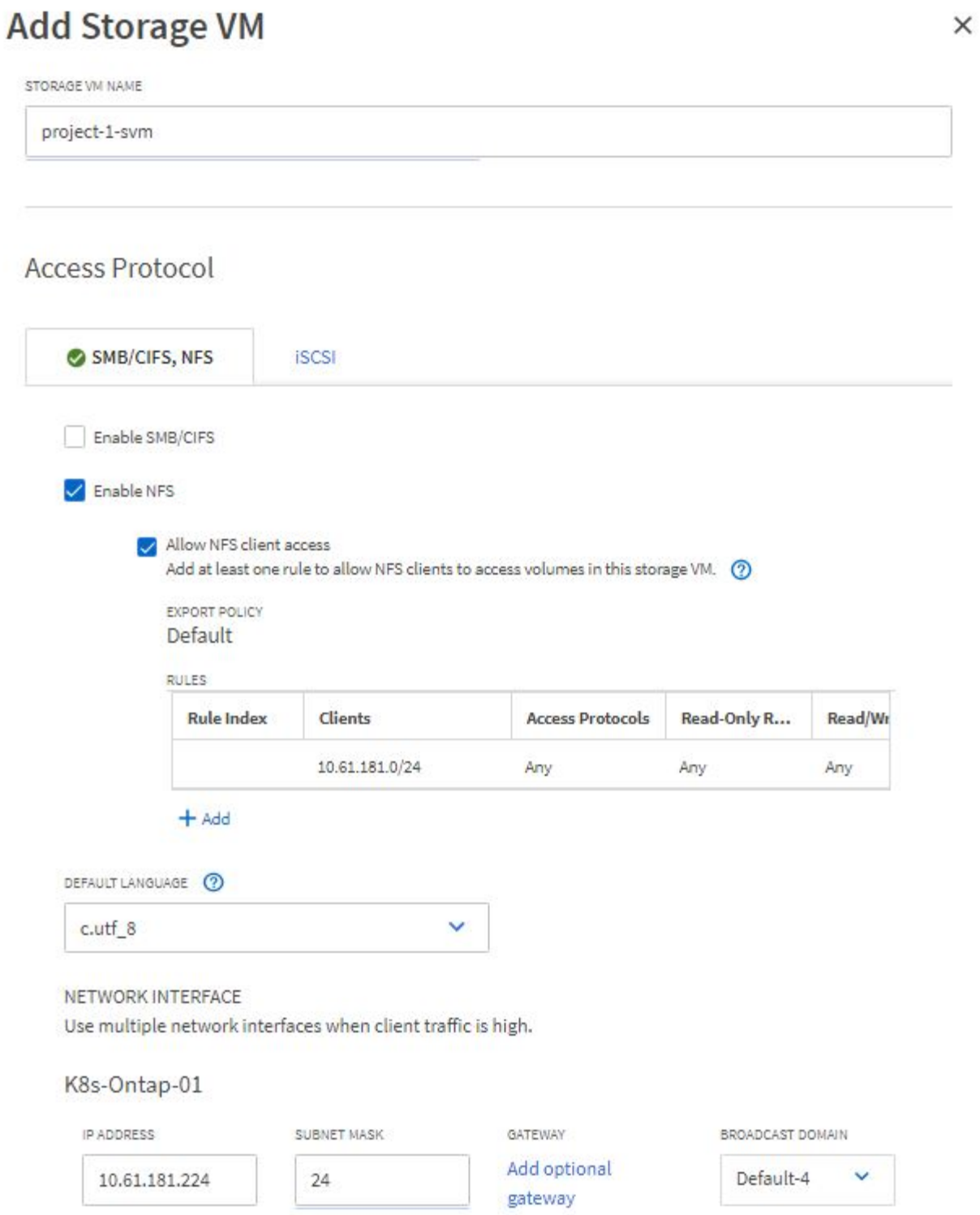
```
cat << EOF | oc create -f -
kind: RoleBinding
apiVersion: rbac.authorization.k8s.io/v1
metadata:
  name: project-1-developer
  namespace: project-1
subjects:
  - kind: Group
    apiGroup: rbac.authorization.k8s.io
    name: ocp-project-1
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: Role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

2. Allo stesso modo, creare RoleBinding per gli sviluppatori che associano i ruoli di sviluppatore al gruppo di utenti corrispondente nel progetto-2.

Configurazione: Attività di amministrazione dello storage

Le seguenti risorse devono essere configurate da un amministratore dello storage:

1. Accedere al cluster NetApp ONTAP come amministratore.
2. Accedere a Storage > Storage VM (Storage > Storage VM) e fare clic su Add (Aggiungi). Creare due SVM, una per il progetto 1 e l'altra per il progetto 2, fornendo i dettagli richiesti. Inoltre, creare un account vsadmin per gestire SVM e le relative risorse.



1. Accedere al cluster Red Hat OpenShift come amministratore dello storage.
2. Creare il backend per il progetto 1 e mapparlo sulla SVM dedicata al progetto. NetApp consiglia di utilizzare l'account vsadmin di SVM per connettere il backend a SVM invece di utilizzare l'amministratore del cluster ONTAP.

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

Per questo esempio, viene utilizzato il driver ontap-nas. Utilizzare il driver appropriato per creare il backend in base al caso d'utilizzo.

Supponiamo che Trident sia installato nel progetto Trident.

1. Analogamente, creare il backend Trident per il progetto 2 e mapparlo sulla SVM dedicata al progetto 2.
2. Quindi, creare le classi di storage. Creare la classe di storage per il project-1 e configurarla per utilizzare i pool di storage dal backend dedicato al project-1 impostando il parametro storagePools.

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

3. Allo stesso modo, creare una classe di storage per il progetto 2 e configurarla per utilizzare i pool di storage dal back-end dedicato al progetto 2.
4. Creare un ResourceQuota per limitare le risorse nel progetto 1, richiedendo storage da storageclasses dedicati ad altri progetti.

```
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. Allo stesso modo, creare un ResourceQuota per limitare le risorse nel progetto 2, richiedendo lo storage da storageclasses dedicati ad altri progetti.

Convalida

Per convalidare l'architettura multi-tenant configurata nei passaggi precedenti, attenersi alla seguente procedura:

Convalidare l'accesso per creare PVC o pod nel progetto assegnato

1. Accedi come ocp-project-1-user, Developer in project-1.
2. Controllare l'accesso per creare un nuovo progetto.

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

3. Creare un PVC nel progetto 1 utilizzando lo storageclass assegnato al progetto 1.

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

4. Controllare il PV associato al PVC.

```
oc get pv
```

5. Convalida che il PV e il suo volume siano creati in una SVM dedicata al progetto 1 su NetApp ONTAP.

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

6. Creare un pod nel progetto 1 e montare il PVC creato nel passaggio precedente.

```
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. Verificare che il pod sia in funzione e che il volume sia stato montato.

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

Convalidare l’accesso per creare PVC o pod in un altro progetto o utilizzare risorse dedicate a un altro progetto

1. Accedi come ocp-project-1-user, Developer in project-1.
2. Creare un PVC nel progetto 1 utilizzando lo storageclass assegnato al progetto 2.

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

3. Creare un PVC nel progetto 2.

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

4. Assicurarsi che i PVC `test-pvc-project-1-sc-2` e. `test-pvc-project-2-sc-1` non sono stati creati.

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

5. Creare un pod nel progetto 2.

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

Convalida l'accesso per visualizzare e modificare progetti, ResourceQuotas e StorageClasses

1. Accedi come ocp-project-1-user, Developer in project-1.
2. Controllare l'accesso per creare nuovi progetti.

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

3. Convalidare l'accesso per visualizzare i progetti.

```
oc get ns
```

4. Verificare se l'utente può visualizzare o modificare ResourceQuotas nel progetto-1.

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

5. Verificare che l'utente abbia accesso per visualizzare gli storageclasses.

```
oc get sc
```

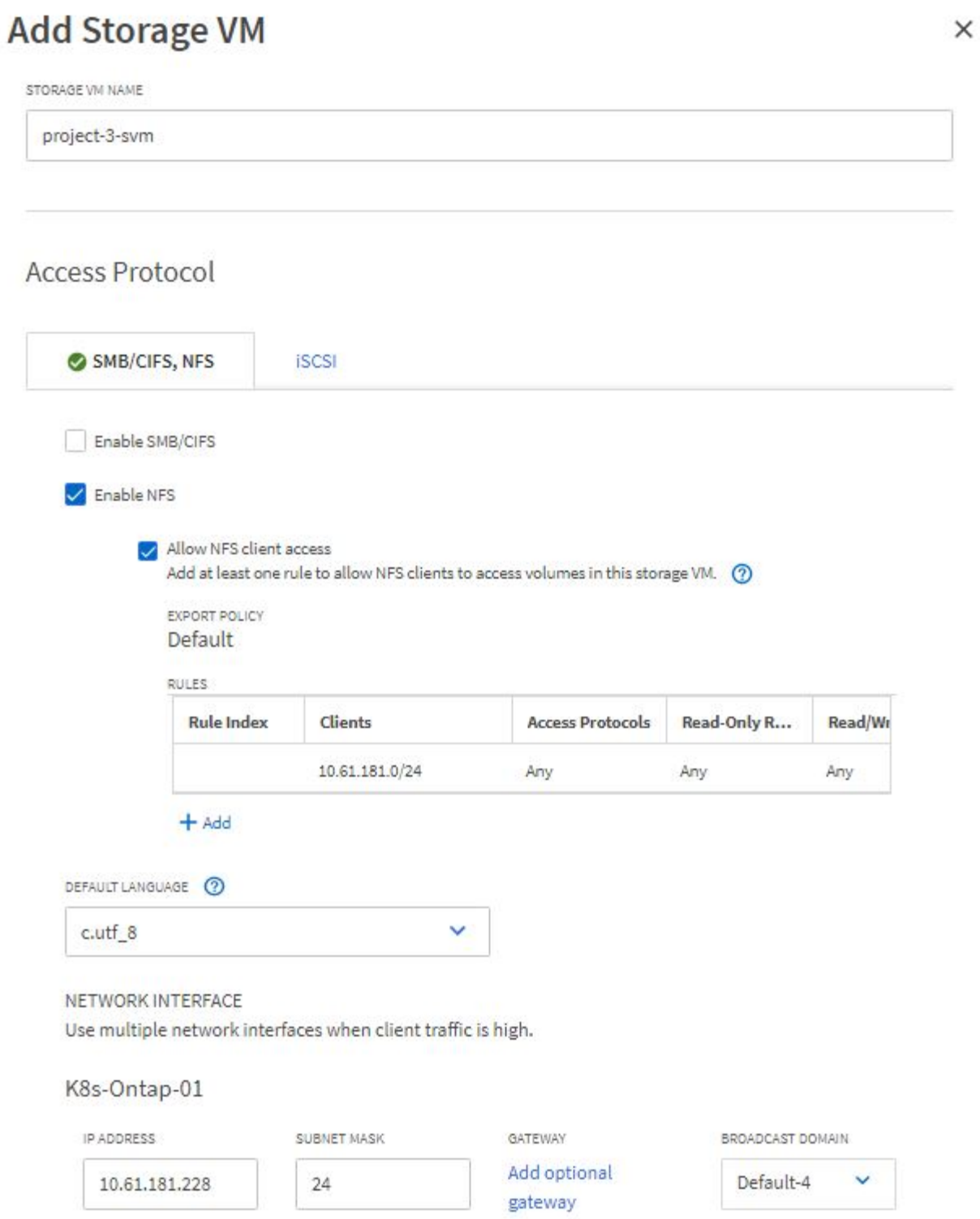
6. Controllare l'accesso per descrivere i magazzini.
7. Convalidare l'accesso dell'utente per modificare gli storageclasses.

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

Scalabilità: Aggiunta di più progetti

In una configurazione multi-tenant, l'aggiunta di nuovi progetti con risorse di storage richiede una configurazione aggiuntiva per garantire che la multi-tenancy non venga violata. Per aggiungere altri progetti in un cluster multi-tenant, attenersi alla seguente procedura:

1. Accedere al cluster NetApp ONTAP come amministratore dello storage.
2. Selezionare `Storage → Storage VMs` e fare clic su `Add`. Creare una nuova SVM dedicata al progetto
3. Inoltre, creare un account vsadmin per gestire SVM e le relative risorse.



1. Accedere al cluster Red Hat OpenShift come amministratore del cluster.
2. Creare un nuovo progetto.

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

3. Assicurarsi che il gruppo di utenti per il project-3 sia creato su IdP e sincronizzato con il cluster OpenShift.

```
oc get groups
```

4. Creare il ruolo di sviluppatore per il progetto 3.

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

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

> La definizione del ruolo fornita in questa sezione è solo un esempio. Il ruolo dello sviluppatore deve essere definito in base ai requisiti dell'utente finale.

1. Creare il RoleBinding per gli sviluppatori nel progetto-3 che legano il ruolo di sviluppatore-progetto-3 al gruppo corrispondente (ocp-progetto-3) nel progetto-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. Accedere al cluster Red Hat OpenShift come amministratore dello storage
3. Creare un backend Trident e mapparlo sulla SVM dedicata al progetto 3. NetApp consiglia di utilizzare l'account vsadmin della SVM per connettere il backend alla SVM invece di utilizzare l'amministratore del cluster ONTAP.

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

Per questo esempio, viene utilizzato il driver ontap-nas. Utilizzare il driver appropriato per creare il backend in base al caso d'utilizzo.

Supponiamo che Trident sia installato nel progetto Trident.

1. Creare la classe di storage per il progetto 3 e configurarla per utilizzare i pool di storage dal back-end dedicato al progetto 3.

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

2. Creare un ResourceQuota per limitare le risorse nel progetto 3, richiedendo storage da storageclasses dedicati ad altri progetti.

```
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. Applicare patch alle ResourceQuotas in altri progetti per limitare l'accesso alle risorse in tali progetti dallo storage dallo storageclass dedicato al progetto-3.

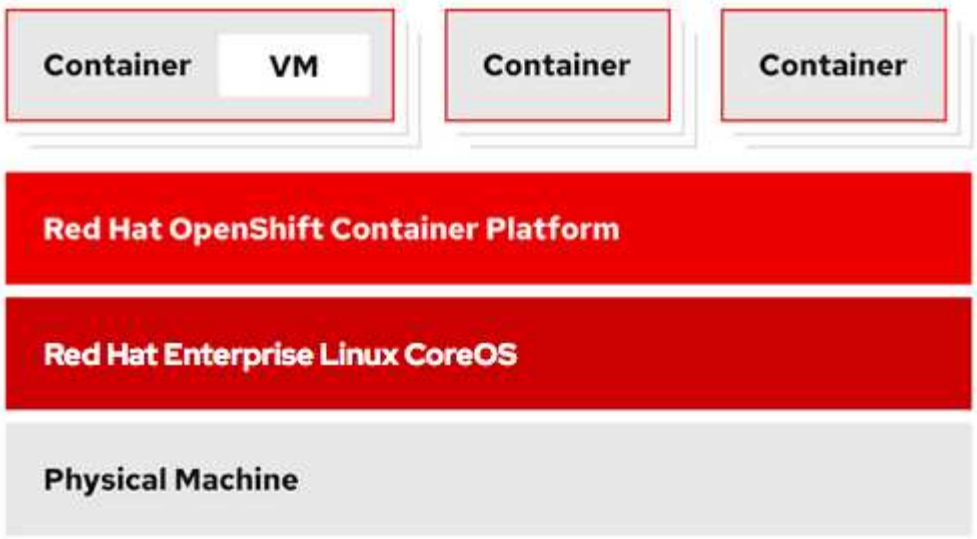
```
oc patch resourcequotas project-1-sc-rq -n project-1 --patch
'{"spec":{"hard":{ "project-3-
sc.storageclass.storage.k8s.io/persistentvolumeclaims": 0}}}'
oc patch resourcequotas project-2-sc-rq -n project-2 --patch
'{"spec":{"hard":{ "project-3-
sc.storageclass.storage.k8s.io/persistentvolumeclaims":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

Virtualizzazione Red Hat OpenShift con NetApp ONTAP

Virtualizzazione Red Hat OpenShift con NetApp ONTAP

A seconda del caso di utilizzo specifico, sia i container che le macchine virtuali (VM) possono fungere da piattaforme ottimali per diversi tipi di applicazioni. Pertanto, molte organizzazioni eseguono alcuni dei propri carichi di lavoro su container e alcune su macchine virtuali. Spesso, questo porta le organizzazioni ad affrontare ulteriori sfide dovendo gestire piattaforme separate: Un hypervisor per le macchine virtuali e un container orchestrator per le applicazioni.

Per affrontare questa sfida, Red Hat ha introdotto la virtualizzazione OpenShift (precedentemente nota come virtualizzazione nativa container) a partire dalla versione 4.6 di OpenShift. La funzionalità di virtualizzazione di OpenShift consente di eseguire e gestire macchine virtuali insieme ai container nella stessa installazione di OpenShift Container Platform, offrendo una funzionalità di gestione ibrida per automatizzare l'implementazione e la gestione delle macchine virtuali attraverso gli operatori. Oltre a creare macchine virtuali in OpenShift, con la virtualizzazione OpenShift, Red Hat supporta anche l'importazione di macchine virtuali da VMware vSphere, Red Hat Virtualization e Red Hat OpenStack Platform.



Alcune funzionalità come la migrazione live delle macchine virtuali, la clonazione dei dischi delle macchine virtuali, le snapshot delle macchine virtuali e così via sono supportate dalla virtualizzazione OpenShift con l'assistenza di Astra Trident, se supportata da NetApp ONTAP. Esempi di ciascuno di questi flussi di lavoro sono discussi più avanti in questo documento nelle rispettive sezioni.

Per ulteriori informazioni sulla virtualizzazione di Red Hat OpenShift, consulta la documentazione "qui".

Implementazione per la virtualizzazione OpenShift

Implementa la virtualizzazione di Red Hat OpenShift con NetApp ONTAP

Prerequisiti

- Un cluster Red Hat OpenShift (successivo alla versione 4.6) installato su un'infrastruttura bare-metal con nodi di lavoro RHCOS
- Il cluster OpenShift deve essere installato tramite l'infrastruttura di provisioning del programma di installazione (IPI)
- Implementare i controlli dello stato delle macchine per mantenere l'ha per le macchine virtuali
- Un cluster NetApp ONTAP
- Astra Trident installato sul cluster OpenShift
- Un backend Trident configurato con una SVM sul cluster ONTAP
- StorageClass configurato sul cluster OpenShift con Astra Trident come provisioner
- Accesso cluster-admin al cluster Red Hat OpenShift
- Accesso amministrativo al cluster NetApp ONTAP
- Una workstation di amministrazione con tridentctl e oc tools installati e aggiunti al percorso dei dollari

Poiché la virtualizzazione OpenShift è gestita da un operatore installato sul cluster OpenShift, impone un overhead aggiuntivo su memoria, CPU e storage, che deve essere tenuto in considerazione durante la pianificazione dei requisiti hardware per il cluster. Consultare la documentazione "qui" per ulteriori dettagli.

In alternativa, è possibile specificare un sottoinsieme dei nodi del cluster OpenShift per ospitare gli operatori, i controller e le macchine virtuali della virtualizzazione OpenShift configurando le regole di posizionamento dei nodi. Per configurare le regole di posizionamento dei nodi per la virtualizzazione OpenShift, seguire la

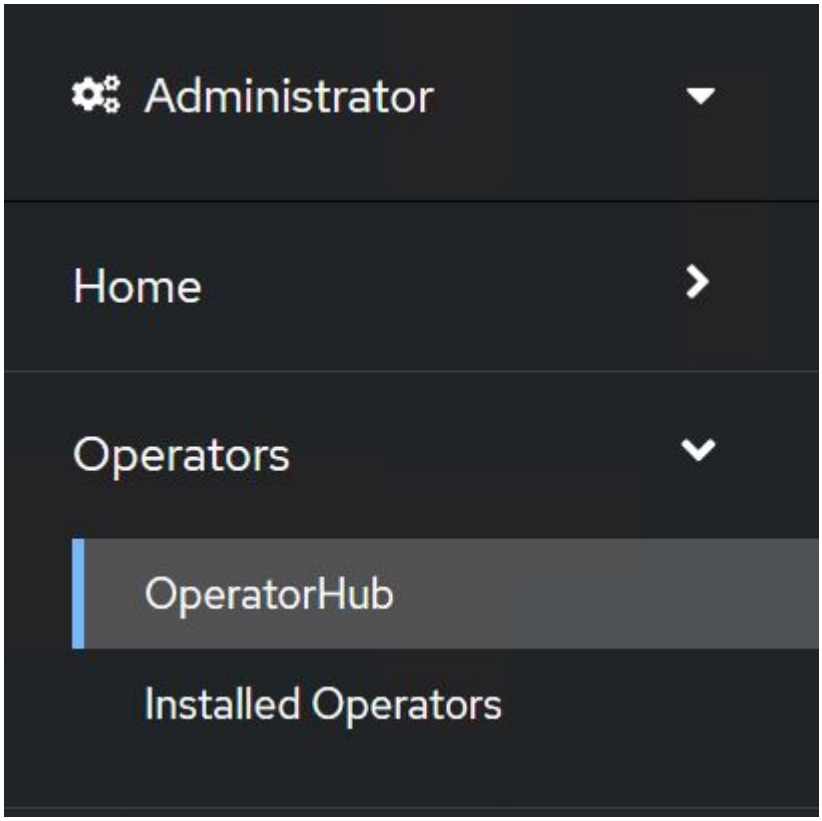
documentazione "qui".

Per il supporto dello storage OpenShift Virtualization, NetApp consiglia di disporre di un StorageClass dedicato che richieda storage da un particolare backend Trident, a sua volta supportato da una SVM dedicata. In questo modo si mantiene un livello di multi-tenancy in relazione ai dati serviti per i carichi di lavoro basati su macchine virtuali sul cluster OpenShift.

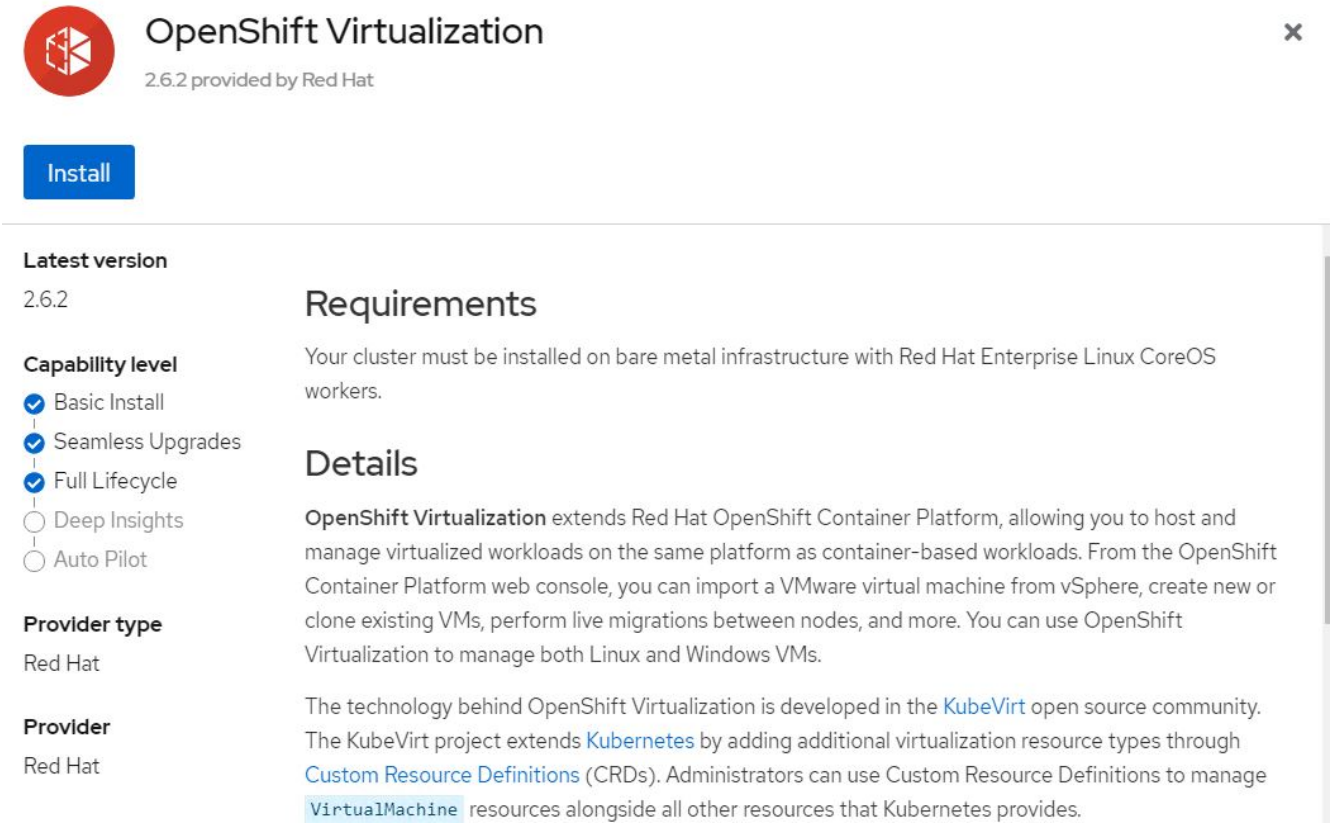
Implementa la virtualizzazione di Red Hat OpenShift con NetApp ONTAP

Per installare OpenShift Virtualization, attenersi alla seguente procedura:

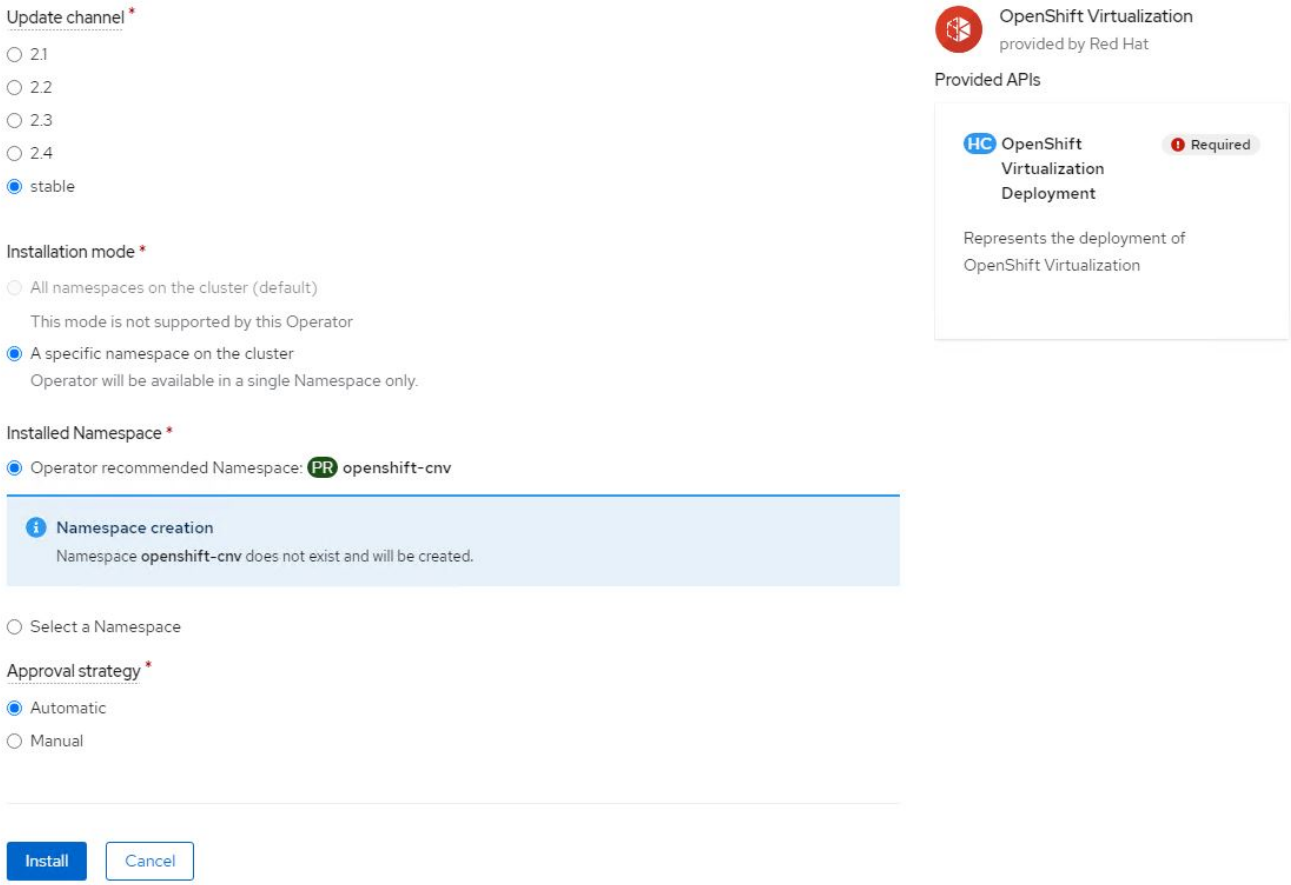
1. Accedi al cluster bare-metal Red Hat OpenShift con accesso cluster-admin.
2. Selezionare Administrator (Amministratore) dal menu a discesa Perspective (prospettiva).
3. Accedere a Operator > OperatorHub e cercare OpenShift Virtualization.



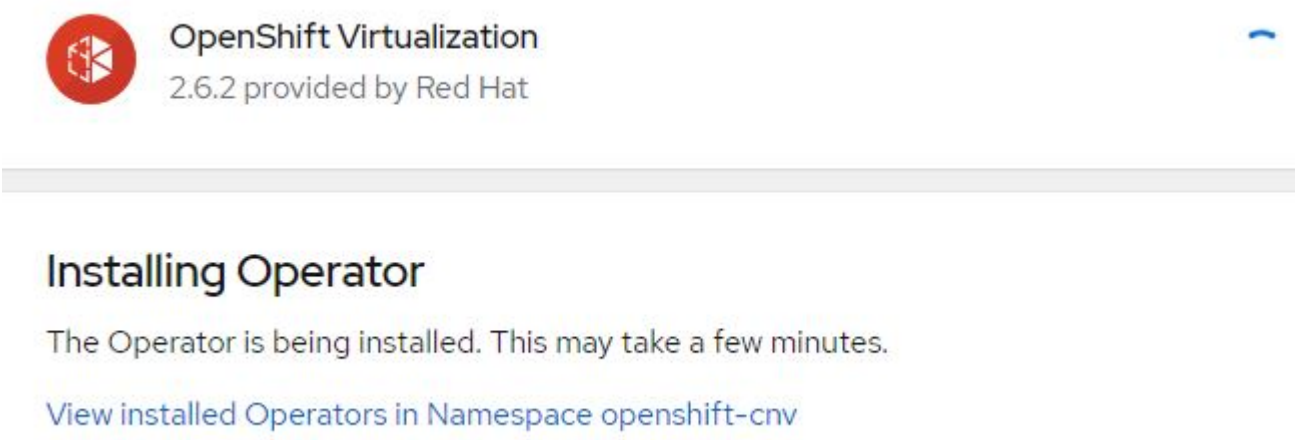
4. Selezionare il riquadro OpenShift Virtualization (virtualizzazione OpenShift) e fare clic su Install (Installa)



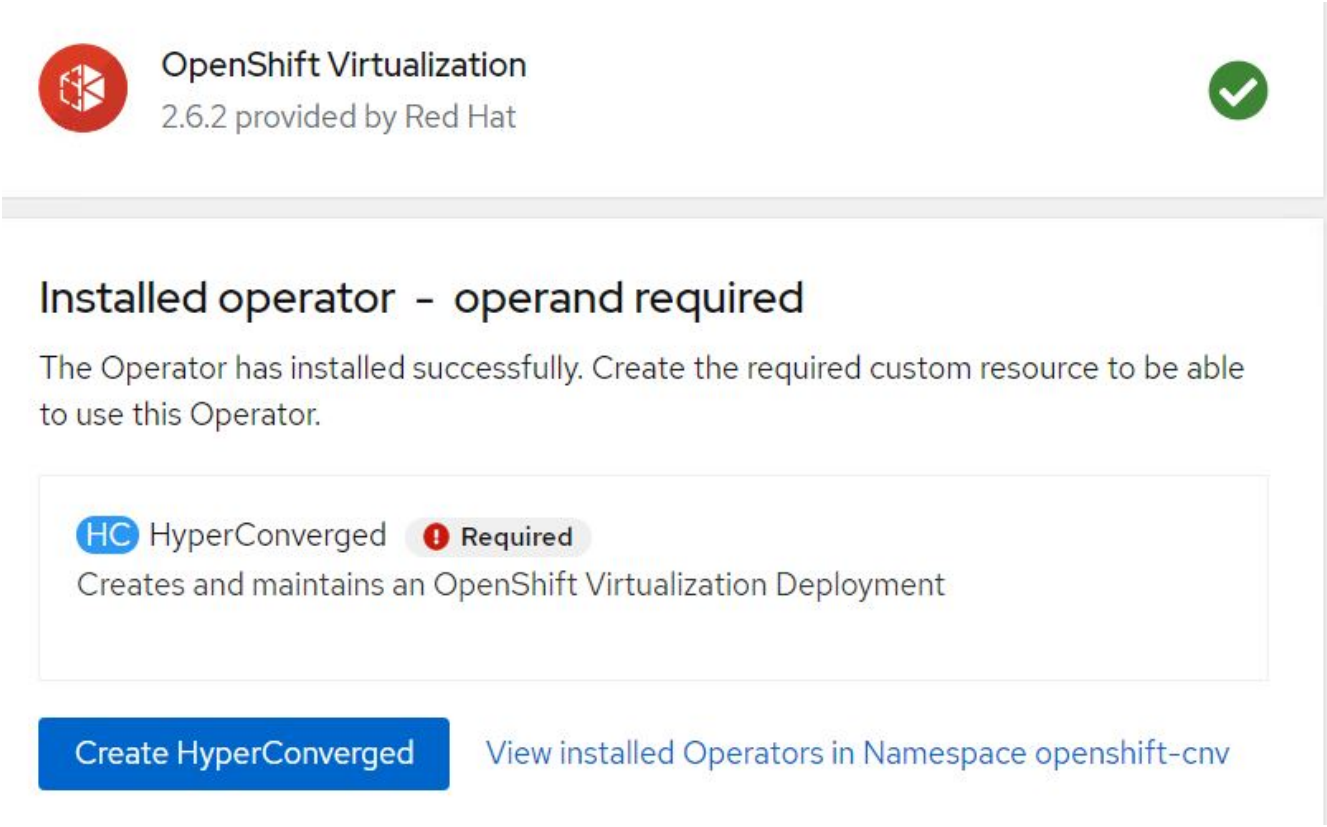
5. Nella schermata Install Operator (Installa operatore), lasciare tutti i parametri predefiniti e fare clic su Install (Installa).



6. Attendere il completamento dell'installazione da parte dell'operatore.



7. Una volta installato l'operatore, fare clic su Create HyperConverged (Crea HyperConverged).



8. Nella schermata Create HyperConverged (Crea HyperConverged), fare clic su Create (Crea), accettando tutti i parametri predefiniti. Questa fase avvia l'installazione di OpenShift Virtualization.

Name *

kubevirt-hyperconverged

Labels

app=frontend

Infra

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

Workloads

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

Bare Metal Platform

true

BareMetalPlatform indicates whether the infrastructure is baremetal.

Feature Gates

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

Local Storage Class Name

LocalStorageClassName the name of the local storage class.

Create Cancel

9. Dopo che tutti i pod sono stati spostati nello stato di esecuzione nello spazio dei nomi openshift-cnv e l'operatore di virtualizzazione OpenShift è in stato di successo, l'operatore è pronto per l'uso. È ora possibile creare macchine virtuali sul cluster OpenShift.

Project: openshift-cnv

Installed Operators

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

Name Search by name...

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

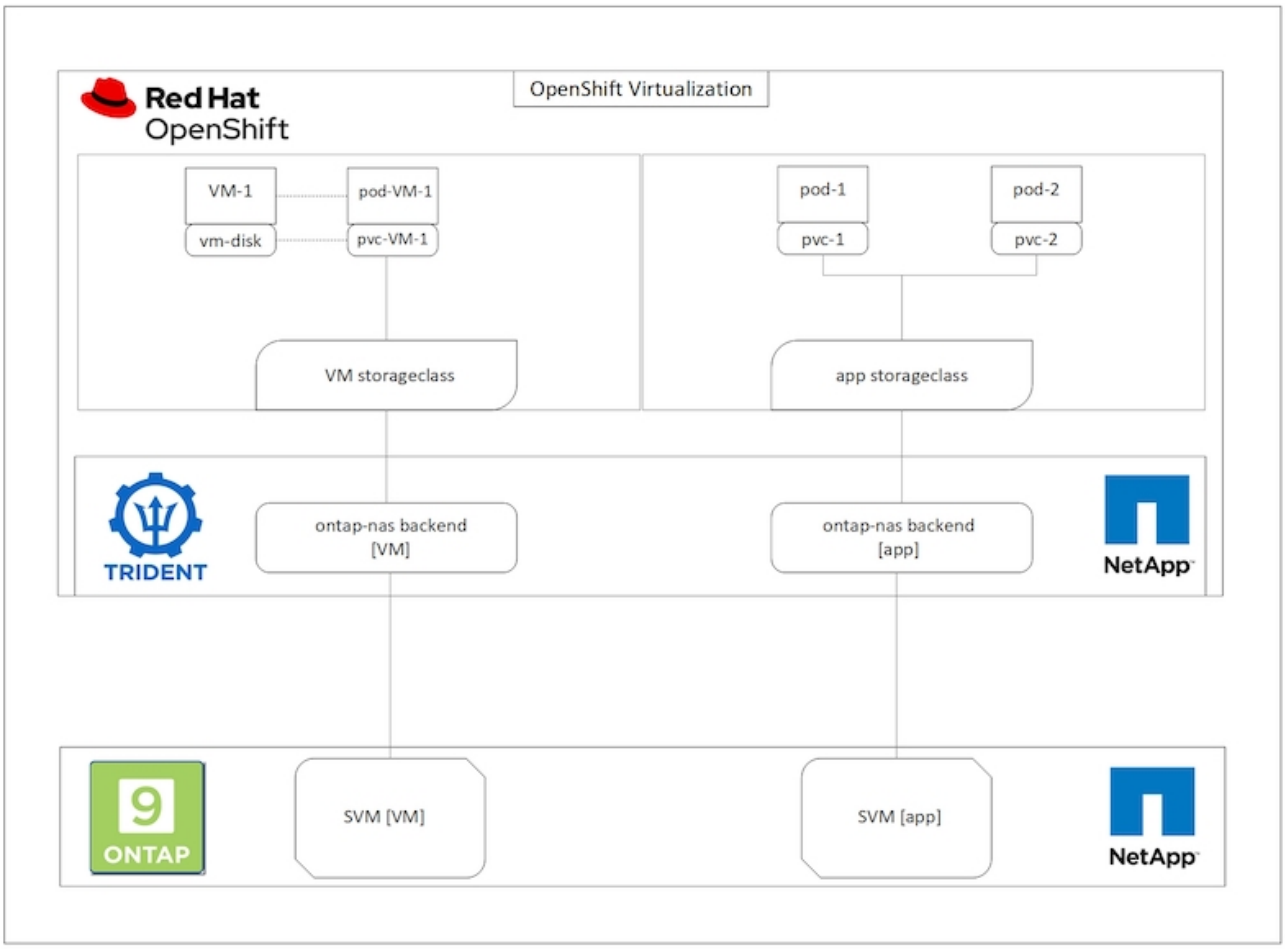
Flussi di lavoro

Workflow: Virtualizzazione Red Hat OpenShift con NetApp ONTAP

Creare una macchina virtuale

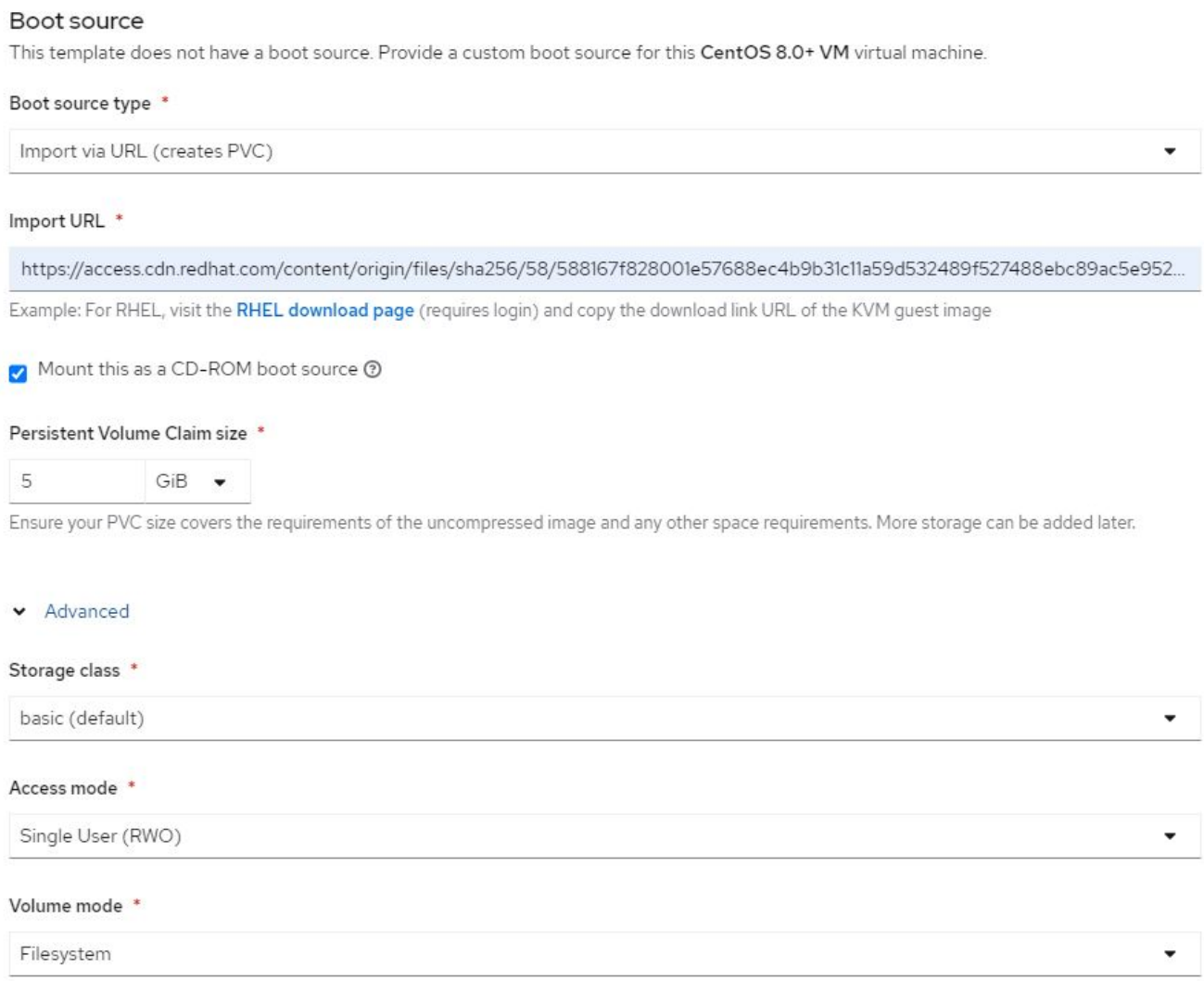
Le VM sono implementazioni stateful che richiedono volumi per ospitare il sistema operativo e i dati. Con CNV, poiché le macchine virtuali vengono eseguite come pod, le macchine virtuali vengono supportate da PVS

ospitati su NetApp ONTAP tramite Trident. Questi volumi sono collegati come dischi e memorizzano l'intero file system, inclusa l'origine di boot della macchina virtuale.

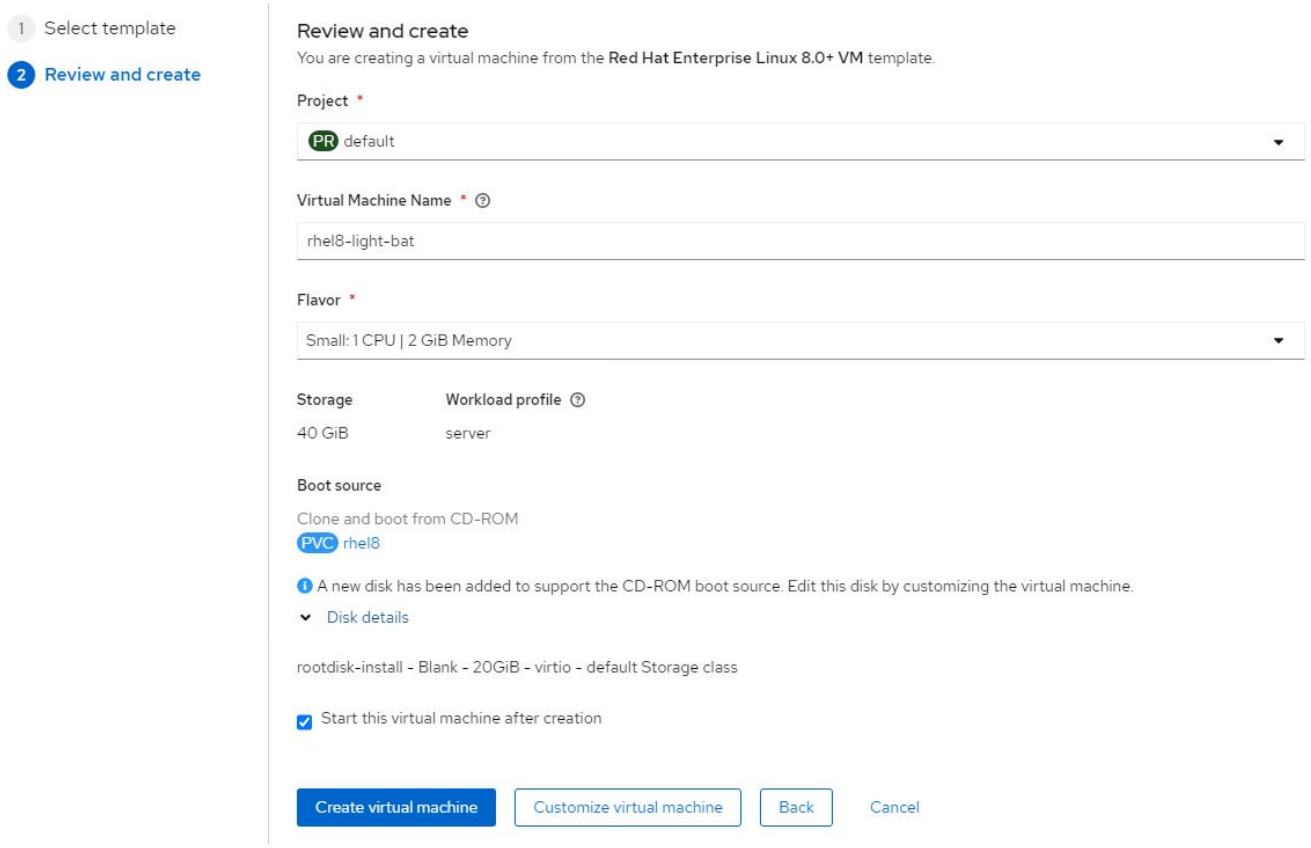


Per creare una macchina virtuale sul cluster OpenShift, attenersi alla seguente procedura:

1. Accedere a workload > virtualizzazione > macchine virtuali e fare clic su Crea > con Wizard.
2. Selezionare il sistema operativo desiderato e fare clic su Next (Avanti).
3. Se nel sistema operativo selezionato non è configurata alcuna origine di avvio, è necessario configurarla. Per Boot Source (origine di avvio), selezionare se si desidera importare l'immagine del sistema operativo da un URL o da un registro e fornire i dettagli corrispondenti. Espandere Advanced (Avanzate) e selezionare Trident-Backed StorageClass (StorageClass supportato da Trident). Quindi fare clic su Next (Avanti).



4. Se il sistema operativo selezionato ha già una sorgente di avvio configurata, il passaggio precedente può essere ignorato.
5. Nel riquadro Review and Create (Revisione e creazione), selezionare il progetto in cui si desidera creare la macchina virtuale e fornire i dettagli della macchina virtuale. Assicurarsi che l'origine di boot sia selezionata come Clone (Clona) e boot from CD-ROM (Avvio da CD-ROM) con il PVC appropriato assegnato per il sistema operativo selezionato.



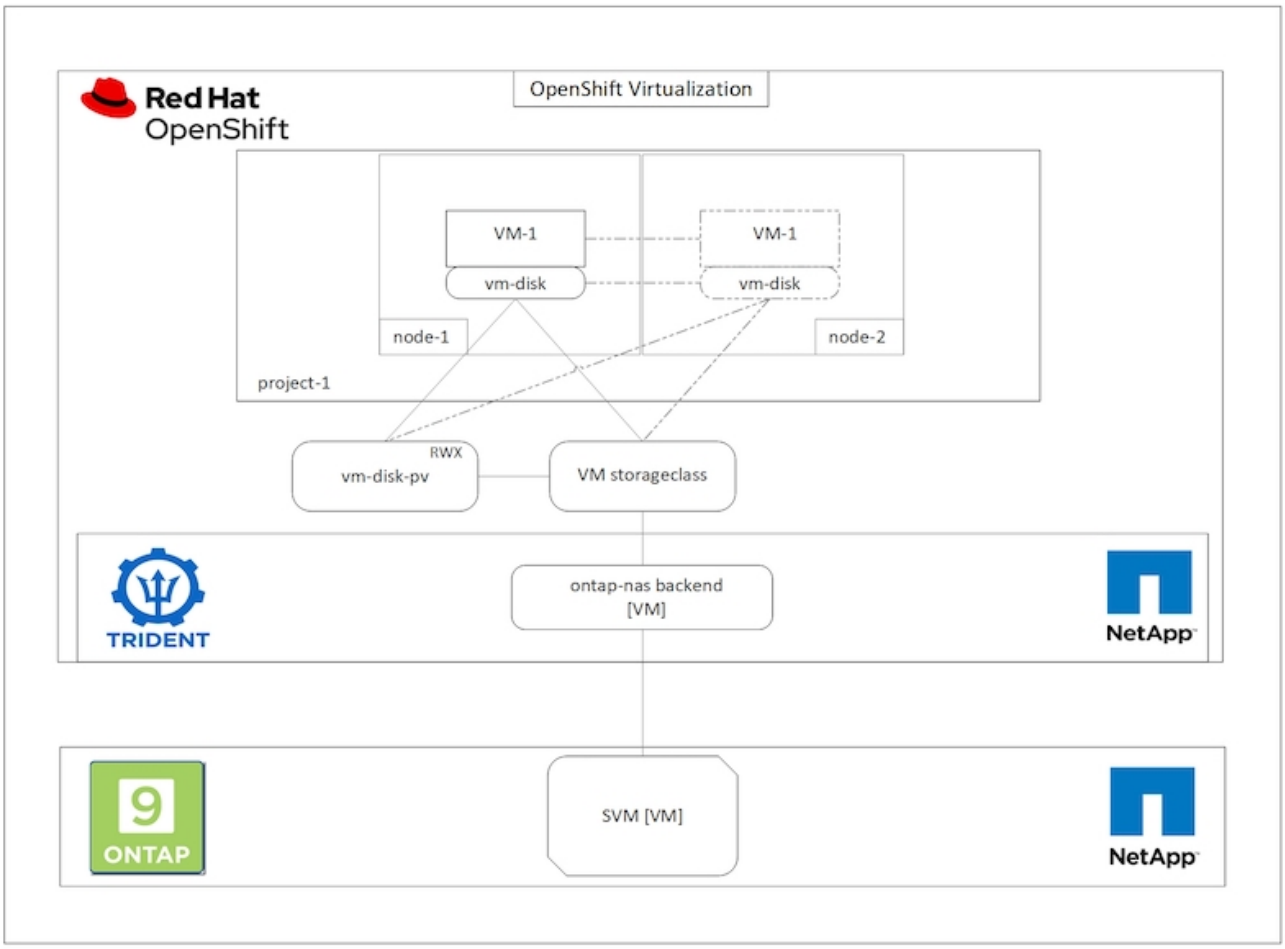
6. Se si desidera personalizzare la macchina virtuale, fare clic su Customize Virtual Machine (Personalizza macchina virtuale) e modificare i parametri richiesti.
7. Fare clic su Create Virtual Machine (Crea macchina virtuale) per creare la macchina virtuale; in questo modo viene fatto rotare in background un pod corrispondente.

Quando un'origine di avvio viene configurata per un modello o un sistema operativo da un URL o da un registro, crea un PVC in `openshift-virtualization-os-images` Proiettare e scaricare l'immagine guest KVM sul PVC. È necessario assicurarsi che i PVC modello dispongano di spazio di provisioning sufficiente per ospitare l'immagine guest KVM per il sistema operativo corrispondente. Questi PVC vengono quindi clonati e collegati come rootdisk alle macchine virtuali quando vengono creati utilizzando i rispettivi modelli in qualsiasi progetto.

Workflow: Virtualizzazione Red Hat OpenShift con NetApp ONTAP

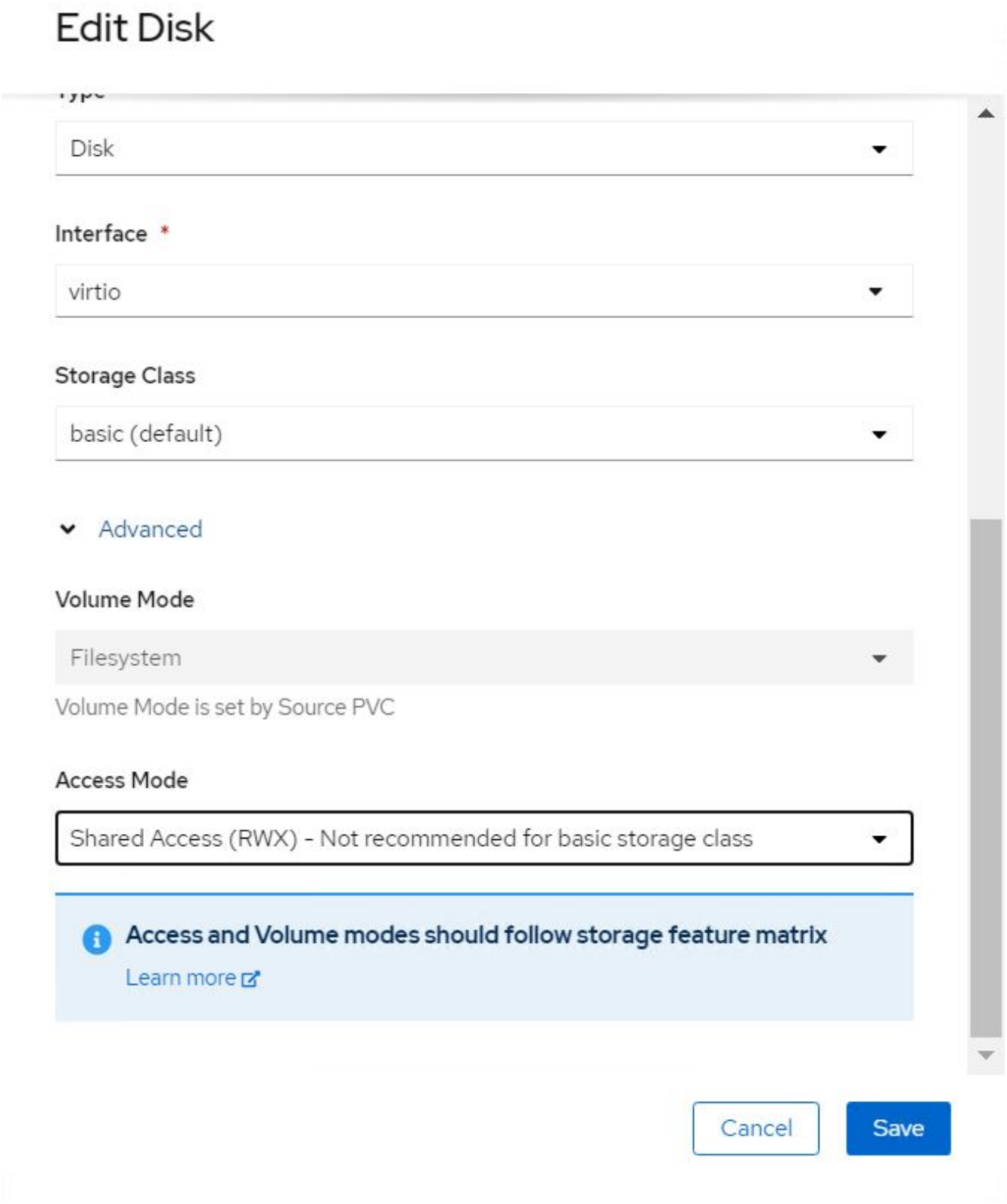
Migrazione VM Live

Live Migration è un processo di migrazione di un'istanza di macchina virtuale da un nodo all'altro in un cluster OpenShift senza downtime. Affinché la migrazione live funzioni in un cluster OpenShift, le macchine virtuali devono essere associate a PVC con modalità di accesso condivisa ReadWriteMany. Il backend Astra Trident configurato con una SVM su un cluster NetApp ONTAP abilitato per il protocollo NFS supporta l'accesso ReadWriteMany condiviso per i PVC. Pertanto, le macchine virtuali con PVC richieste da StorageClasses fornite da Trident da SVM abilitato NFS possono essere migrate senza downtime.



Per creare una macchina virtuale associata a PVC con accesso condiviso ReadWriteMany:

1. Accedere a workload > virtualizzazione > macchine virtuali e fare clic su Crea > con Wizard.
2. Selezionare il sistema operativo desiderato e fare clic su Next (Avanti). Supponiamo che il sistema operativo selezionato abbia già configurato una fonte di avvio.
3. Nel riquadro Review and Create (Revisione e creazione), selezionare il progetto in cui si desidera creare la macchina virtuale e fornire i dettagli della macchina virtuale. Assicurarsi che l'origine di boot sia selezionata come Clone (Clona) e boot from CD-ROM (Avvio da CD-ROM) con il PVC appropriato assegnato per il sistema operativo selezionato.
4. Fare clic su Customize Virtual Machine (Personalizza macchina virtuale), quindi su Storage (Storage).
5. Fare clic sui puntini di sospensione accanto a rootdisk e assicurarsi che sia selezionato lo storageclass con provisioning mediante Trident. Espandere Advanced (Avanzate) e selezionare Shared Access (RWX) (accesso condiviso) per Access Mode (modalità di accesso). Quindi fare clic su Save (Salva).



6. Fare clic su Revisiona e conferma, quindi su Crea macchina virtuale.

Per migrare manualmente una macchina virtuale in un altro nodo del cluster OpenShift, attenersi alla seguente procedura.

1. Accedere a workload > virtualizzazione > macchine virtuali.

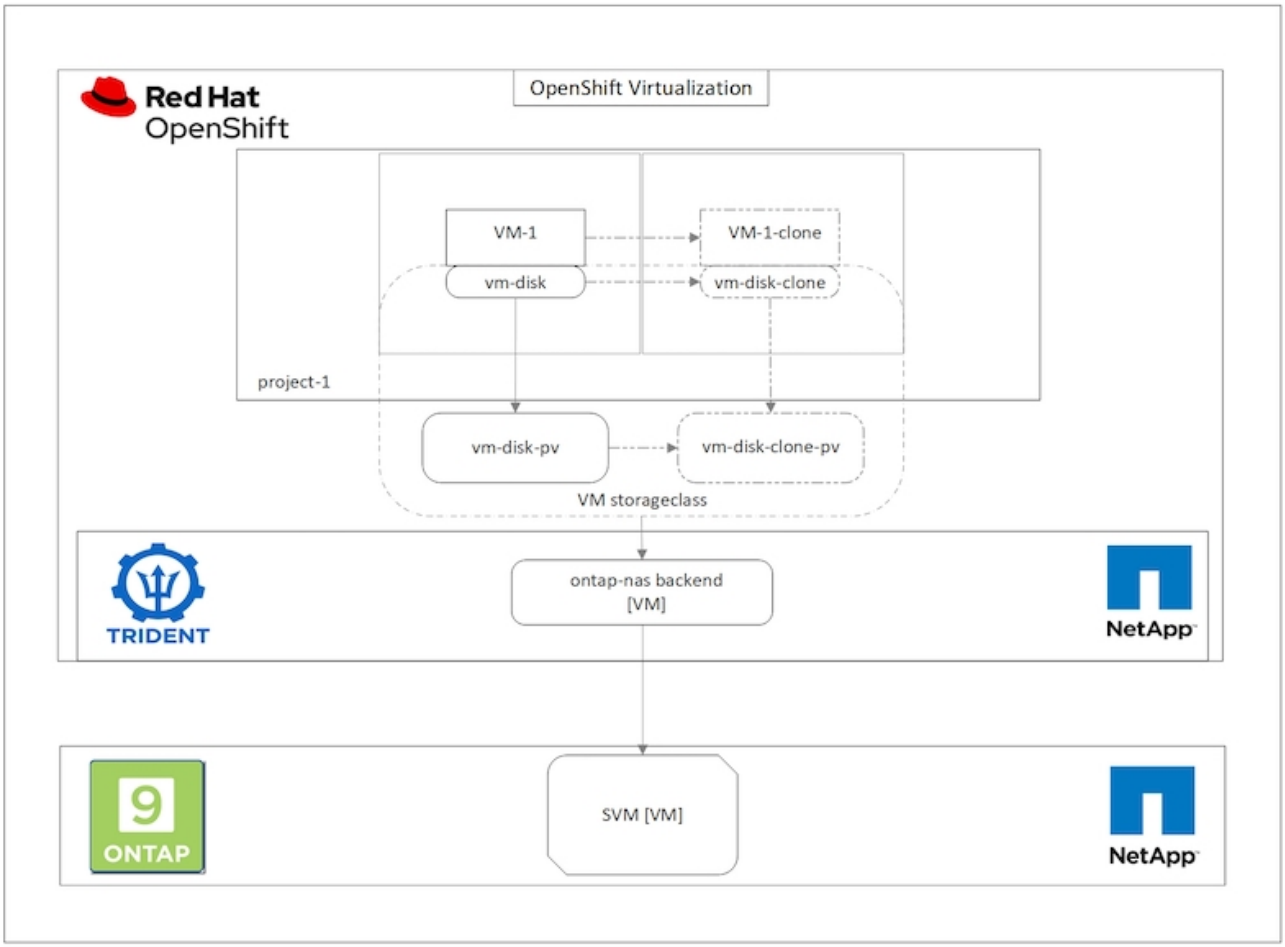
2. Per la macchina virtuale che si desidera migrare, fare clic sui puntini di sospensione, quindi fare clic su Migrate the Virtual Machine (Migra macchina virtuale).
3. Fare clic su Migrate (Migra) quando viene visualizzato il messaggio per confermare.

> Un'istanza della macchina virtuale in un cluster OpenShift esegue automaticamente la migrazione a un altro nodo quando il nodo originale viene messo in modalità di manutenzione se evictionStrategy è impostato su LiveMigrate.

Workflow: Virtualizzazione Red Hat OpenShift con NetApp ONTAP

Cloning delle macchine virtuali

Il cloning di una macchina virtuale esistente in OpenShift viene ottenuto con il supporto della funzionalità di cloning di Volume CSI di Astra Trident. Il cloning del volume CSI consente la creazione di un nuovo PVC utilizzando un PVC esistente come origine dati duplicando il suo PV. Dopo la creazione del nuovo PVC, funziona come entità separata e senza alcun collegamento o dipendenza dal PVC di origine.



La clonazione dei volumi CSI è soggetta a determinate restrizioni:

1. Il PVC di origine e il PVC di destinazione devono trovarsi nello stesso progetto.
2. La clonazione è supportata all'interno della stessa classe di storage.
3. La clonazione può essere eseguita solo quando i volumi di origine e di destinazione utilizzano la stessa

impostazione VolumeMode; ad esempio, un volume di blocco può essere clonato solo su un altro volume di blocco.

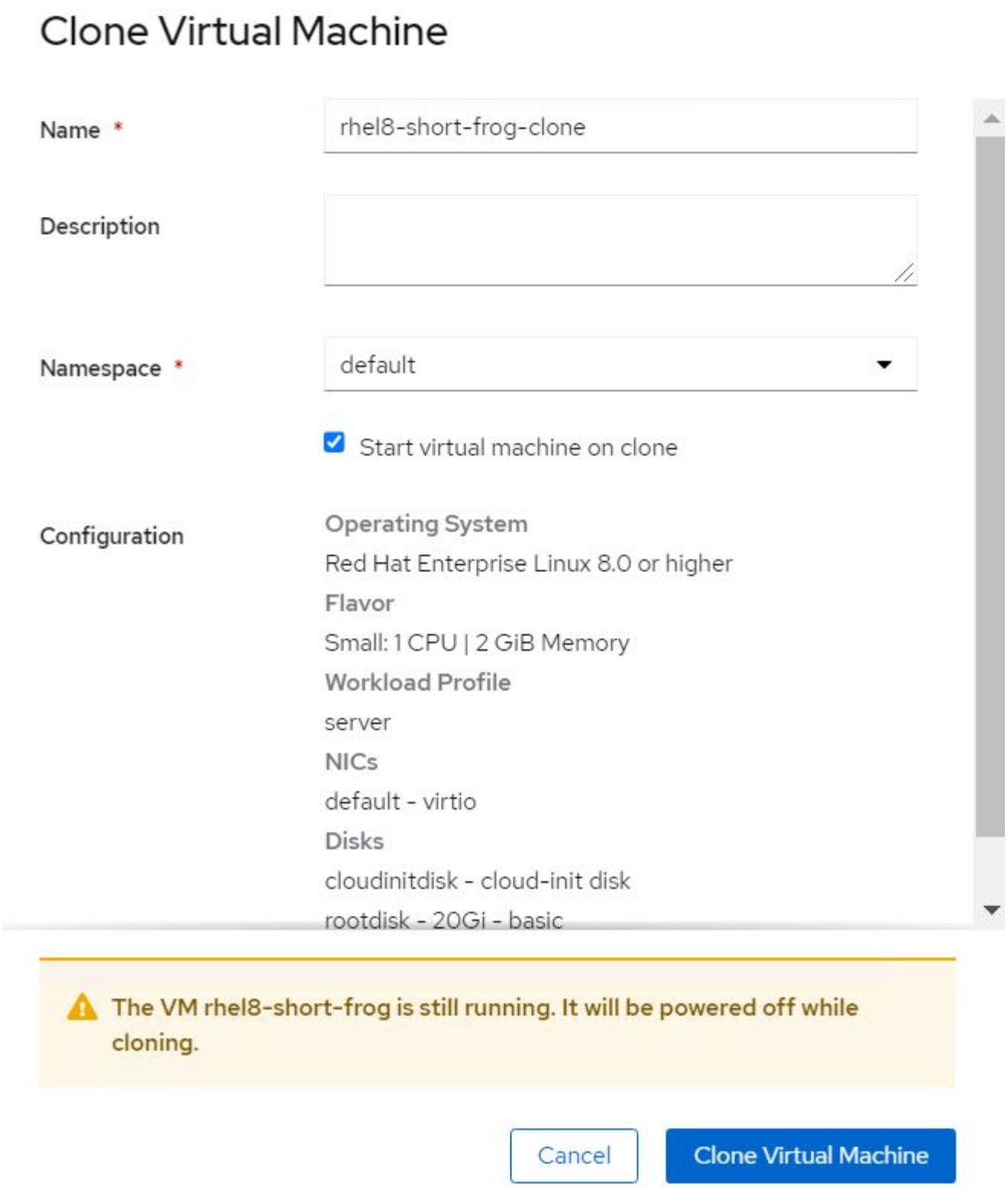
Le VM in un cluster OpenShift possono essere clonate in due modi:

1. Spegnendo la VM di origine
2. Mantenendo attiva la VM di origine

Spegnendo la VM di origine

Clonare una macchina virtuale esistente spegnendo la macchina virtuale è una funzionalità OpenShift nativa implementata con il supporto di Astra Trident. Per clonare una macchina virtuale, attenersi alla seguente procedura.

1. Accedere a workload > Virtualization > Virtual Machines (carichi di lavoro > virtualizzazione > macchine virtuali) e fare clic sui puntini di sospensione accanto alla macchina virtuale che si desidera clonare.
2. Fare clic su Clone Virtual Machine (Clona macchina virtuale) e fornire i dettagli della nuova macchina virtuale.

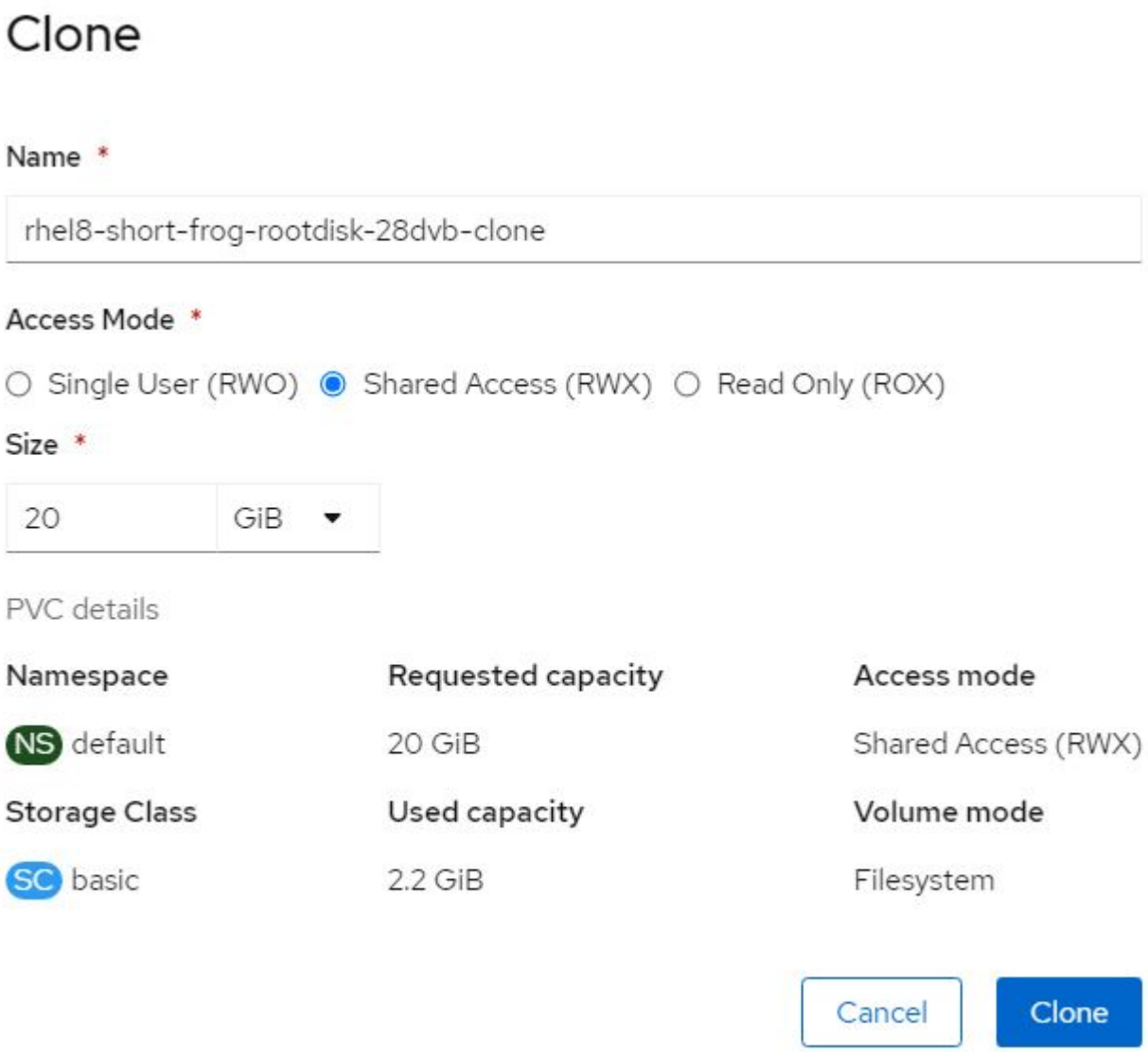


3. Fare clic su Clone Virtual Machine (Clona macchina virtuale) per chiudere la macchina virtuale di origine e avviare la creazione della macchina virtuale clone.
4. Al termine di questa fase, è possibile accedere e verificare il contenuto della VM clonata.

Mantenendo attiva la VM di origine

Una macchina virtuale esistente può anche essere clonata clonando il PVC esistente della macchina virtuale di origine e quindi creando una nuova macchina virtuale utilizzando il PVC clonato. Questo metodo non richiede l'arresto della VM di origine. Per clonare una macchina virtuale senza spegnerla, attenersi alla procedura riportata di seguito.

1. Accedere a Storage > PersistentVolumeClaims (Storage > PersistentVolumeClaims) e fare clic sui puntini di sospensione accanto al PVC collegato alla VM di origine.
2. Fare clic su Clone PVC e fornire i dettagli del nuovo PVC.



3. Quindi fare clic su Clone (Clona). In questo modo si crea un PVC per la nuova macchina virtuale.
4. Accedere a workload > virtualizzazione > macchine virtuali e fare clic su Create > with YAML (Crea > con YAML).
5. Nella sezione spec > template > spec > Volumes (specifiche > modello > specifiche > volumi), collegare il PVC clonato invece del disco container. Fornire tutti gli altri dettagli della nuova macchina virtuale in base alle proprie esigenze.

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

6. Fare clic su Create (Crea) per creare la nuova macchina virtuale.
7. Una volta creata correttamente la macchina virtuale, accedere e verificare che la nuova macchina sia un clone della macchina virtuale di origine.

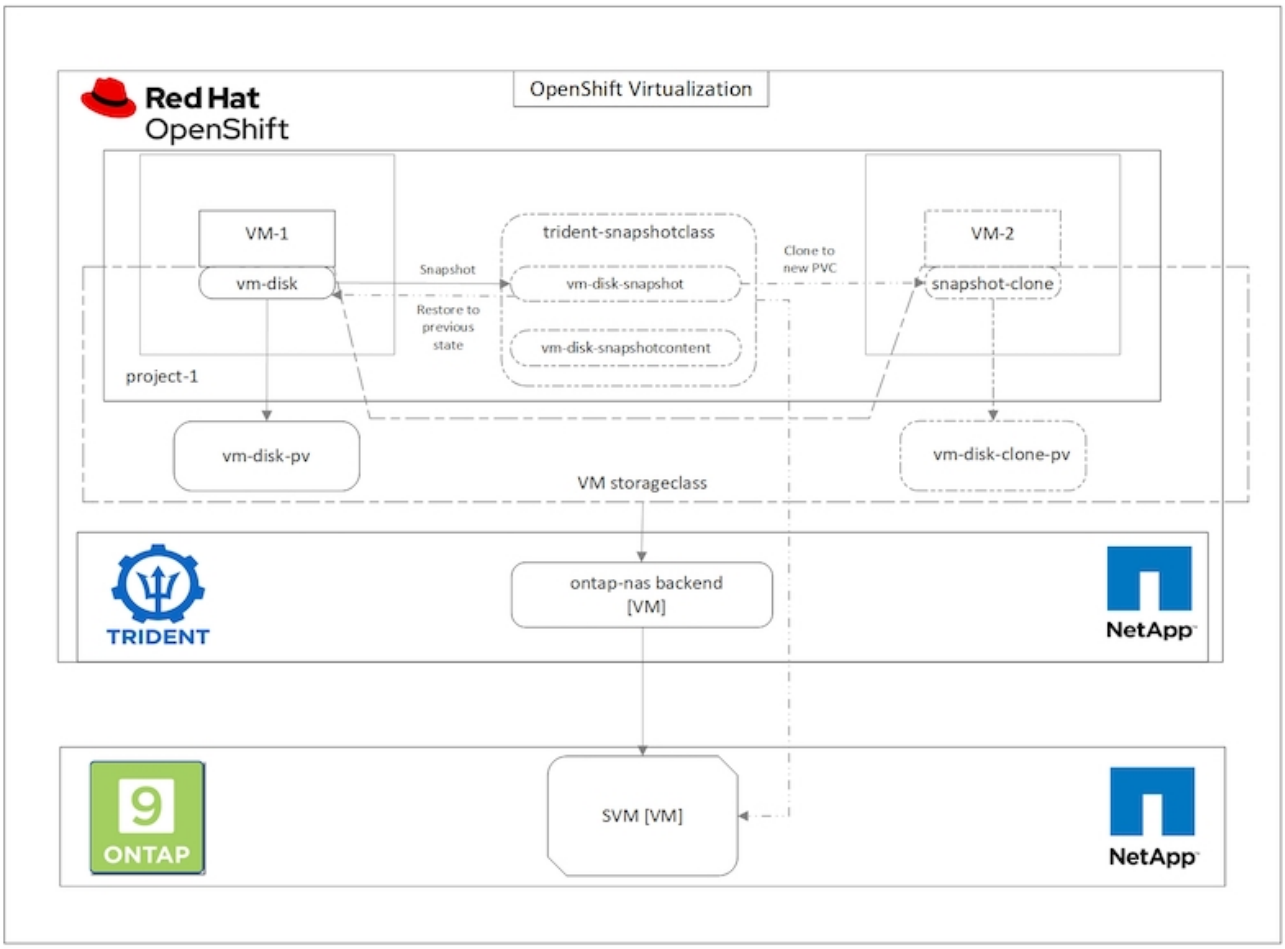
Workflow: Virtualizzazione Red Hat OpenShift con NetApp ONTAP

Creare una macchina virtuale da un'istantanea

Con Astra Trident e Red Hat OpenShift, gli utenti possono creare un'istantanea di un volume persistente su classi di storage fornite dall'IT. Con questa funzione, gli utenti possono eseguire una copia point-in-time di un volume e utilizzarlo per creare un nuovo volume o ripristinare lo stato precedente dello stesso volume. Ciò consente o supporta una varietà di casi di utilizzo, dal rollback ai cloni al ripristino dei dati.

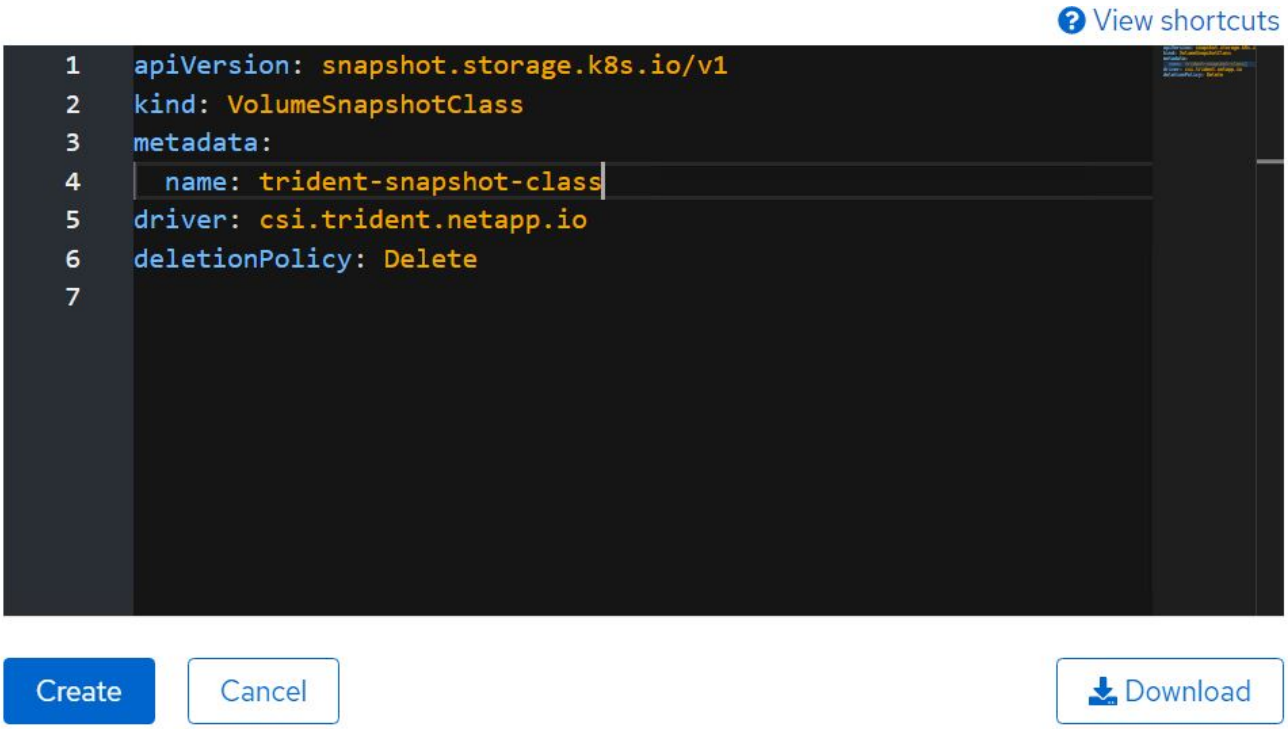
Per le operazioni Snapshot in OpenShift, è necessario definire le risorse VolumeSnapshotClass, VolumeSnapshot e VolumeSnapshotContent.

- Un VolumeSnapshotContent è lo snapshot effettivo preso da un volume nel cluster. Si tratta di una risorsa a livello di cluster analoga a PersistentVolume per lo storage.
- VolumeSnapshot è una richiesta per la creazione dello snapshot di un volume. È analogo a un PersistentVolumeClaim.
- VolumeSnapshotClass consente all'amministratore di specificare attributi diversi per un'istantanea VolumeSnapshot. Consente di avere attributi diversi per diversi snapshot acquisiti dallo stesso volume.

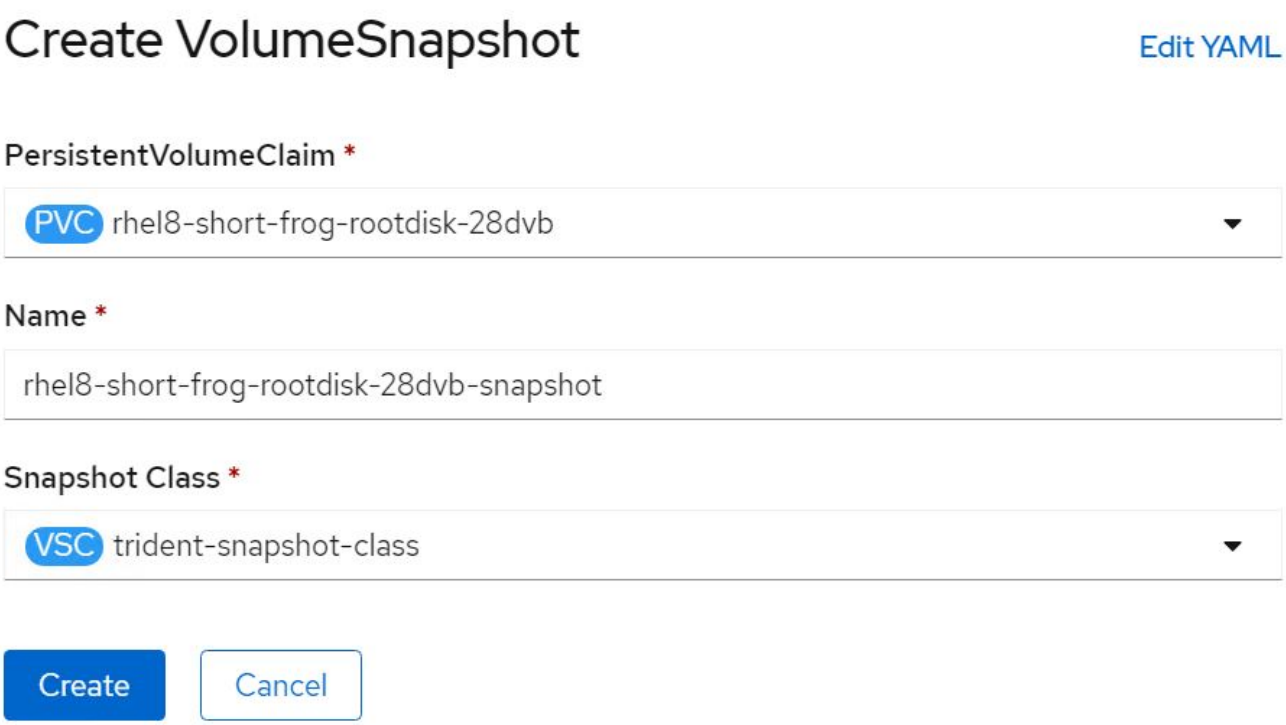


Per creare un'istantanea di una macchina virtuale, attenersi alla seguente procedura:

1. Creare una classe VolumeSnapshotClass da utilizzare per creare un'istantanea VolumeSnapshot. Accedere a Storage > VolumeSnapshotClasses e fare clic su Create VolumeSnapshotClass (Crea VolumeSnapshotClass).
2. Immettere il nome della classe Snapshot, immettere csi.trident.netapp.io per il driver e fare clic su Create (Crea).



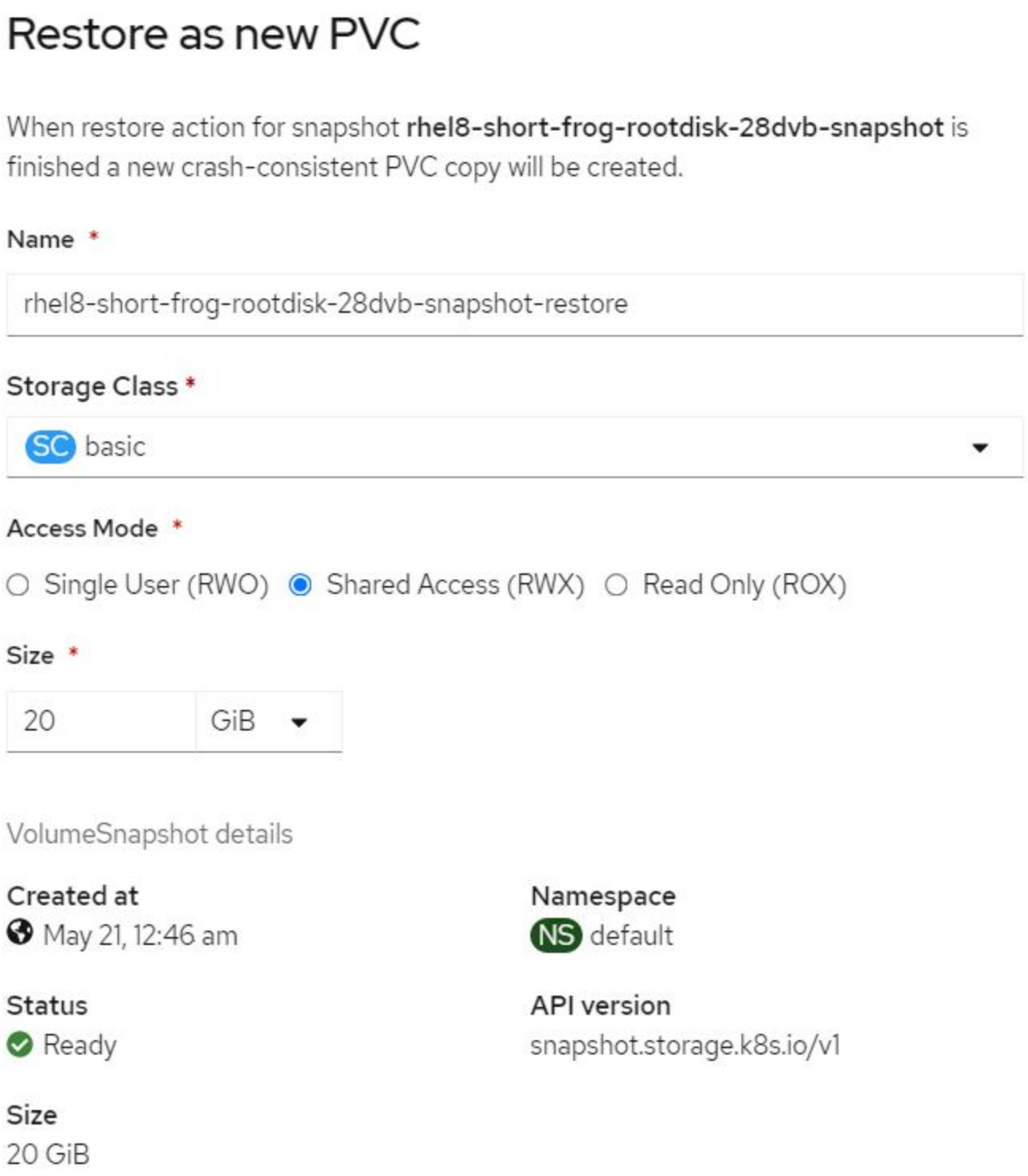
3. Identificare il PVC collegato alla VM di origine e creare un'istantanea del PVC. Selezionare `Storage > VolumeSnapshots` E fare clic su Create VolumeSnapshots (Crea snapshot Volume).
4. Selezionare il PVC per il quale si desidera creare l'istantanea, immettere il nome dell'istantanea o accettare il valore predefinito, quindi selezionare la VolumeSnapshotClass appropriata. Quindi fare clic su Create (Crea).



5. In questo modo viene creata l'istantanea del PVC in quel momento.

Creare una nuova macchina virtuale dall'istantanea

1. Innanzitutto, ripristinare l'istantanea in un nuovo PVC. Accedere a Storage > VolumeSnapshots (Storage > VolumeSnapshots), fare clic sui puntini di sospensione accanto all'istantanea che si desidera ripristinare e fare clic su Restore as new PVC (Ripristina come nuovo PVC).
2. Inserire i dettagli del nuovo PVC e fare clic su Restore (Ripristina). In questo modo si crea un nuovo PVC.



3. Quindi, creare una nuova macchina virtuale da questo PVC. Accedere a workload > virtualizzazione > macchine virtuali e fare clic su Create > with YAML (Crea > con YAML).

4. Nella sezione spec > template > spec > Volumes (specifiche > modello > specifiche > volumi), specificare il nuovo PVC creato da Snapshot anziché dal disco container. Fornire tutti gli altri dettagli della nuova macchina virtuale in base alle proprie esigenze.

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

5. Fare clic su Create (Crea) per creare la nuova macchina virtuale.
6. Una volta creata correttamente la macchina virtuale, accedere e verificare che la nuova macchina virtuale abbia lo stesso stato della macchina virtuale il cui PVC è stato utilizzato per creare lo snapshot al momento della creazione dello snapshot.

Workflow: Virtualizzazione Red Hat OpenShift con NetApp ONTAP

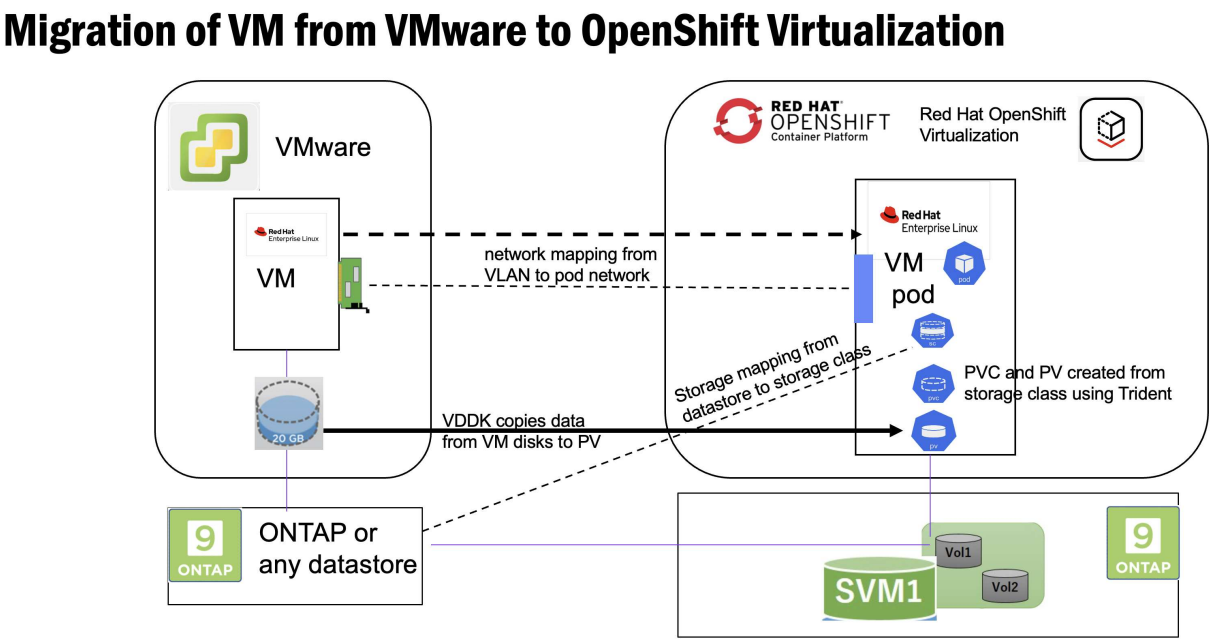
Migrazione di VM da VMware alla virtualizzazione OpenShift mediante Migration Toolkit for Virtualization

In questa sezione, vedremo come utilizzare l'Toolkit di migrazione per la virtualizzazione (MTV) per migrare le macchine virtuali da VMware alla virtualizzazione OpenShift eseguita sulla piattaforma contenitore OpenShift e integrata con lo storage NetApp ONTAP utilizzando Astra Trident.

Il seguente video mostra una dimostrazione della migrazione di una macchina virtuale RHEL da VMware alla virtualizzazione OpenShift utilizzando ontap-san per lo storage persistente.

Utilizzo di Red Hat MTV per migrare le VM alla virtualizzazione OpenShift con lo storage NetApp ONTAP

Il diagramma seguente mostra una vista di alto livello della migrazione di una VM da VMware a Red Hat OpenShift Virtualization.



Prerequisiti per la migrazione dei campioni

Su VMware

- È stata installata una macchina virtuale rhel 9 che utilizza rhel 9,3 con le seguenti configurazioni:
  - CPU: 2, memoria: 20 GB, disco rigido: 20 GB
  - credenziali utente: credenziali utente root e amministratore
- Dopo che la VM era pronta, il server postgresql è stato installato.
  - postgresql server è stato avviato e abilitato all'avvio

```
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
```

  - Sono stati aggiunti 2 database, 1 tabella e 1 riga nella tabella. Fare riferimento a. "qui" Per le istruzioni per l'installazione del server postgresql su RHEL e per la creazione di database e voci di tabella.

> Assicurarsi di avviare il server postgresql e abilitare il servizio all'avvio.

Sul quadro strumenti OpenShift

Le seguenti installazioni sono state completate prima di installare MTV:

- Gruppo OpenShift 4.13.34
- "Astra Trident 23,10"
- Multipath sui nodi del cluster abilitato per iSCSI (per storage ontap-san). Consultare il codice yaml fornito per creare un set di daemon che abiliti iSCSI su ciascun nodo del cluster.
- Backend Trident e classe di storage per SAN ONTAP utilizzando iSCSI. Vedere i file yaml forniti per il backend tridente e la classe di archiviazione.
- "Virtualizzazione OpenShift"

Per installare iscsi e multipath sui nodi del cluster OpenShift, utilizzare il file yaml riportato di seguito
Preparazione dei nodi cluster per iSCSI

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

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

Utilizzare il seguente file yaml per creare una configurazione back-end tridente per l'utilizzo dello storage san ONTAP
Backend Trident per iSCSI

```yaml
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
```

Utilizzare il seguente file yaml per creare la configurazione della classe di archiviazione tridente per l'utilizzo dello storage san ONTAP
Classe di storage Trident per iSCSI

```yaml
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
```

Installare MTV

A questo punto è possibile installare il Migration Toolkit for Virtualization (MTV). Fare riferimento alle istruzioni fornite "qui" per informazioni sull'installazione.

L'interfaccia utente di Migration Toolkit for Virtualization (MTV) è integrata nella console Web OpenShift. È possibile fare riferimento "qui" per iniziare a utilizzare l'interfaccia utente per varie attività.

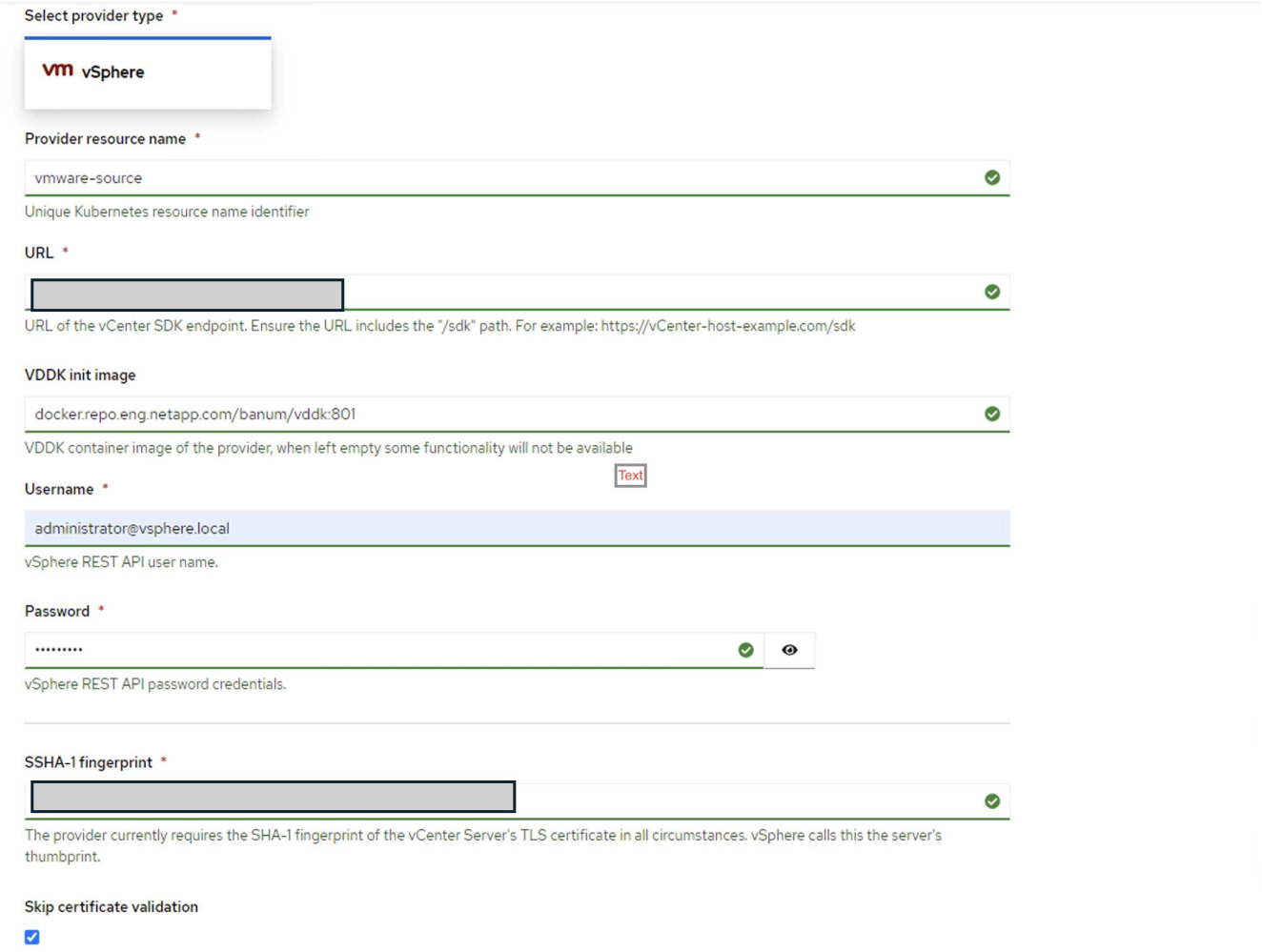
Creare il fornitore di origine

Per migrare RHEL VM da VMware a OpenShift Virtualization, è necessario innanzitutto creare il provider di origine per VMware. Fare riferimento alle istruzioni "qui" per creare il provider di origine.

Per creare il provider di origine VMware sono necessari i seguenti elementi:

- URL vCenter
- Credenziali vCenter
- Identificazione utente del server vCenter
- Immagine VDDK in un repository

Creazione del provider di origine campione:



> MTV (Migration Toolkit for Virtualization) utilizza VMware Virtual Disk Development Kit (VDDK) SDK per accelerare il trasferimento dei dischi virtuali da VMware vSphere. Pertanto, si consiglia vivamente di creare un'immagine VDDK, anche se facoltativa.
> Per utilizzare questa funzione, è necessario scaricare VMware Virtual Disk Development Kit (VDDK), creare un'immagine VDDK e inviare l'immagine VDDK al registro delle immagini.

Seguire le istruzioni fornite "qui" Per creare e inviare l'immagine VDDK a un registro accessibile dal cluster OpenShift.

Crea fornitore di destinazione

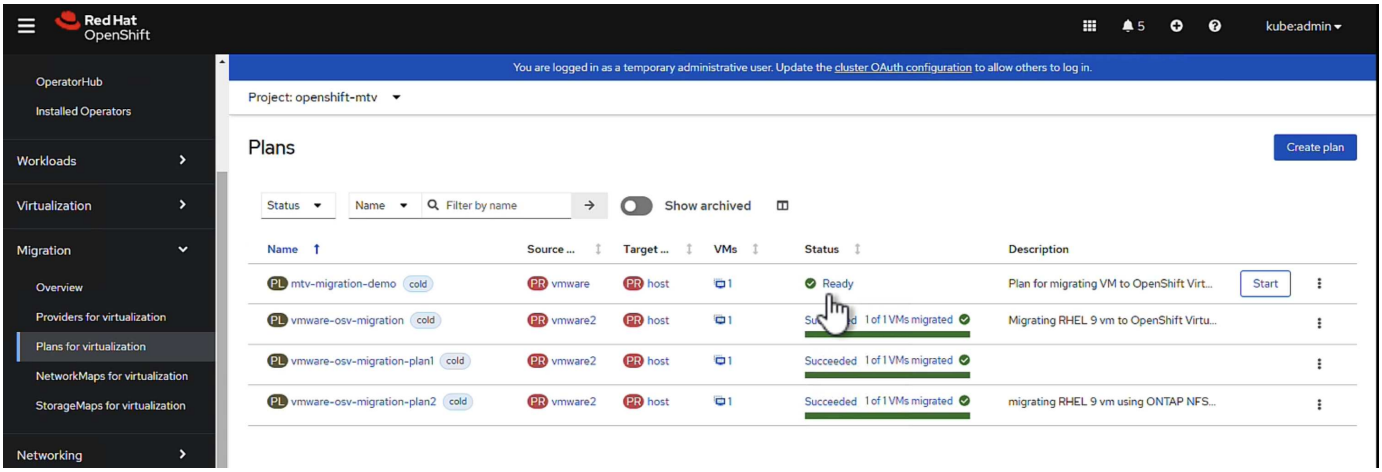
Il cluster host viene aggiunto automaticamente in quanto il provider di virtualizzazione OpenShift è il provider di origine.

Creare un piano di migrazione

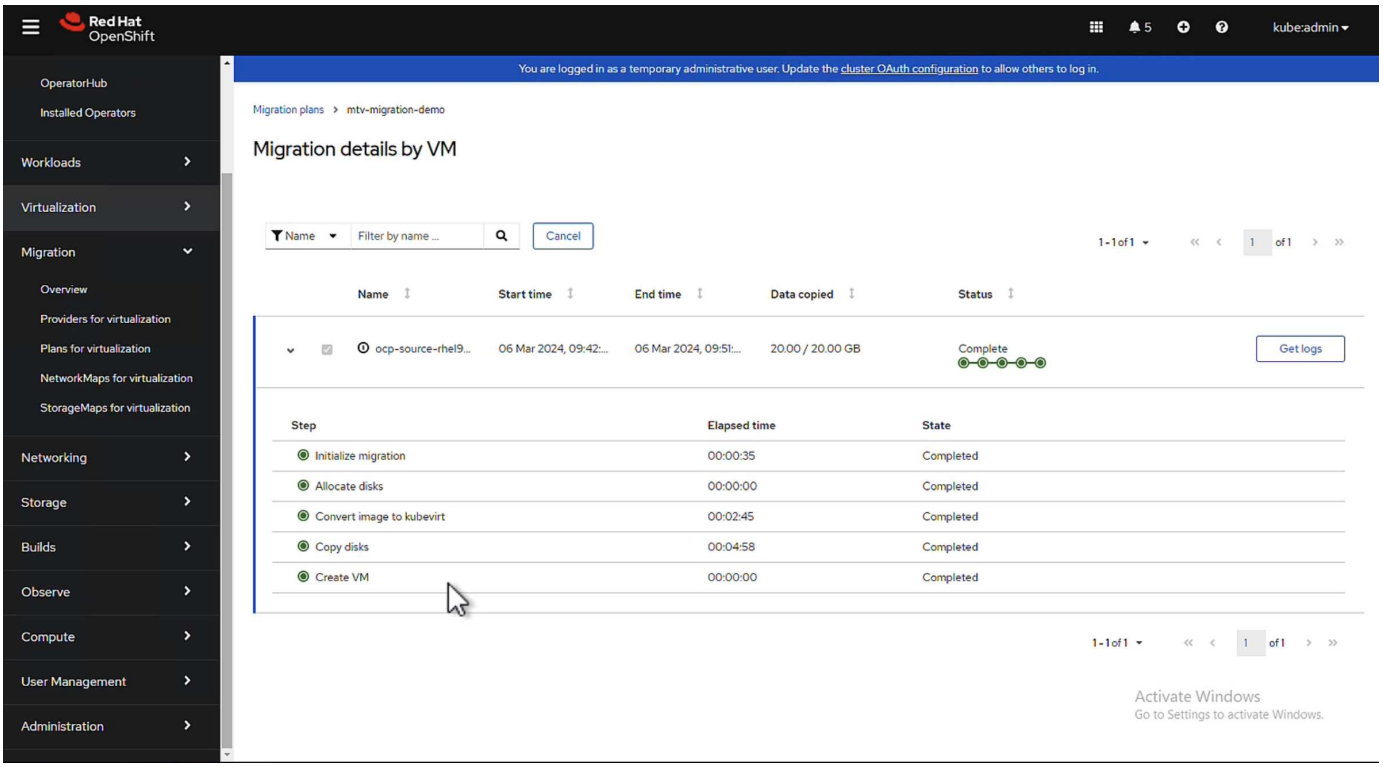
Seguire le istruzioni fornite "qui" per creare un piano di migrazione.

Durante la creazione di un piano, è necessario creare quanto segue se non è già stato creato:

- Mappatura di rete per mappare la rete di origine alla rete di destinazione.
- Mappatura dello storage per mappare il datastore di origine alla classe dello storage di destinazione. Per questo puoi scegliere la classe dello storage ontap-san.
  Una volta creato il piano di migrazione, lo stato del piano dovrebbe mostrare **Ready** e si dovrebbe ora essere in grado di **Start** il piano.



Facendo clic su **Start** verrà eseguita una sequenza di passaggi per completare la migrazione della VM.



Al termine di tutte le fasi, è possibile visualizzare le VM migrate facendo clic su **macchine virtuali** in **virtualizzazione** nel menu di navigazione a sinistra.
Vengono fornite le istruzioni per accedere alle macchine virtuali "qui".

È possibile accedere alla macchina virtuale e verificare il contenuto dei database posgresql. I database, le tabelle e le voci nella tabella devono essere uguali a quelli creati sulla macchina virtuale di origine.

Protezione dei dati per la virtualizzazione OpenShift

Protezione dei dati delle VM in OpenShift Virtualization con OpenShift API for Data Protection (OADP)

Autore: Banu Sundhar, NetApp

Questa sezione del documento di riferimento fornisce dettagli per la creazione di backup di macchine virtuali utilizzando l'API OpenShift per la protezione dei dati (OADP) con Velero su NetApp ONTAP S3 o NetApp StorageGRID S3. I backup dei volumi persistenti (PVS) dei dischi della macchina virtuale vengono creati utilizzando gli Snapshot CSI Astra Trident.

Le macchine virtuali nell'ambiente di virtualizzazione OpenShift sono applicazioni containerizzate che vengono eseguite nei nodi di lavoro della piattaforma container OpenShift. È importante proteggere i metadati delle macchine virtuali e i dischi persistenti delle macchine virtuali, in modo che in caso di perdita o danneggiamento possano essere ripristinati.

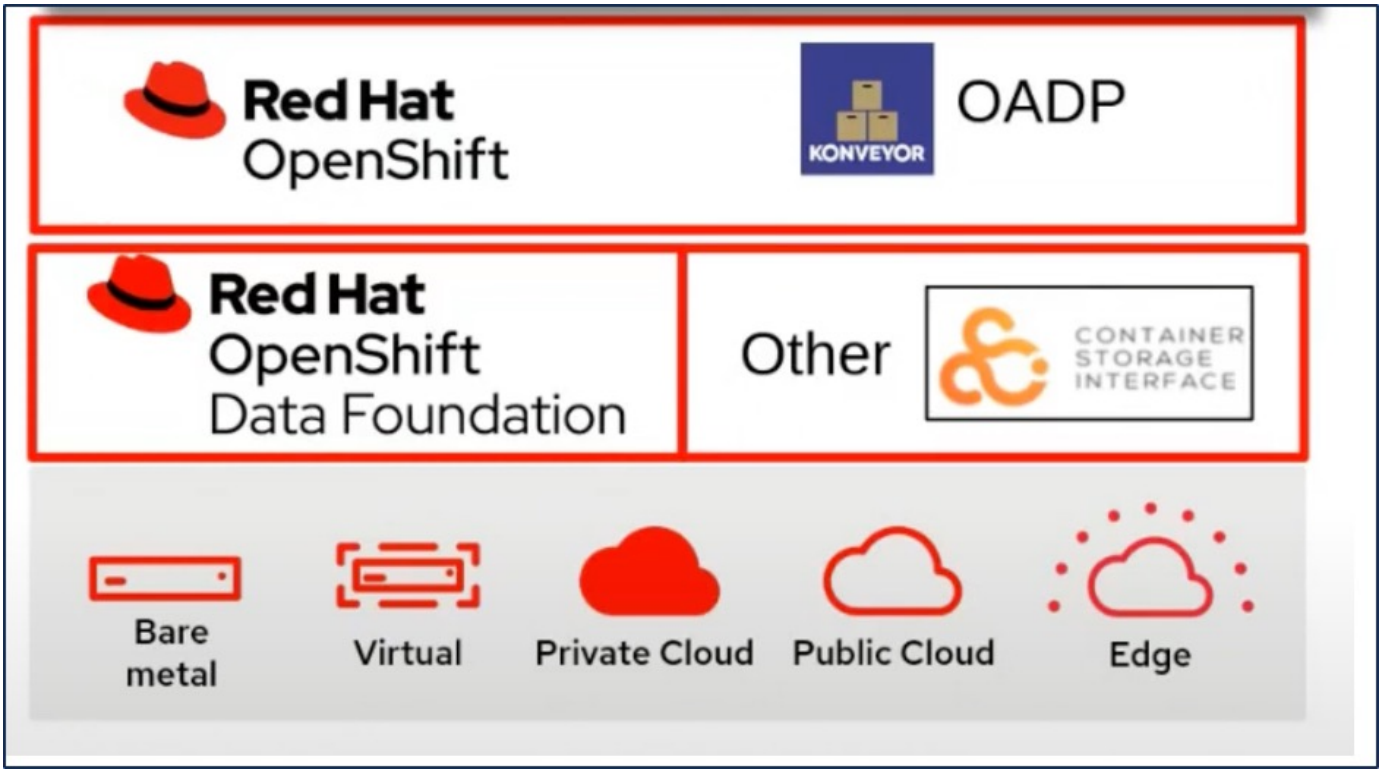
I dischi persistenti delle macchine virtuali di virtualizzazione OpenShift possono essere sottoposti a backup dallo storage ONTAP integrato nel cluster OpenShift utilizzando "CSI Astra Trident". In questa sezione usiamo "OpenShift API per la protezione dei dati (OADP)" Per eseguire il backup delle VM, inclusi i relativi volumi di dati su

- Storage a oggetti ONTAP

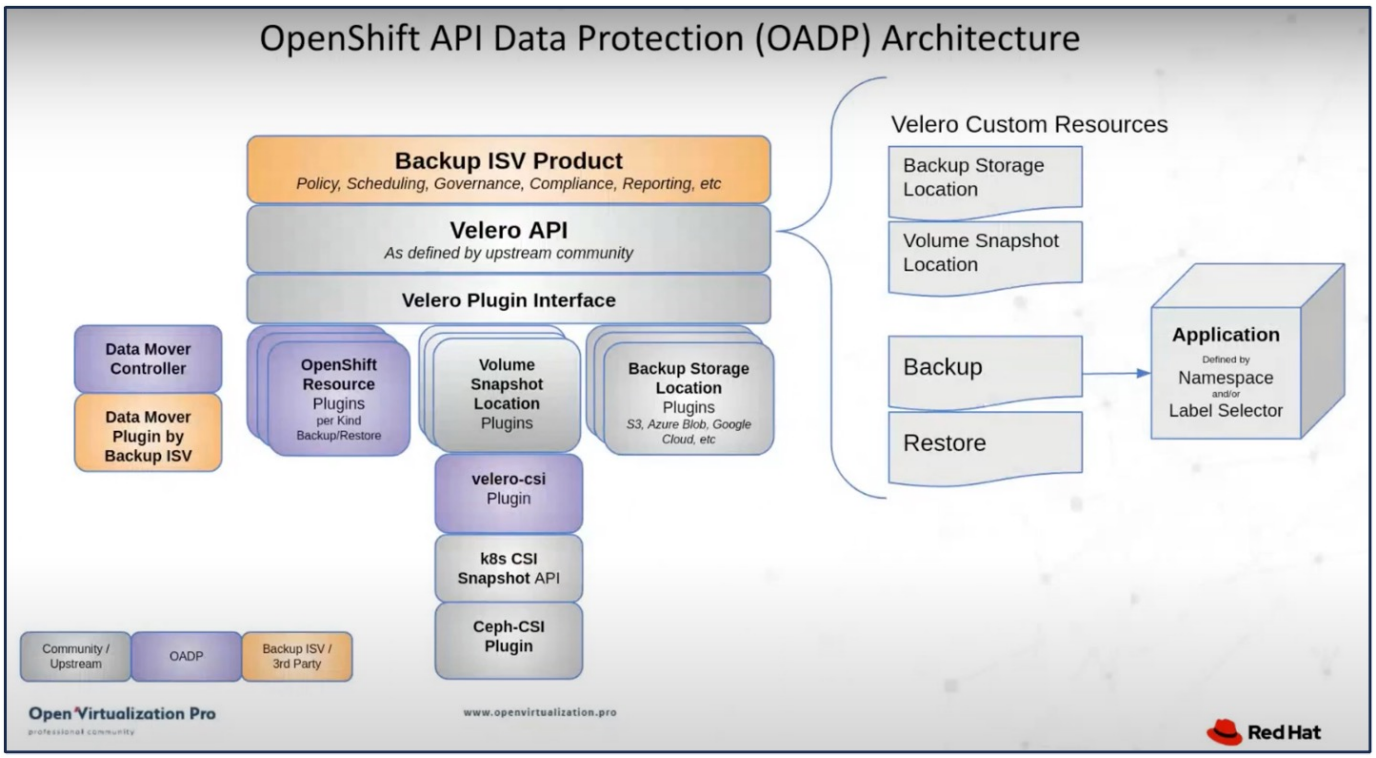
- StorageGRID

Quindi, eseguiamo il ripristino dal backup quando necessario.

OADP consente il backup, il ripristino e il disaster recovery delle applicazioni su un cluster OpenShift. I dati che possono essere protetti con OADP includono oggetti risorsa Kubernetes, volumi persistenti e immagini interne.



Red Hat OpenShift ha sfruttato le soluzioni sviluppate dalla comunità OpenSource per la protezione dei dati. "Velero" È uno strumento open-source per eseguire backup e ripristino in tutta sicurezza, eseguire disaster recovery e migrare risorse del cluster e volumi persistenti di Kubernetes. Per utilizzare Velero facilmente, OpenShift ha sviluppato l'operatore OADP e il plugin Velero per integrarsi con i driver di storage CSI. Il nucleo delle API OADP esposte si basa sulle API di Velero. Dopo aver installato e configurato l'operatore OADP, le operazioni di backup/ripristino che possono essere eseguite si basano sulle operazioni esposte dall'API Velero.



OADP 1,3 è disponibile dall'hub operatore del gruppo OpenShift 4,12 e versioni successive. Dispone di un Data Mover integrato che può spostare gli snapshot di volume CSI in un archivio di oggetti remoto. In questo modo è possibile ottenere portabilità e durata spostando le snapshot in una posizione di storage a oggetti durante il backup. Le snapshot sono quindi disponibili per il ripristino dopo un disastro.

Di seguito sono riportate le versioni dei vari componenti utilizzati per gli esempi di questa sezione

- Gruppo OpenShift 4,14
- OpenShift Virtualization installato tramite OperatorOpenShift Virtualization Operator fornito da Red Hat
- OADP Operator 1,13 fornito da Red Hat
- Velero CLI 1,13 per Linux
- Astra Trident 24,02
- ONTAP 9,12

"CSI Astra Trident"
"OpenShift API per la protezione dei dati (OADP)"
"Velero"

Installazione dell'operatore OpenShift API for Data Protection (OADP)

Prerequisiti

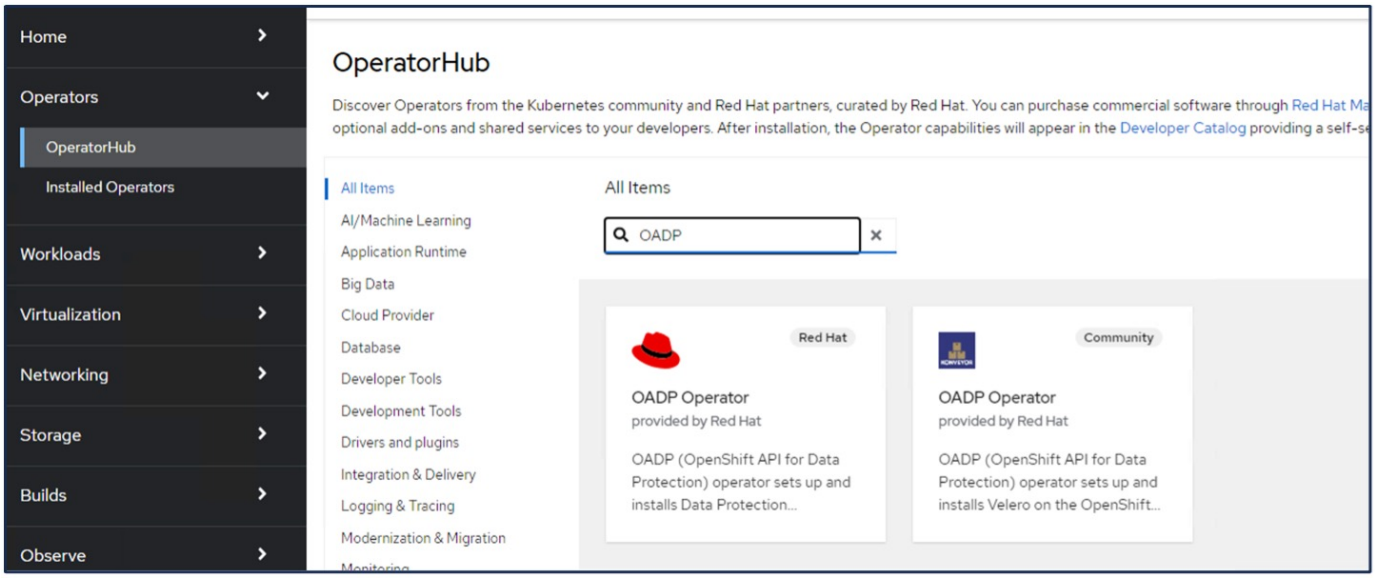
- Un cluster Red Hat OpenShift (versione successiva alla 4,12) installato in un'infrastruttura bare-metal con nodi di lavoro RHCOS
- Un cluster NetApp ONTAP integrato con il cluster utilizzando Astra Trident
- Un backend Trident configurato con una SVM sul cluster ONTAP
- StorageClass configurato sul cluster OpenShift con Astra Trident come provisioner

- Classe Snapshot Trident creata nel cluster
- Accesso cluster-admin al cluster Red Hat OpenShift
- Accesso amministrativo al cluster NetApp ONTAP
- Operatore di virtualizzazione OpenShift installato e configurato
- VM implementate in uno spazio dei nomi su OpenShift Virtualization
- Una workstation di amministrazione con tridentctl e oc tools installati e aggiunti al percorso dei dollari

> Se si desidera eseguire un backup di una macchina virtuale quando è in esecuzione, è necessario installare l'agente guest QEMU su tale macchina virtuale. Se si installa la macchina virtuale utilizzando un modello esistente, l'agente QEMU viene installato automaticamente. QEMU consente all'agente ospite di disattivare i dati in-flight nel sistema operativo guest durante il processo di snapshot ed evitare possibili danneggiamenti dei dati. Se QEMU non è installato, è possibile arrestare la macchina virtuale prima di eseguire un backup.

Procedura per l'installazione dell'operatore OADP

1. Andare all'Operator Hub del cluster e selezionare Red Hat OADP operator. Nella pagina Installa, utilizzare tutte le selezioni predefinite e fare clic su Installa. Nella pagina successiva, utilizzare nuovamente tutte le impostazioni predefinite e fare clic su Installa. L'operatore OADP sarà installato nello spazio dei nomi openshift-adp.



OADP Operator

1.3.0 provided by Red Hat

Install

Channel

stable-1.3

Version

1.3.0

Capability level

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

Source

Red Hat

Provider

Red Hat

Infrastructure features

Disconnected

OpenShift API for Data Protection (OADP) operator sets up and installs Velero on the OpenShift platform, allowing users to backup and restore applications.

Backup and restore Kubernetes resources and internal images, at the granularity of a namespace, using a version of Velero appropriate for the installed version of OADP.

OADP backs up Kubernetes objects and internal images by saving them as an archive file on object storage. OADP backs up persistent volumes (PVs) by creating snapshots with the native cloud snapshot API or with the Container Storage Interface (CSI). For cloud providers that do not support snapshots, OADP backs up resources and PV data with Restic or Kopia.

- Installing OADP for application backup and restore
- Installing OADP on a ROSA cluster and using STS, please follow the Getting Started Steps 1-3 in order to obtain the role ARN needed for using the standardized STS configuration flow via OLM
- Frequently Asked Questions

Project: All Projects

Installed Operators

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

Name | Search by name... /

Name	Namespace	Managed Namespaces	Status
OpenShift Virtualization 4.14.4 provided by Red Hat	NS openshift-cnv	NS openshift-cnv	Succeeded Up to date
OADP Operator 1.3.0 provided by Red Hat	NS openshift-adp	NS openshift-adp	Succeeded Up to date
Package Server 0.0.1-snapshot provided by	NS openshift-operator-lifecycle-manager	NS openshift-operator-lifecycle-manager	Succeeded

Prerequisiti per la configurazione di Velero con i dettagli di ONTAP S3

Una volta completata l'installazione dell'operatore, configurare l'istanza di Velero.
Velero può essere configurato per utilizzare l'archiviazione oggetti compatibile con S3. Configurare ONTAP S3 utilizzando le procedure illustrate nella "Sezione Gestione dello storage a oggetti della documentazione di ONTAP". Per l'integrazione con Velero, sono necessarie le seguenti informazioni della configurazione di ONTAP S3.

- Un'interfaccia logica (LIF) che può essere usata per accedere a S3
- Credenziali utente per accedere a S3 che include la chiave di accesso e la chiave di accesso segreta
- Un nome bucket in S3 per i backup con autorizzazioni di accesso per l'utente
- Per un accesso sicuro all'archiviazione a oggetti, è necessario installare il certificato TLS sul server di archiviazione a oggetti.

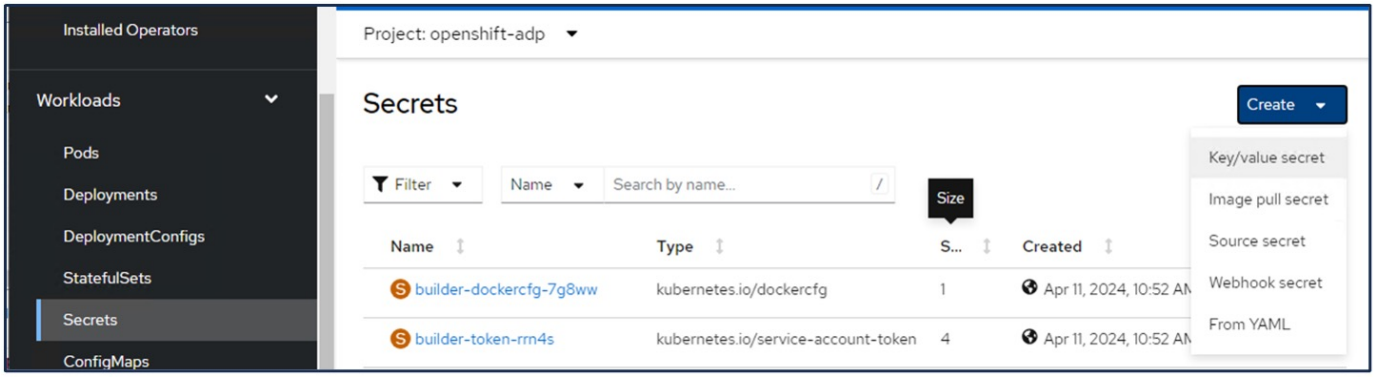
Prerequisiti per la configurazione di Velero con i dettagli di StorageGRID S3

Velero può essere configurato per utilizzare l'archiviazione oggetti compatibile con S3. È possibile configurare StorageGRID S3 utilizzando le procedure illustrate nella "Documentazione StorageGRID". Per l'integrazione con Velero, sono necessarie le seguenti informazioni della configurazione di StorageGRID S3.

- L'endpoint che può essere utilizzato per accedere a S3
- Credenziali utente per accedere a S3 che include la chiave di accesso e la chiave di accesso segreta
- Un nome bucket in S3 per i backup con autorizzazioni di accesso per l'utente
- Per un accesso sicuro all'archiviazione a oggetti, è necessario installare il certificato TLS sul server di archiviazione a oggetti.

Procedura di configurazione di Velero

- Innanzitutto, creare un segreto per una credenziale utente ONTAP S3 o per le credenziali utente StorageGRID tenant. Verrà utilizzato per configurare Velero in un secondo momento. È possibile creare un segreto dall'interfaccia CLI o dalla console Web.
  Per creare un segreto dalla console Web, selezionare segreti, quindi fare clic su chiave/valore segreto. Fornire i valori per il nome della credenziale, la chiave e il valore come mostrato. Assicurarsi di utilizzare l'ID chiave di accesso e la chiave di accesso segreta dell'utente S3. Assegnare un nome appropriato al segreto. Nell'esempio seguente, viene creato un segreto con credenziali utente di ONTAP S3 denominato credenziali ontap-S3.



Project: openshift-adp

Create key/value secret

Key/value secrets let you inject sensitive data into your application as files or environment variables.

Secret name *

cloud-credentials

Unique name of the new secret.

Key *

cloud

Value

Browse...

Drag and drop file with your value here or browse to upload it.

```
[default]
aws_access_key_id=<Access Key Id of S3 user>
aws_secret_access_key=<Secret Access Key of S3 user>
```

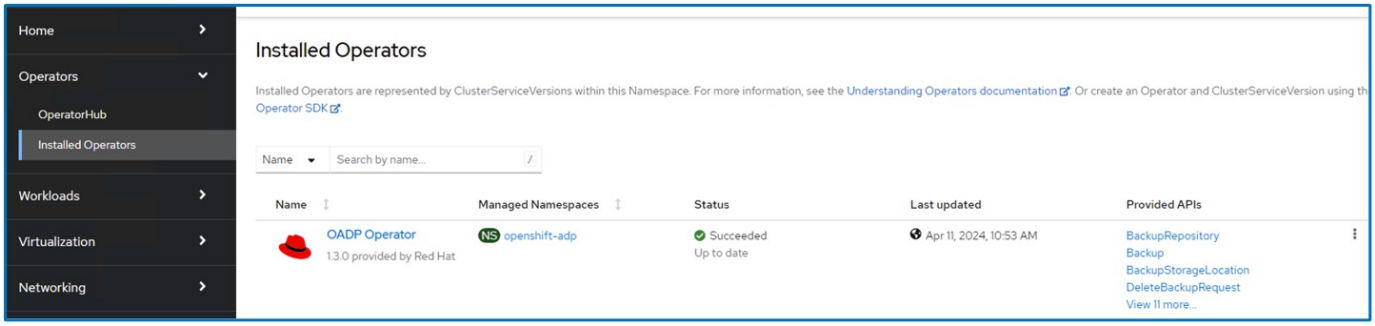
Add key/value

Create Cancel

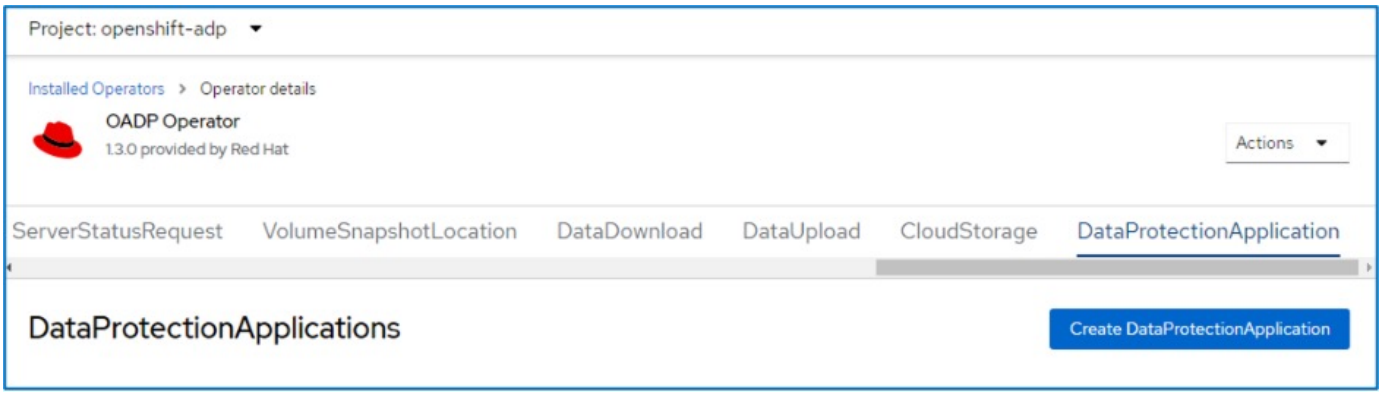
Per creare un segreto denominato sg-S3-credenziali dall'interfaccia CLI, è possibile utilizzare il seguente comando.

```
# oc create secret generic cloud-credentials --namespace openshift-adp --
from-file cloud=cloud-credentials.txt

credentials.txt file contains the Access Key Id and the Secret Access Key of
the S3 user in the following format:

[default]
aws_access_key_id=<Access Key Id of S3 user>
aws_secret_access_key=<Secret Access Key of S3 user>
```

- Quindi, per configurare Velero, selezionare Installed Operators dalla voce di menu in Operators, fare clic sull'operatore OADP, quindi selezionare la scheda DataProtectionApplication.



Fare clic su Create DataProtectionApplication. Nella vista modulo, specificare un nome per l'applicazione DataProtection o utilizzare il nome predefinito.



Passare ora alla visualizzazione YAML e sostituire le informazioni sulle specifiche come mostrato negli esempi di file yaml riportati di seguito.

Esempio di file yaml per la configurazione di Velero con ONTAP S3 come backupLocation

```
spec:
  backupLocations:
    - velero:
        config:
          insecureSkipTLSVerify: 'true' ->use this for https communication
with ONTAP S3
          profile: default
          region: us-east
          s3ForcePathStyle: 'True' ->This allows use of IP in s3URL
          s3Url: 'https://10.xx.xx.xx' ->Ensure TLS certificate for S3 is
configured
        credential:
          key: cloud
          name: ontap-s3-credentials ->previously created secret
        default: true
        objectStorage:
          bucket: velero ->Your bucket name previously created in S3 for
backups
          prefix: demobackup ->The folder that will be created in the
bucket
        provider: aws
  configuration:
    nodeAgent:
      enable: true
      uploaderType: kopia
      #default Data Mover uses Kopia to move snapshots to Object Storage
    velero:
      defaultPlugins:
        - csi ->Add this plugin
        - openshift
        - aws
        - kubevirt ->Add this plugin
```

File yaml di esempio per la configurazione di Velero con StorageGRID S3 come backupLocation e snapshotLocation

```
spec:
  backupLocations:
    - velero:
        config:
          insecureSkipTLSVerify: 'true'
          profile: default
          region: us-east-1 ->region of your StorageGrid system
          s3ForcePathStyle: 'True'
          s3Url: 'https://172.21.254.25:10443' ->the IP used to access S3
        credential:
          key: cloud
          name: sg-s3-credentials ->secret created earlier
        default: true
        objectStorage:
          bucket: velero
          prefix: demobackup
        provider: aws
  configuration:
    nodeAgent:
      enable: true
      uploaderType: kopia
    velero:
      defaultPlugins:
        - csi
        - openshift
        - aws
        - kubevirt
```

La sezione delle specifiche nel file yaml deve essere configurata in modo appropriato per i seguenti parametri, come nell'esempio precedente

BackupLocations
ONTAP S3 o StorageGRID S3 (con le relative credenziali e altre informazioni come mostrato in yaml) è configurato come BackupLocation predefinito per velero.

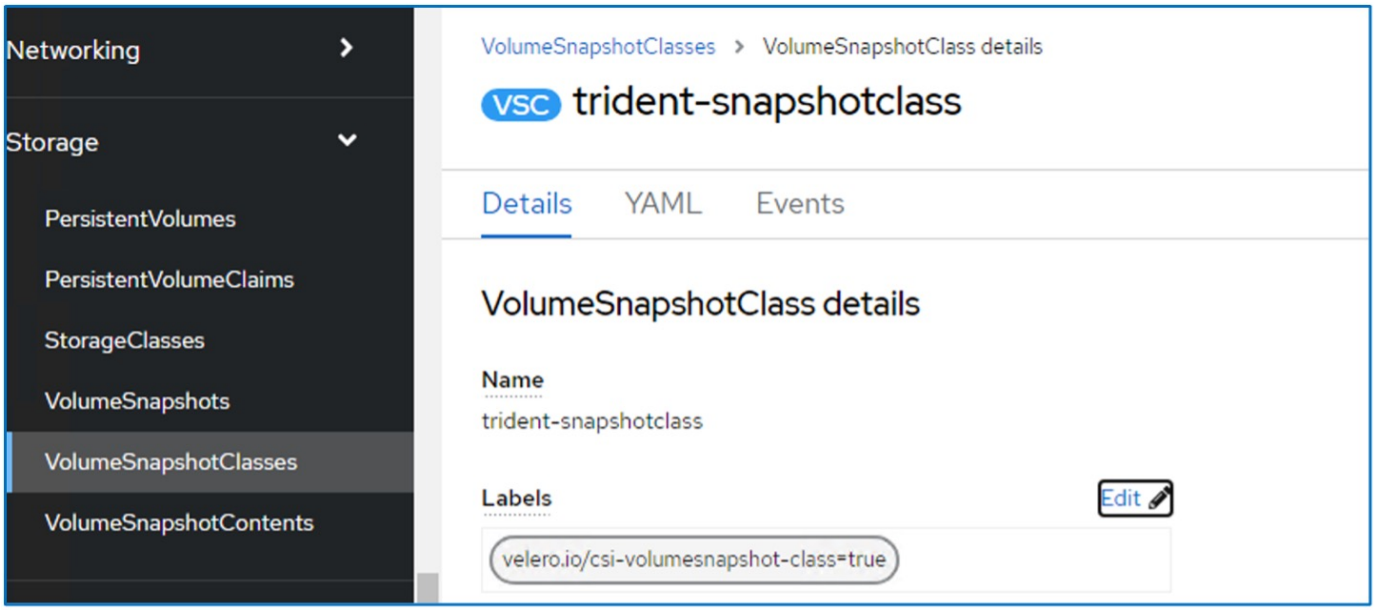
SnapshotLocations
Se si utilizzano gli snapshot Container Storage Interface (CSI), non è necessario specificare una posizione dello snapshot perché si creerà un VolumeSnapshotClass CR per registrare il driver CSI. Nel nostro esempio, si utilizza Astra Trident CSI e in precedenza si è creato VolumeSnapShotClass CR utilizzando il driver Trident CSI.

Attiva plugin CSI

Aggiungere csi ai prefaultPlugin per Velero per eseguire il backup dei volumi persistenti con gli snapshot CSI. I plug-in di Velero CSI, per eseguire il backup dei PVC supportati da CSI, sceglieranno VolumeSnapshotClass nel cluster su cui è impostata l'etichetta **velero.io/csi-volumesnapshot-class**. Per questo

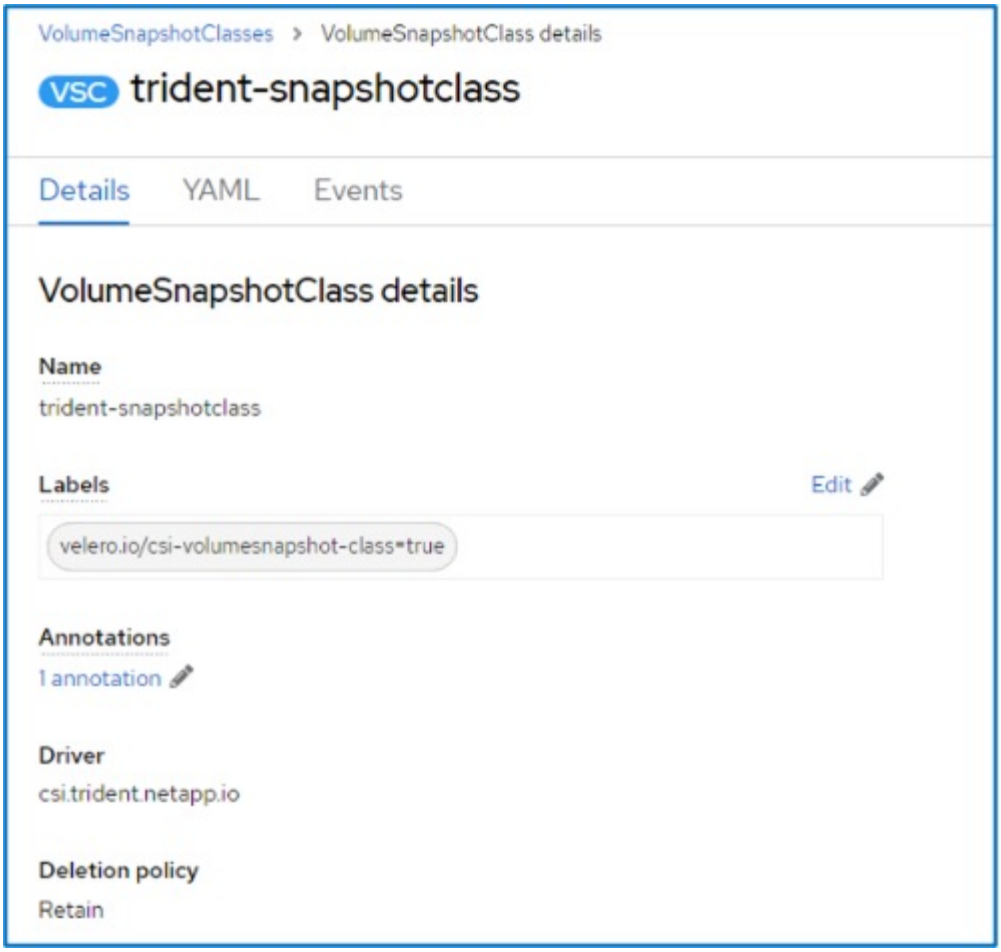
- È necessario creare il tridente VolumeSnapshotClass.

- Modificare l'etichetta della classe trident-snapshotclass e impostarla su **velero.io/csi-volumesnapshot-class=true** come mostrato di seguito.

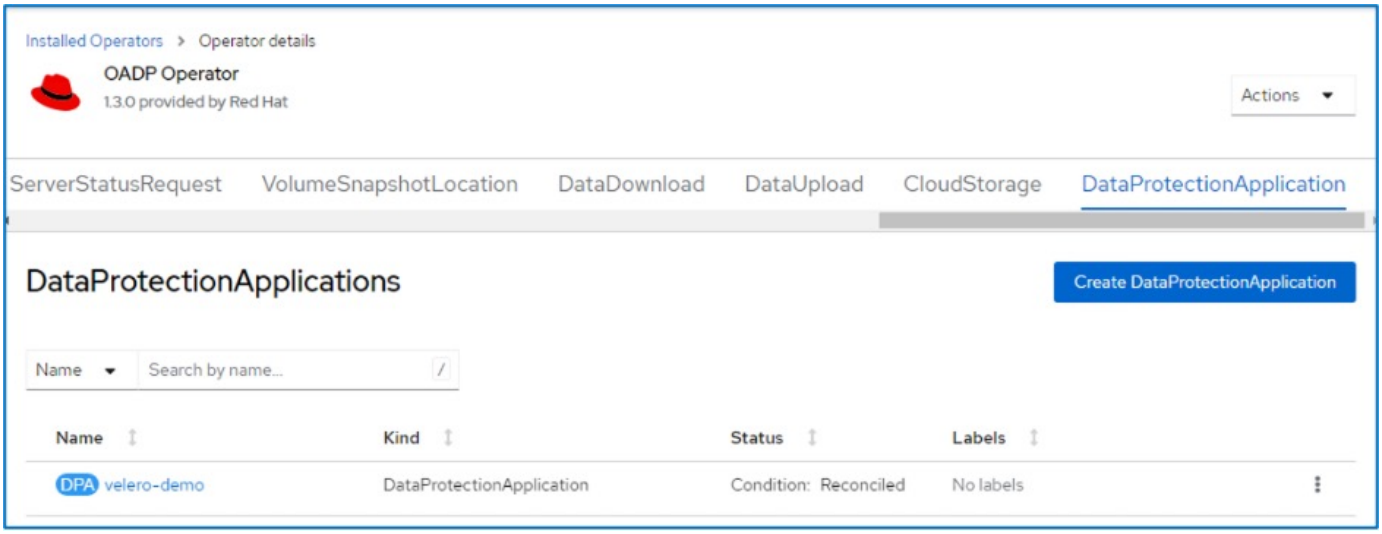


Verificare che gli snapshot possano persistere anche se gli oggetti VolumeSnapshot vengono eliminati. A tale scopo, impostare **deletionPolicy** su Retain. In caso contrario, l'eliminazione di uno spazio dei nomi perderà completamente tutti i PVC di cui è stato eseguito il backup.

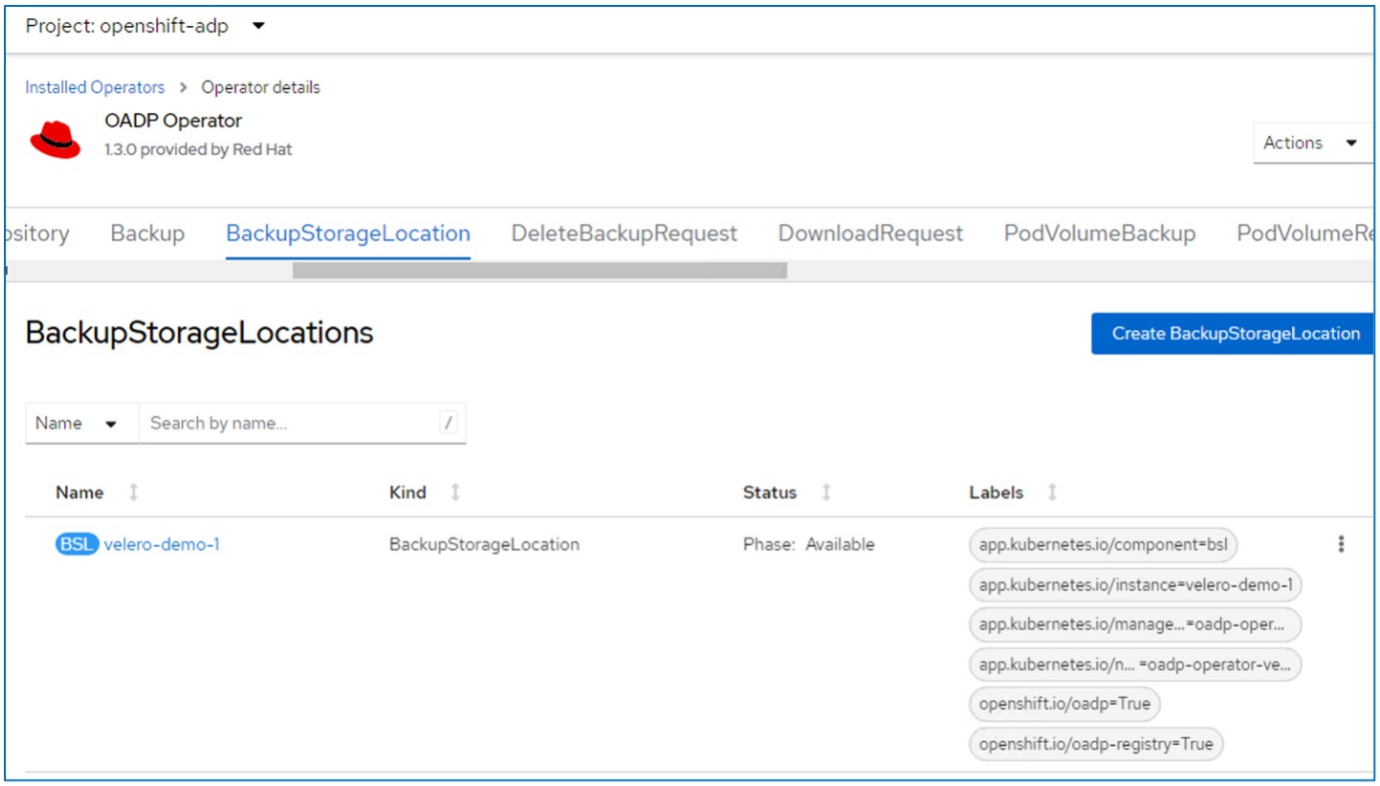
```
apiVersion: snapshot.storage.k8s.io/v1
kind: VolumeSnapshotClass
metadata:
  name: trident-snapshotclass
driver: csi.trident.netapp.io
deletionPolicy: Retain
```



Verificare che DataProtectionApplication sia stato creato e che sia in condizioni:riconciliato.



L'operatore OADP creerà un BackupStorageLocation corrispondente. Questo verrà utilizzato durante la creazione di un backup.



Creazione di backup su richiesta per le VM in OpenShift Virtualization

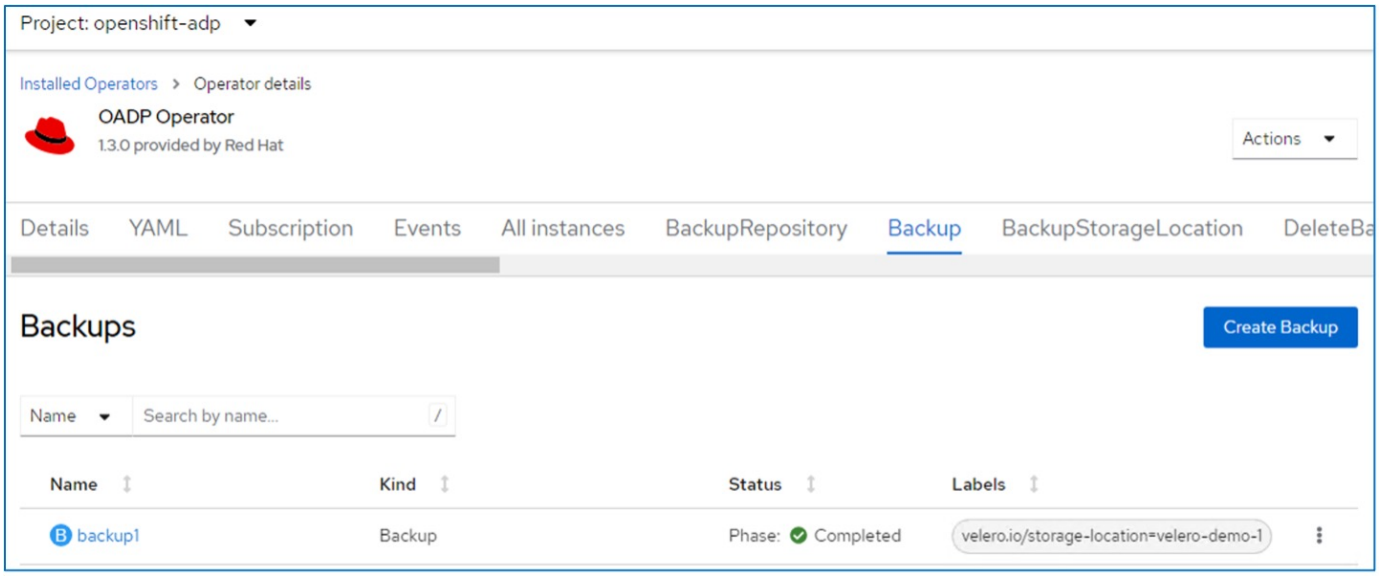
Procedura per creare un backup di una VM

Per creare un backup su richiesta dell'intera VM (metadati VM e dischi VM), fare clic sulla scheda **Backup**. In questo modo viene creata una risorsa personalizzata di backup (CR). Viene fornito un yaml di esempio per creare la CR di backup. Utilizzando questo yaml, verrà eseguito il backup della VM e dei relativi dischi nello spazio dei nomi specificato. È possibile impostare parametri aggiuntivi come illustrato nella "documentazione".

Uno snapshot dei volumi persistenti che eseguono il backup dei dischi verrà creato dal CSI. Viene creato un backup della macchina virtuale insieme all'istantanea dei relativi dischi e memorizzato nella posizione di backup specificata nel codice yaml. Il backup rimarrà nel sistema per 30 giorni come specificato nel ttl.

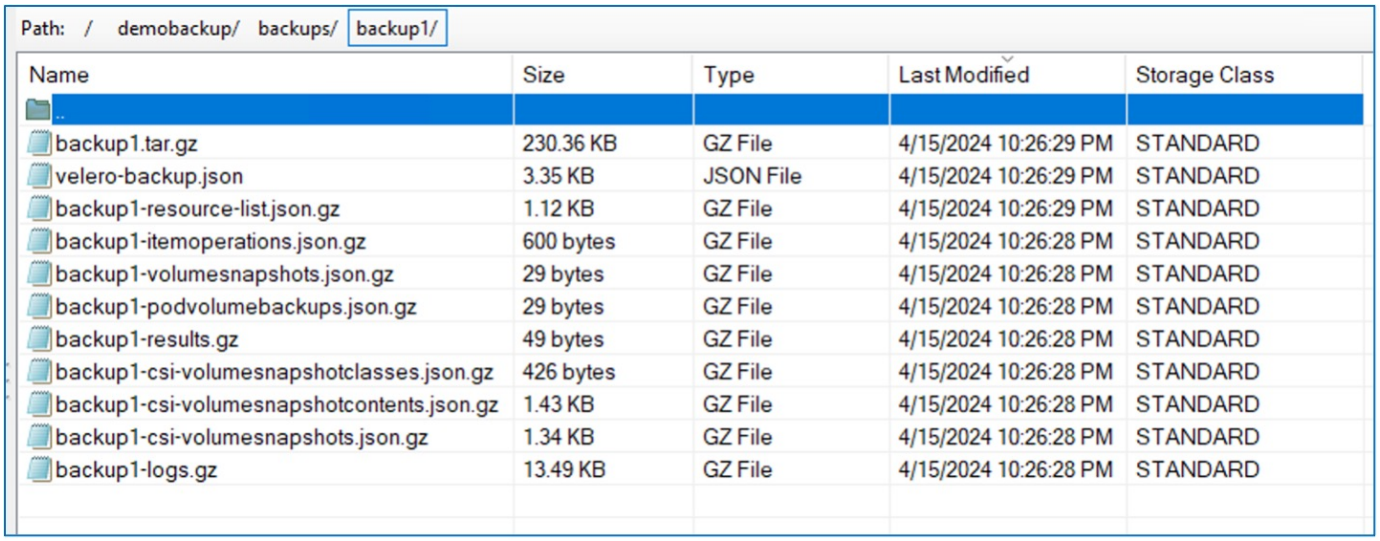
```
apiVersion: velero.io/v1
kind: Backup
metadata:
  name: backup1
  namespace: openshift-adp
spec:
  includedNamespaces:
  - virtual-machines-demo
  snapshotVolumes: true
  storageLocation: velero-demo-1 -->this is the backupStorageLocation
previously created
                                    when Velero is configured.
  ttl: 720h0m0s
```

Una volta completato il backup, la sua fase viene visualizzata come completata.



È possibile esaminare il backup nell'archiviazione a oggetti con l'aiuto di un'applicazione browser S3. Il percorso del backup viene visualizzato nel bucket configurato con il nome del prefisso (velero/demobackup). Il contenuto del backup include gli snapshot del volume, i log e altri metadati della macchina virtuale.

> In StorageGRID, è anche possibile utilizzare la console S3 disponibile in Gestione tenant per visualizzare gli oggetti di backup.



Name	Size	Type	Last Modified	Storage Class
..				
backup1.tar.gz	230.36 KB	GZ File	4/15/2024 10:26:29 PM	STANDARD
velero-backup.json	3.35 KB	JSON File	4/15/2024 10:26:29 PM	STANDARD
backup1-resource-list.json.gz	1.12 KB	GZ File	4/15/2024 10:26:29 PM	STANDARD
backup1-itemoperations.json.gz	600 bytes	GZ File	4/15/2024 10:26:28 PM	STANDARD
backup1-volumesnapshots.json.gz	29 bytes	GZ File	4/15/2024 10:26:28 PM	STANDARD
backup1-podvolumebackups.json.gz	29 bytes	GZ File	4/15/2024 10:26:28 PM	STANDARD
backup1-results.gz	49 bytes	GZ File	4/15/2024 10:26:28 PM	STANDARD
backup1-csi-volumesnapshotclasses.json.gz	426 bytes	GZ File	4/15/2024 10:26:28 PM	STANDARD
backup1-csi-volumesnapshotcontents.json.gz	1.43 KB	GZ File	4/15/2024 10:26:28 PM	STANDARD
backup1-csi-volumesnapshots.json.gz	1.34 KB	GZ File	4/15/2024 10:26:28 PM	STANDARD
backup1-logs.gz	13.49 KB	GZ File	4/15/2024 10:26:28 PM	STANDARD

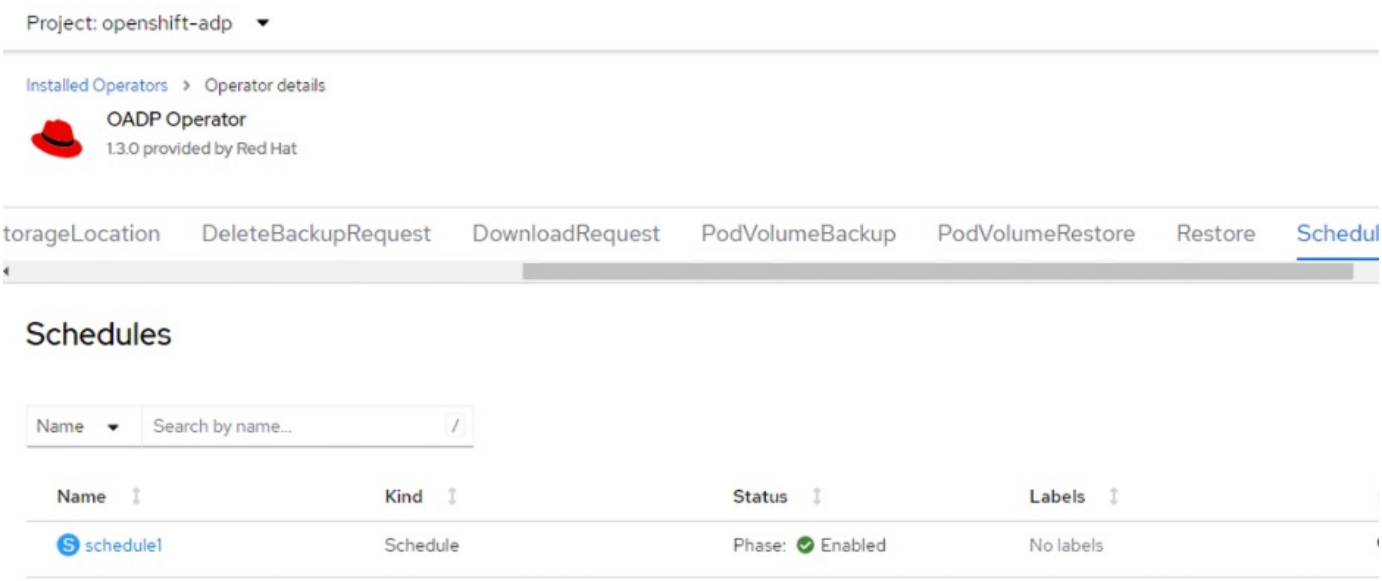
Creazione di backup pianificati per le VM in OpenShift Virtualization

Per creare backup in base a una pianificazione, è necessario creare una pianificazione CR.
La pianificazione è semplicemente un'espressione Cron che consente di specificare l'ora in cui si desidera creare il backup. Un esempio di yaml per creare una pianificazione CR.

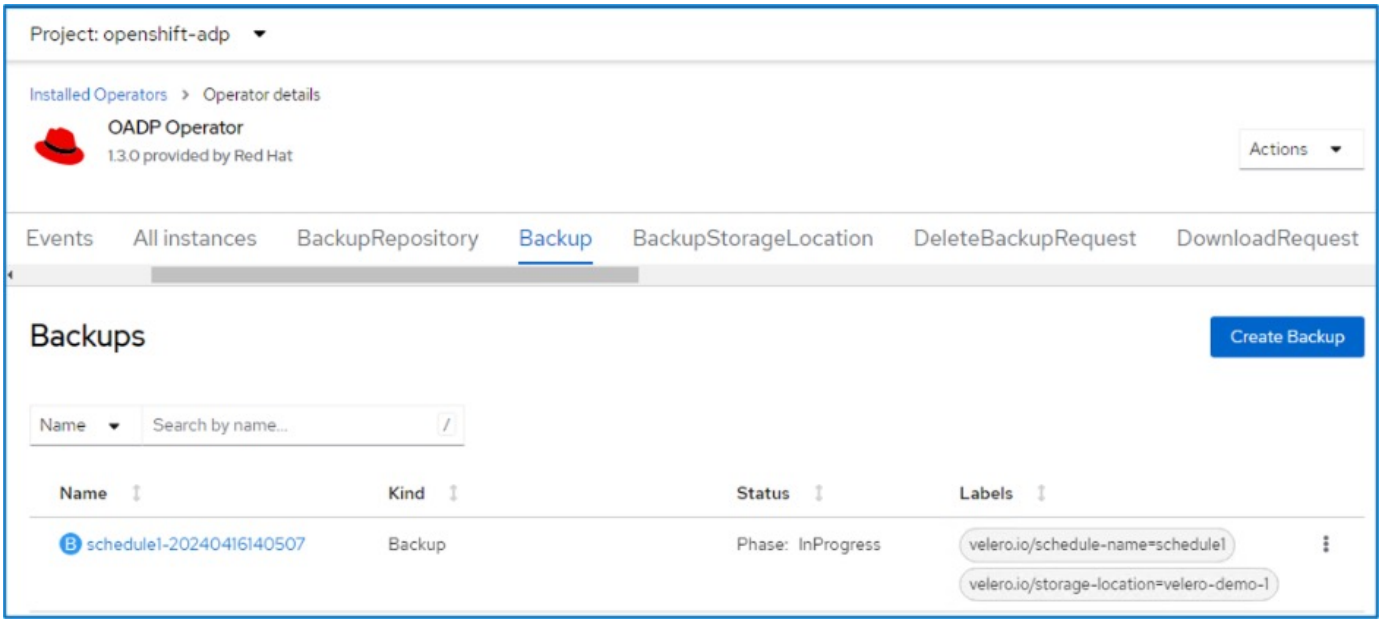
```
apiVersion: velero.io/v1
kind: Schedule
metadata:
  name: <schedule>
  namespace: openshift-adp
spec:
  schedule: 0 7 * * *
  template:
    hooks: {}
    includedNamespaces:
    - <namespace>
    storageLocation: velero-demo-1
    defaultVolumesToFsBackup: true
    ttl: 720h0m0s
```

Cron Expression 0 7 * * * significa che ogni giorno verrà creato un backup alle 7:00:00.
Vengono inoltre specificati gli spazi dei nomi da includere nel backup e la posizione di archiviazione per il backup. Quindi, invece di un CR di backup, il CR di pianificazione viene utilizzato per creare un backup all'ora e alla frequenza specificate.

Una volta creata, la pianificazione viene attivata.



I backup verranno creati in base a questa pianificazione e possono essere visualizzati dalla scheda Backup.



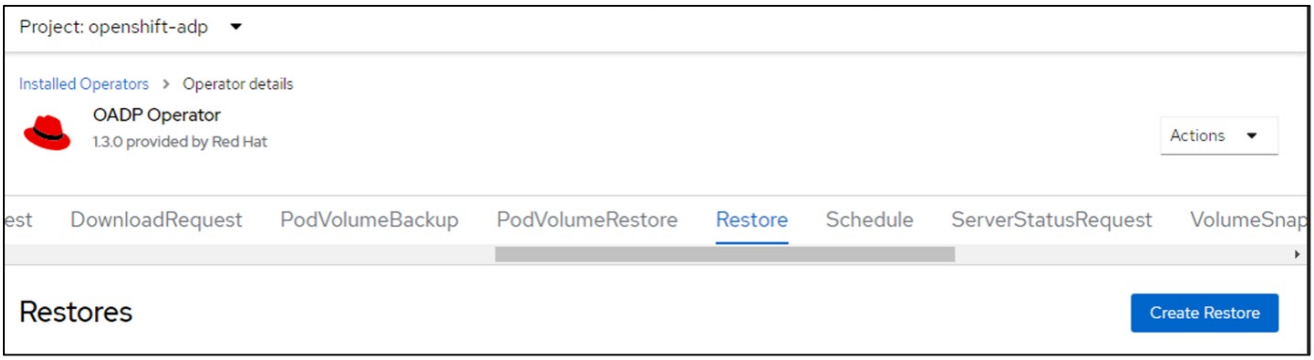
Ripristinare una VM da un backup

Prerequisiti

Per eseguire il ripristino da un backup, supponiamo che lo spazio dei nomi in cui esisteva la macchina virtuale sia stato eliminato accidentalmente.

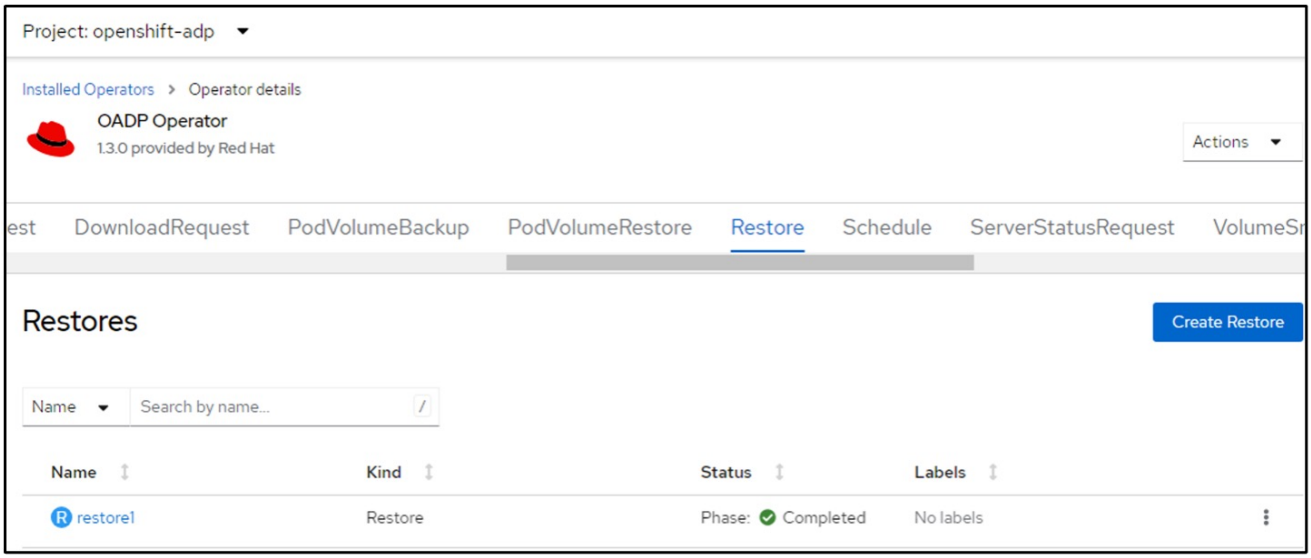
Ripristinare nello stesso namespace

Per eseguire il ripristino dal backup appena creato, è necessario creare una risorsa personalizzata di ripristino (CR). Dobbiamo fornirgli un nome, fornire il nome del backup da cui eseguire il ripristino e impostare su true. È possibile impostare parametri aggiuntivi come illustrato nella "documentazione". Fare clic sul pulsante Crea.




```
apiVersion: velero.io/v1
kind: Restore
metadata:
  name: restore1
  namespace: openshift-adp
spec:
  backupName: backup1
  restorePVs: true
```

Quando la fase è completata, è possibile vedere che le macchine virtuali sono state ripristinate allo stato in cui è stato acquisito lo snapshot. (Se il backup è stato creato quando la VM era in esecuzione, ripristinando la VM dal backup si avvia la VM ripristinata e la si porta in esecuzione). La VM viene ripristinata nello stesso namespace.



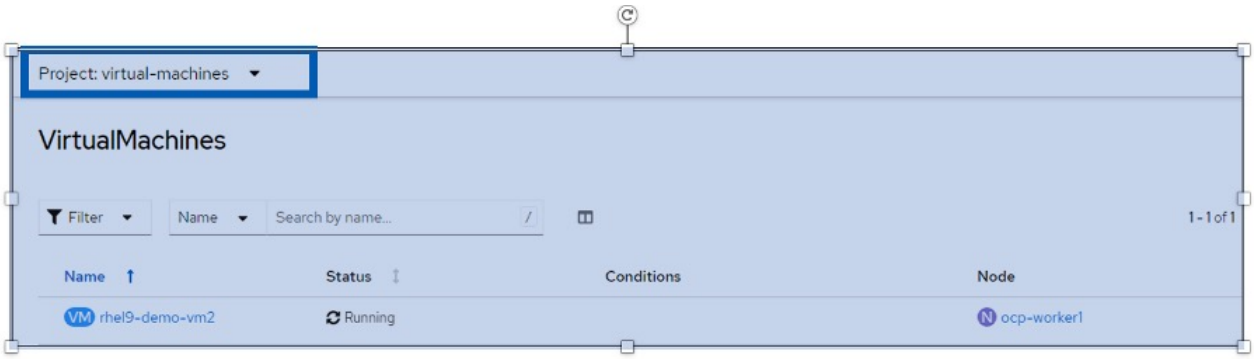
Ripristinare in un namespace diverso

Per ripristinare la macchina virtuale in uno spazio dei nomi diverso, è possibile fornire un namespaceMapping nella definizione yaml di Restore CR.

Il seguente file yaml di esempio crea un Restore CR per ripristinare una VM e i relativi dischi nello spazio dei nomi virtual-machine-demo quando il backup è stato eseguito nello spazio dei nomi virtual-machine.

```
apiVersion: velero.io/v1
kind: Restore
metadata:
  name: restore-to-different-ns
  namespace: openshift-adp
spec:
  backupName: backup
  restorePVs: true
  includedNamespaces:
  - virtual-machines-demo
  namespaceMapping:
    virtual-machines-demo: virtual-machines
```

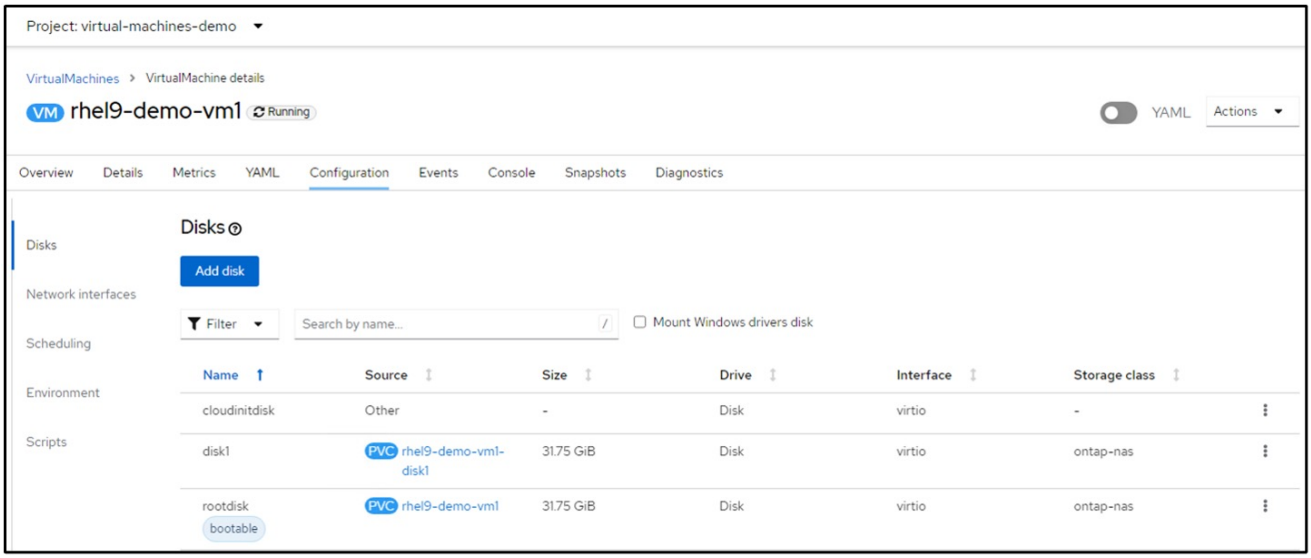
Quando la fase è completata, è possibile vedere che le macchine virtuali sono state ripristinate allo stato in cui è stato acquisito lo snapshot. (Se il backup è stato creato quando la VM era in esecuzione, ripristinando la VM dal backup si avvia la VM ripristinata e la si porta in esecuzione). La VM viene ripristinata in uno spazio dei nomi diverso, come specificato in yaml.

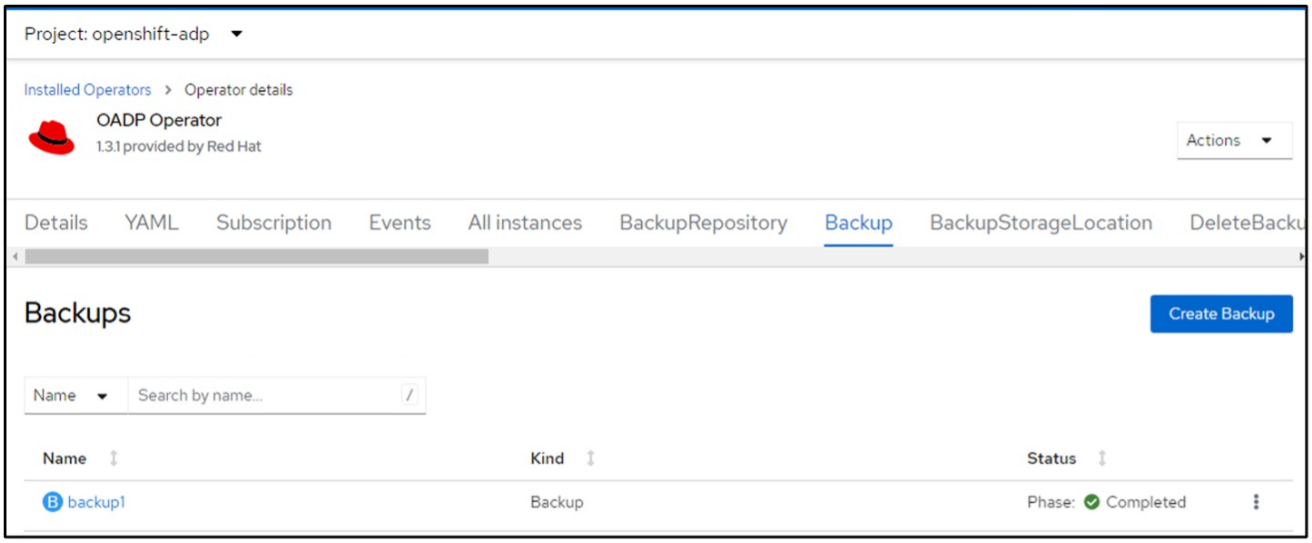


Ripristinare in una classe di archiviazione diversa

Velero fornisce una capacità generica di modificare le risorse durante il ripristino specificando le patch json. Le patch json vengono applicate alle risorse prima di essere ripristinate. Le patch json sono specificate in una configmap e la configmap è referenziata nel comando restore. Questa funzione consente di eseguire il ripristino utilizzando una classe di archiviazione diversa.

Nell'esempio seguente, la macchina virtuale, in fase di creazione, utilizza ontap-nas come classe di storage per i dischi. Viene creato un backup della macchina virtuale denominata Backup1.





Simula la perdita della macchina virtuale eliminando la macchina virtuale.

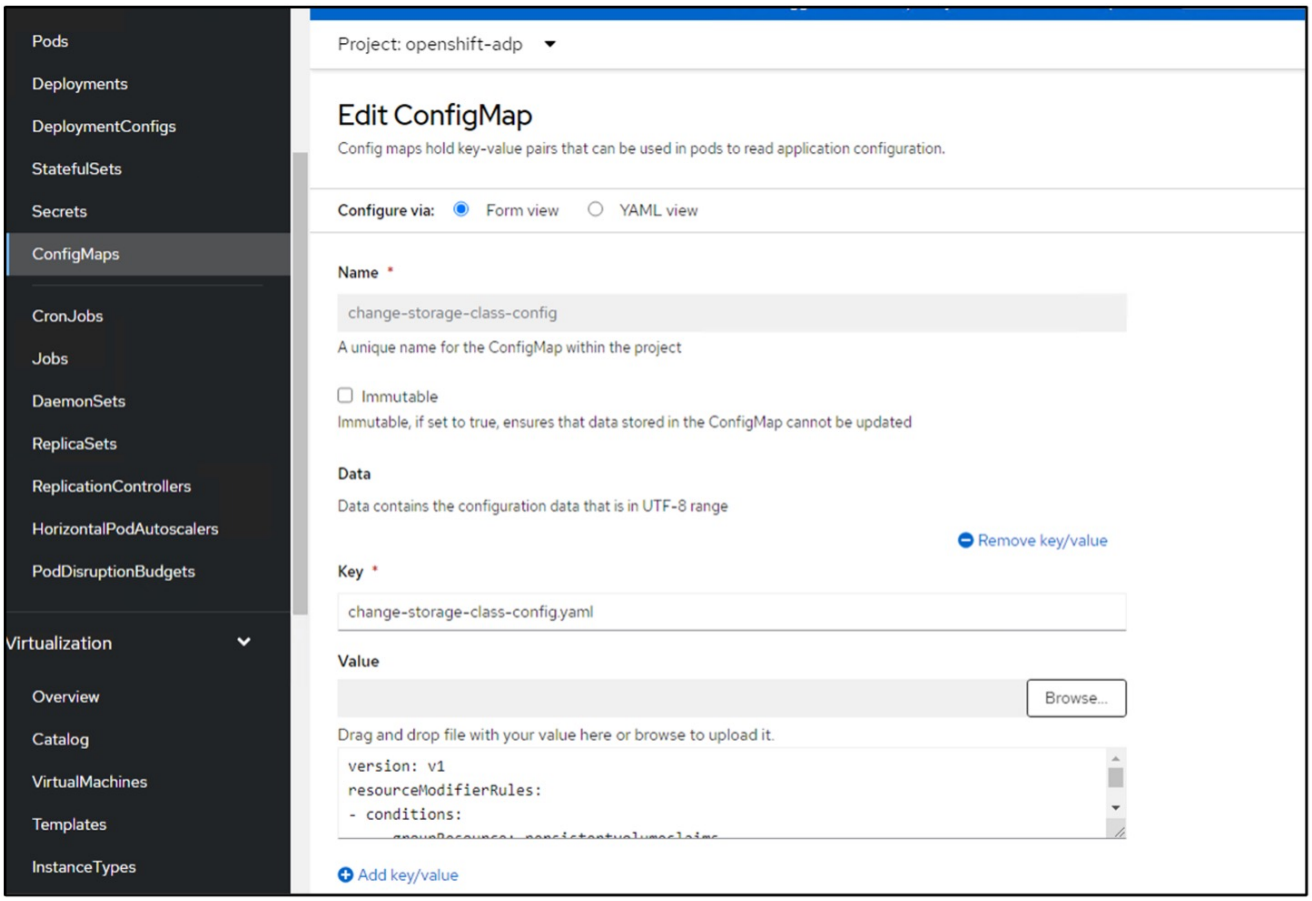
Per ripristinare la macchina virtuale utilizzando una classe di storage diversa, ad esempio ontap-nas-eco storage, devi effettuare i due seguenti passaggi:

Passo 1

Creare una mappa di configurazione (console) nello spazio dei nomi openshift-adp come segue:
Inserisci i dettagli come mostrato nella schermata:
Selezionare spazio dei nomi : openshift-adp
Nome: Change-storage-class-config (può essere qualsiasi nome)

Chiave: Change-storage-class-config.yaml:
Valore:

```
version: v1
    resourceModifierRules:
    - conditions:
        groupResource: persistentvolumeclaims
        resourceNameRegex: "^rhel*"
        namespaces:
        - virtual-machines-demo
      patches:
      - operation: replace
        path: "/spec/storageClassName"
        value: "ontap-nas-eco"
```



L'oggetto della mappa di configurazione risultante dovrebbe essere simile al seguente (CLI):

```
# kubectl describe cm/change-storage-class-config -n openshift-
adp
Name:         change-storage-class-config
Namespace:    openshift-adp
Labels:       velero.io/change-storage-class=RestoreItemAction
              velero.io/plugin-config=
Annotations:  <none>

Data
====
change-storage-class-config.yaml:
----
version: v1
resourceModifierRules:
- conditions:
     groupResource: persistentvolumeclaims
     resourceNameRegex: "^rhel*"
     namespaces:
     - virtual-machines-demo
  patches:
  - operation: replace
    path: "/spec/storageClassName"
    value: "ontap-nas-eco"



BinaryData
====

Events:  <none>
```

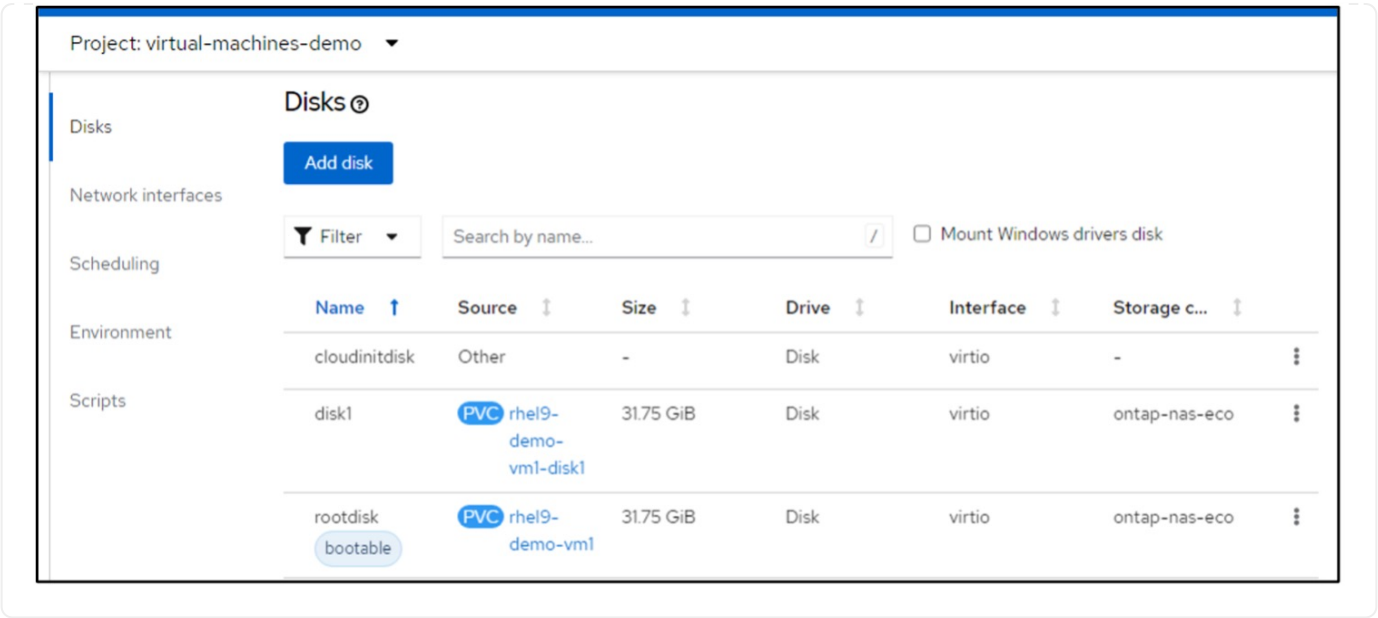
Questa mappa di configurazione applicherà la regola del modificatore di risorse quando viene creato il ripristino. Verrà applicata una patch per sostituire il nome della classe storage in ontap-nas-eco per tutte le richieste di volume persistenti a partire da rhel.

Passo 2

Per ripristinare la macchina virtuale, utilizzare il seguente comando dall'interfaccia CLI di Velero:

```
#velero restore create restore1 --from-backup backup1 --resource
-modifier-configmap change-storage-class-config -n openshift-adp
```

La macchina virtuale viene ripristinata con lo stesso namespace con i dischi creati utilizzando la classe storage ontap-nas-eco.



Eliminazione di backup e ripristini mediante Velero

Eliminazione di un backup

È possibile eliminare una CR di backup senza eliminare i dati di archiviazione oggetti utilizzando lo strumento CLI OC.

```
oc delete backup <backup_CR_name> -n <velero_namespace>
```

Se si desidera eliminare la CR di backup ed eliminare i dati di archiviazione degli oggetti associati, è possibile farlo utilizzando lo strumento CLI Velero.

Scaricare l'interfaccia CLI come indicato nelle istruzioni nella "Documentazione Velero".

Eseguire il seguente comando delete utilizzando l'interfaccia CLI di Velero

```
velero backup delete <backup_CR_name> -n <velero_namespace>
```

È inoltre possibile eliminare il ripristino CR utilizzando l'interfaccia CLI Velero

```
velero restore delete restore --namespace openshift-adp
```

È possibile utilizzare il comando oc e l'interfaccia utente per eliminare la CR di ripristino

```
oc delete backup <backup_CR_name> -n <velero_namespace>
```

Monitoraggio tramite Cloud Insights

Monitoraggio utilizzando Cloud Insights per le VM nella virtualizzazione Red Hat OpenShift

Autore: Banu Sundhar, NetApp

Questa sezione del documento di riferimento fornisce dettagli sull'integrazione di NetApp Cloud Insights con un cluster Red Hat OpenShift per il monitoraggio delle VM di virtualizzazione OpenShift.

NetApp Cloud Insights è uno strumento di monitoraggio dell'infrastruttura cloud che offre visibilità sull'intera infrastruttura. Con Cloud Insights, puoi monitorare, risolvere i problemi e ottimizzare tutte le risorse, inclusi i cloud pubblici e i data center privati. Per ulteriori informazioni su NetApp Cloud Insights, consultare la "Documentazione Cloud Insights".

Per iniziare a utilizzare Cloud Insights, devi iscriverti al portale NetApp BlueXP. Per ulteriori informazioni, fare riferimento a. "Assunzione di Cloud Insights"

Cloud Insights dispone di diverse funzionalità che ti consentono di trovare i dati in modo rapido e semplice, risolvere i problemi e fornire informazioni dettagliate sull'ambiente. È possibile trovare facilmente i dati con potenti query, visualizzare i dati nelle dashboard e inviare avvisi e-mail per le soglie di dati impostate. Fare riferimento a. "tutorial video" per facilitare la comprensione di queste funzioni.

Per avviare la raccolta dei dati da parte di Cloud Insights, è necessario disporre di quanto segue

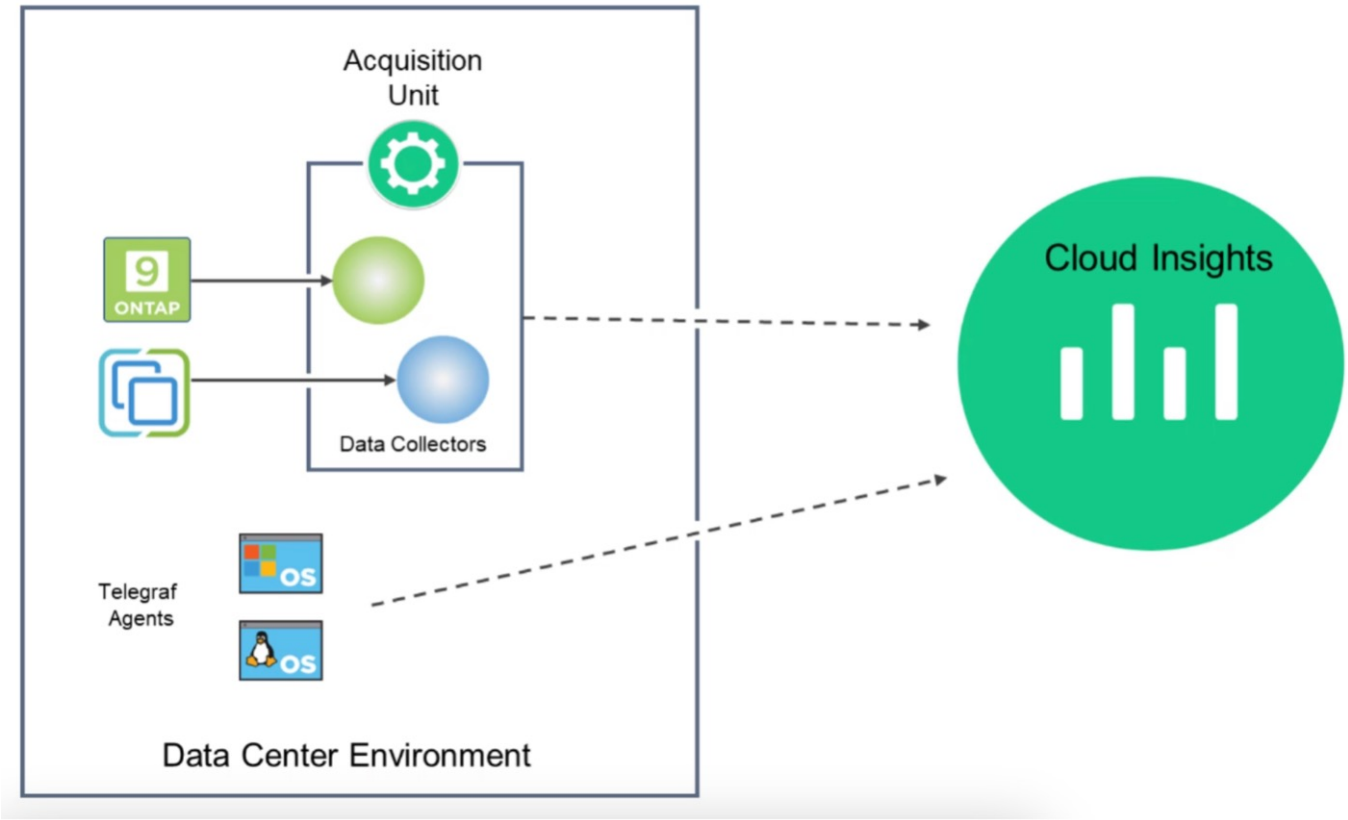
Data Collector
Esistono 3 tipi di Data Collector:
* Infrastruttura (dispositivi di storage, switch di rete, infrastruttura di elaborazione)
* Sistemi operativi (come VMware o Windows)
* Servizi (come Kafka)

I Data Collector rilevano le informazioni provenienti dalle origini dati, ad esempio il dispositivo di archiviazione ONTAP (raccoglitore dati infrastruttura). Le informazioni raccolte vengono utilizzate per l'analisi, la convalida, il monitoraggio e la risoluzione dei problemi.

Unità di acquisizione
Se si utilizza un servizio Data Collector di infrastruttura, è necessaria anche un'unità di acquisizione per inserire i dati in Cloud Insights. Un'unità di acquisizione è un computer dedicato all'hosting di raccoglitori di dati, in genere una macchina virtuale. In genere, questo computer si trova nello stesso centro dati/VPC degli elementi monitorati.

Agenti Telegraf
Cloud Insights supporta anche Telegraf come agente per la raccolta dei dati di integrazione. Telegraf è un agente server basato su plug-in che può essere utilizzato per raccogliere e generare report su metriche, eventi e registri.

Architettura Cloud Insights



Integrazione con Cloud Insights per VM nella virtualizzazione Red Hat OpenShift

Per iniziare a raccogliere dati per le VM in OpenShift Virtualization è necessario installare:

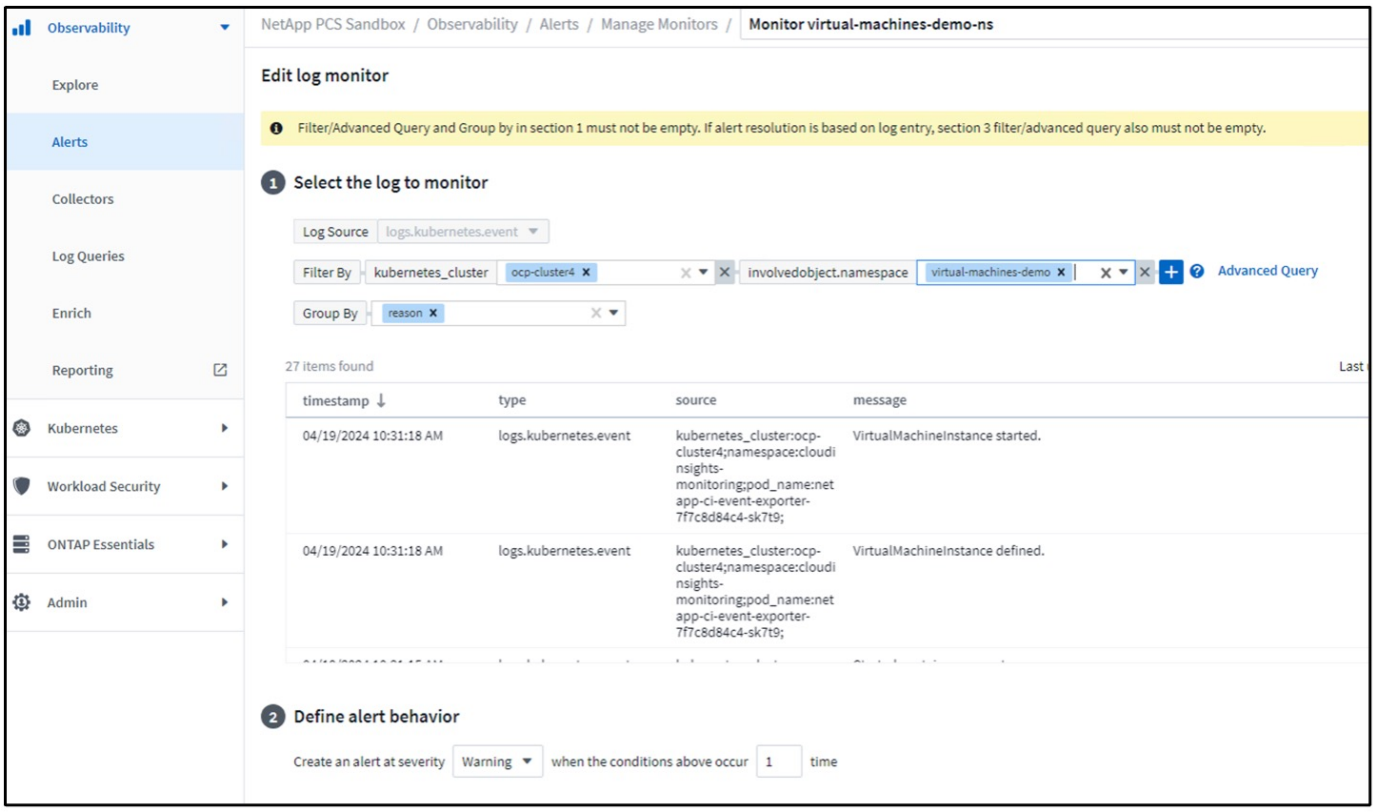
1. Un operatore di monitoring e un data collector Kubernetes per raccogliere i dati Kubernetes
   Per istruzioni complete, fare riferimento a. "documentazione".
2. Un'unità di acquisizione per raccogliere dati dallo storage ONTAP che fornisce storage persistente per i dischi delle macchine virtuali
   Per istruzioni complete, fare riferimento a. "documentazione".
3. Un data collector per ONTAP
   Per istruzioni complete, fare riferimento a. "documentazione"

Inoltre, se si utilizza StorageGRID per i backup delle VM, è necessario disporre di un data collector anche per StorageGRID.

Esempio di funzionalità di monitoraggio per le VM in Red Hat OpenShift Virtualization

Monitoraggio basato su eventi e creazione di avvisi

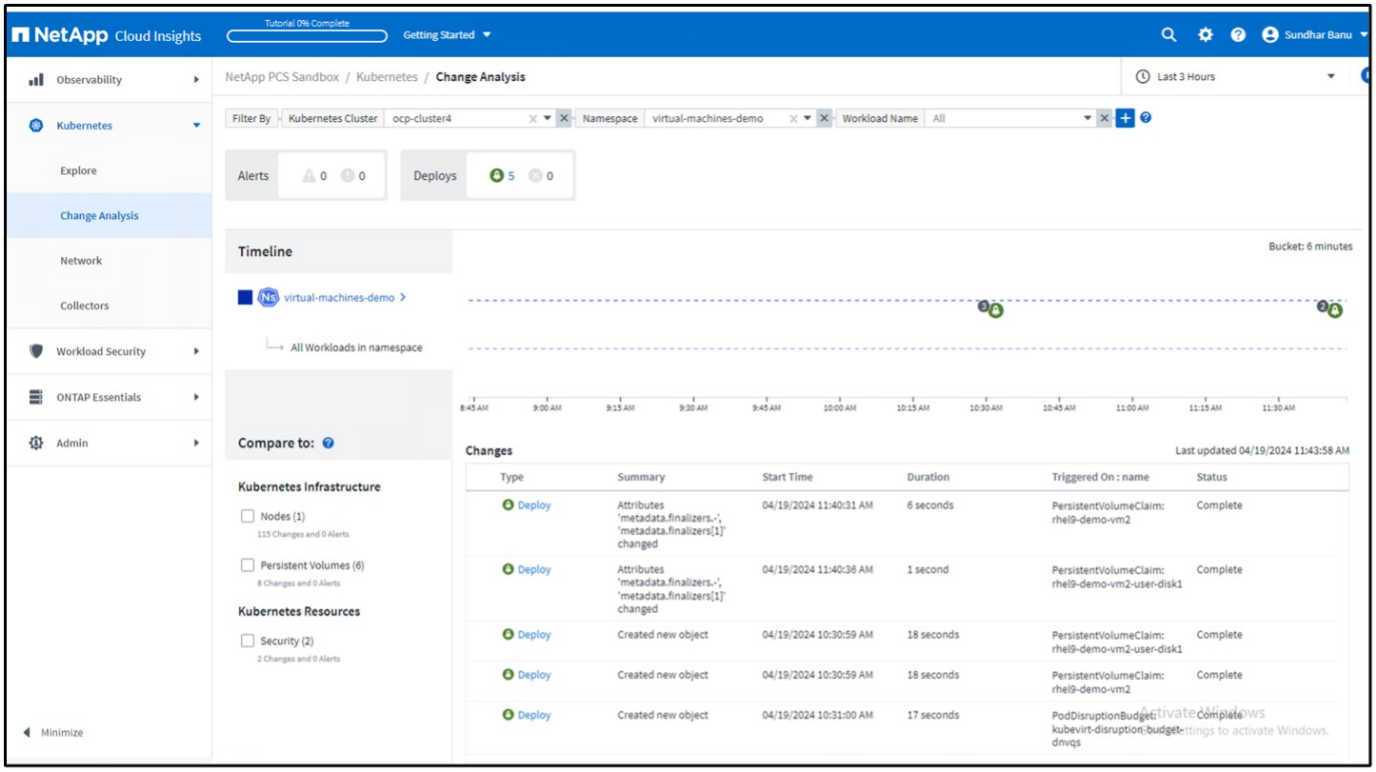
Di seguito viene riportato un esempio in cui lo spazio dei nomi che contiene una VM in OpenShift Virtualization viene monitorato in base agli eventi. In questo esempio, viene creato un monitor in base all'evento **logs.kuPand** per lo spazio dei nomi specificato nel cluster.



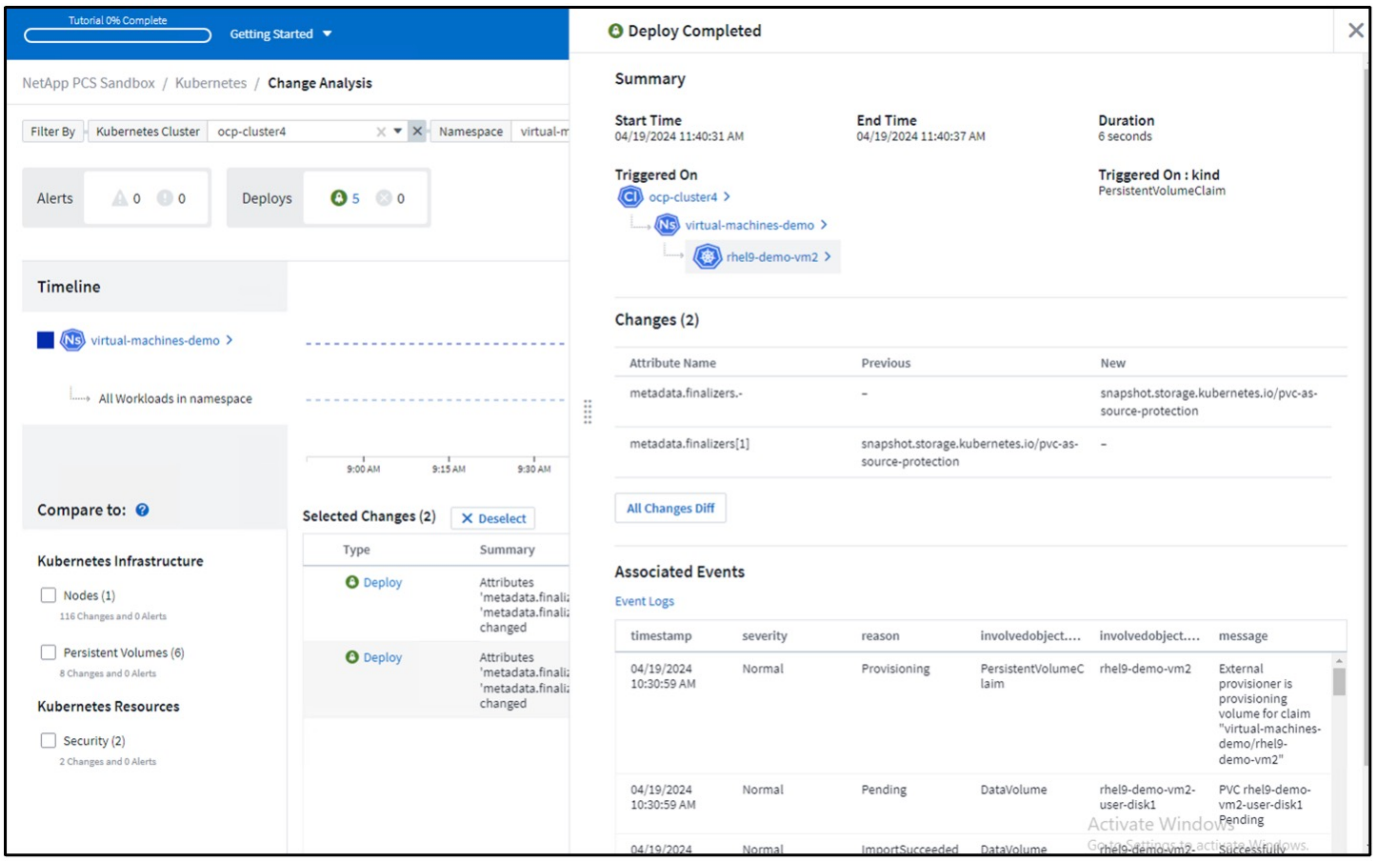
Questa query fornisce tutti gli eventi per la macchina virtuale nello spazio dei nomi. (Esiste una sola macchina virtuale nello spazio dei nomi). Una query avanzata può anche essere costruita per filtrare in base all'evento in cui il motivo è "fallito" o "non riuscito". Questi eventi vengono in genere creati quando si verifica un problema nella creazione di un PV o nel montaggio del PV su un pod che indica problemi nel provisioner dinamico per la creazione di dati persistenti dei volumi per la macchina virtuale.
Durante la creazione del monitoraggio degli avvisi, come illustrato sopra, è anche possibile configurare la notifica ai destinatari. È inoltre possibile fornire azioni correttive o informazioni aggiuntive utili per risolvere l'errore. Nell'esempio precedente, informazioni aggiuntive potrebbero essere la ricerca della configurazione di back-end Trident e delle definizioni delle classi di storage per la risoluzione del problema.

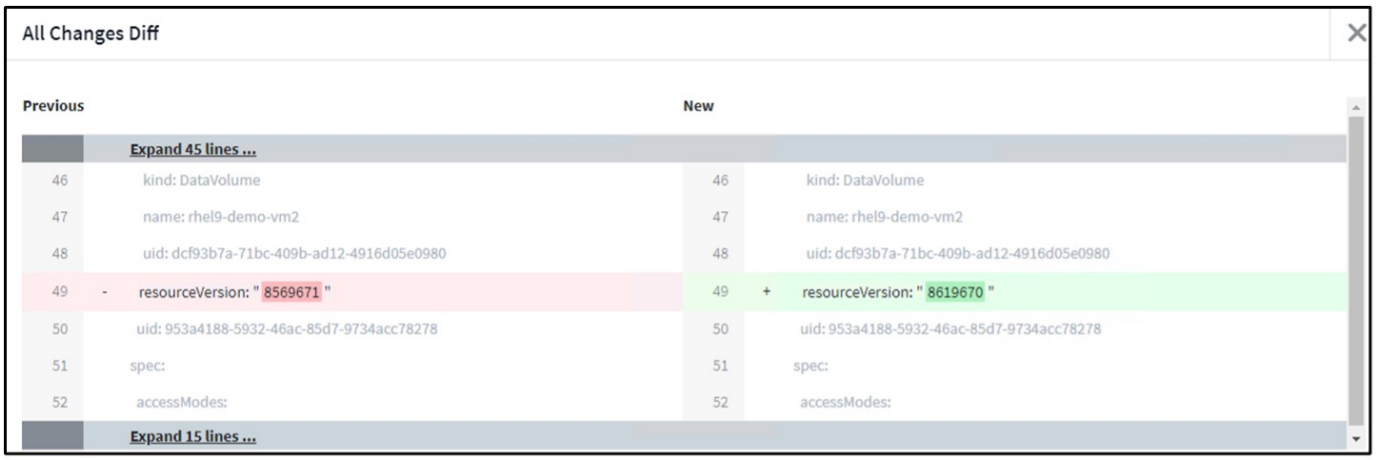
Cambia analisi

Con Change Analytics, è possibile ottenere una panoramica delle modifiche apportate allo stato del cluster, incluse le persone che hanno apportato tale modifica, che possono contribuire alla risoluzione dei problemi.



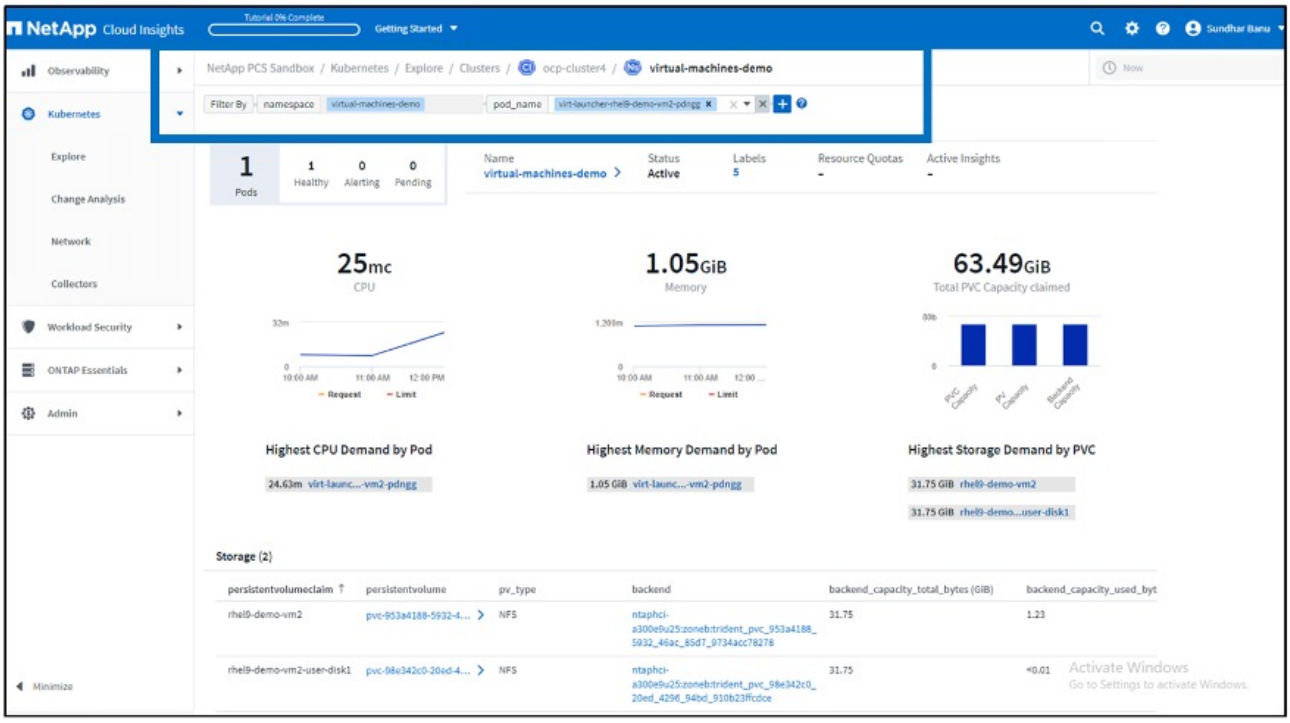
Nell'esempio precedente, Change Analysis è configurato sul cluster OpenShift per lo spazio dei nomi che contiene una VM di virtualizzazione OpenShift. Il dashboard mostra le modifiche rispetto alla timeline. Si può drill-down per vedere cosa è cambiato e fare clic su tutte le modifiche Diff per vedere la diff dei manifesti. Dal manifesto, è possibile vedere che è stato creato un nuovo backup dei dischi permanenti.



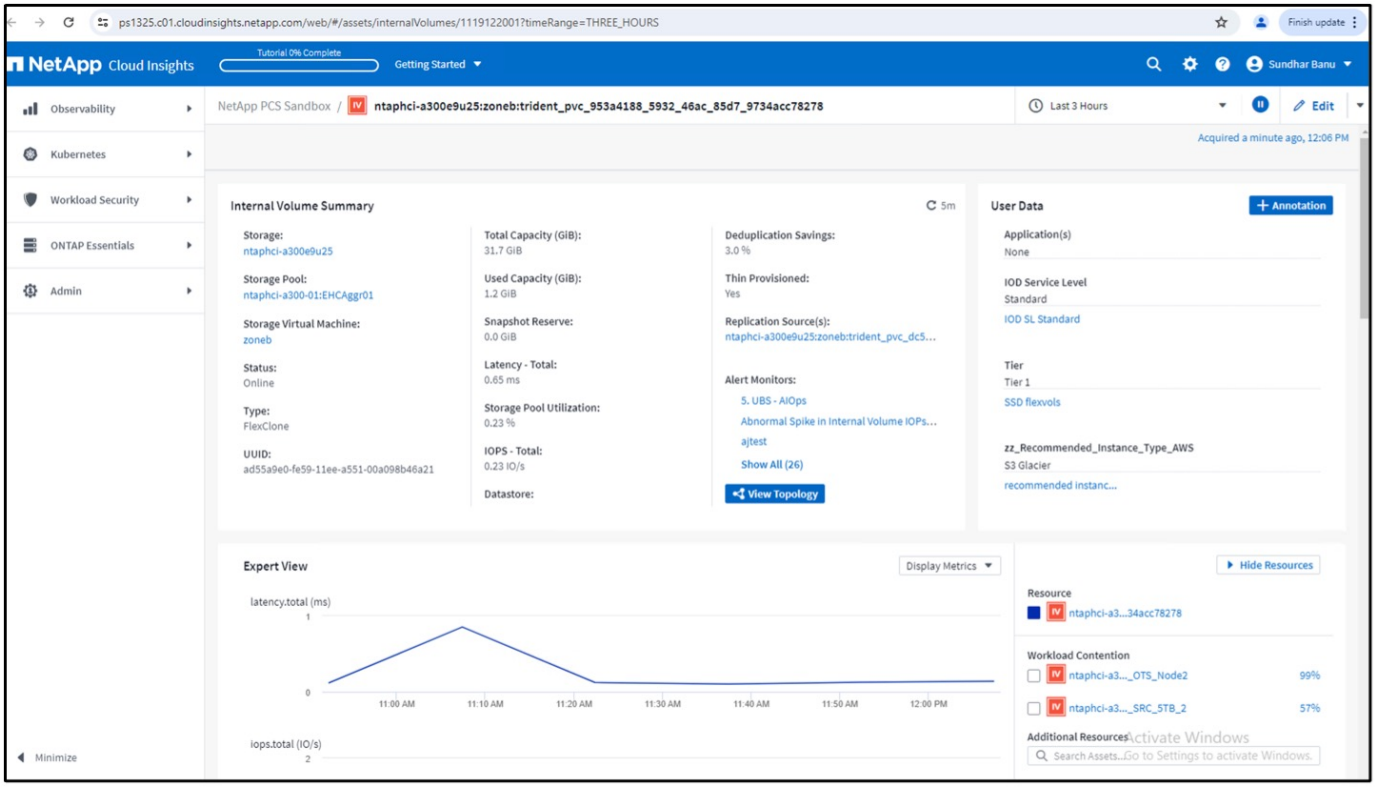


Mappatura archiviazione backend

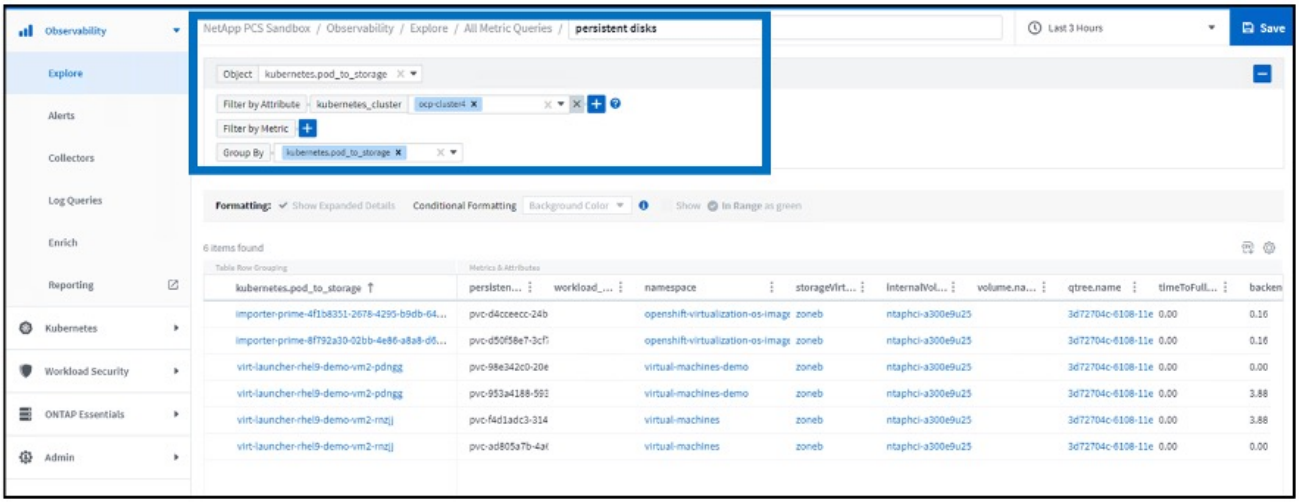
Con Cloud Insights, è possibile vedere facilmente lo storage backend dei dischi della macchina virtuale e le diverse statistiche sui PVC.



È possibile fare clic sui link presenti nella colonna backend per estrarre i dati direttamente dallo storage ONTAP back-end.

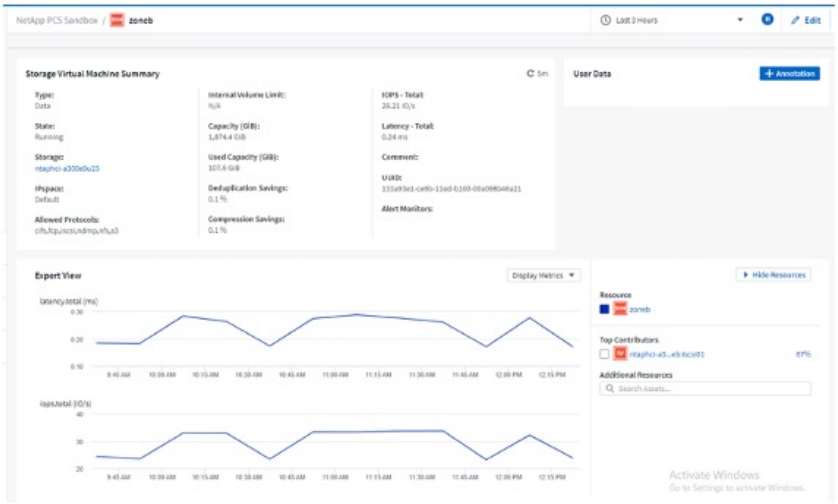


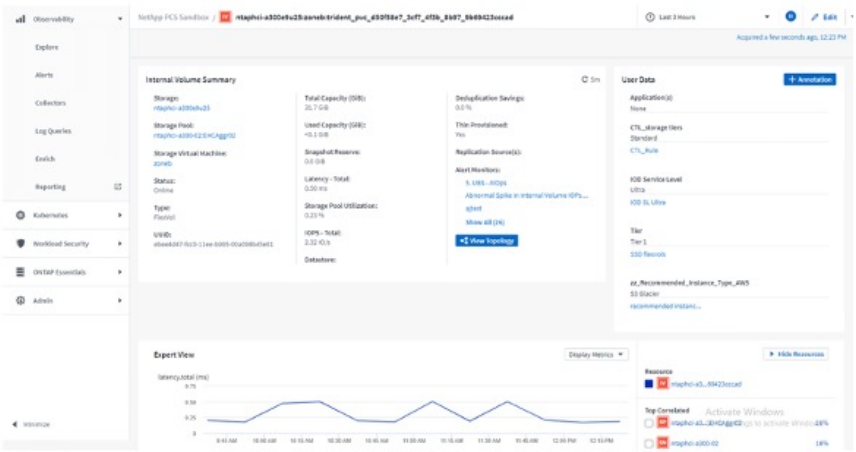
Un altro modo per esaminare tutte le mappature pod-storage è creare una query All Metrics dal menu Observability (osservabilità) in Explore (Esplora).



Facendo clic su uno dei collegamenti si otterranno i dettagli corrispondenti dall'archivio ONTP. Ad esempio, facendo clic sul nome di una SVM nella colonna storageVirtualMachine verranno estratti i dettagli relativi alla SVM da ONTAP. Facendo clic sul nome di un volume interno vengono visualizzati i dettagli relativi al volume in ONTAP.

	storageVirtualMachin...	internalVolume.name	volume.na..
zation-os-image	zoneb	ntaphci-a300e9u25:zoneb:trident_p	
zation-os-image	zoneb	ntaphci-a300e9u25:zoneb:trident_p	
demo	zoneb	ntaphci-a300e9u25:zoneb:trident_p	
demo	zoneb	ntaphci-a300e9u25:zoneb:trident_p	
	zoneb	ntaphci-a300e9u25:zoneb:trident_p	
	zoneb	ntaphci-a300e9u25:zoneb:trident_p	





Gestione avanzata dei cluster per Kubernetes su Red Hat OpenShift con NetApp

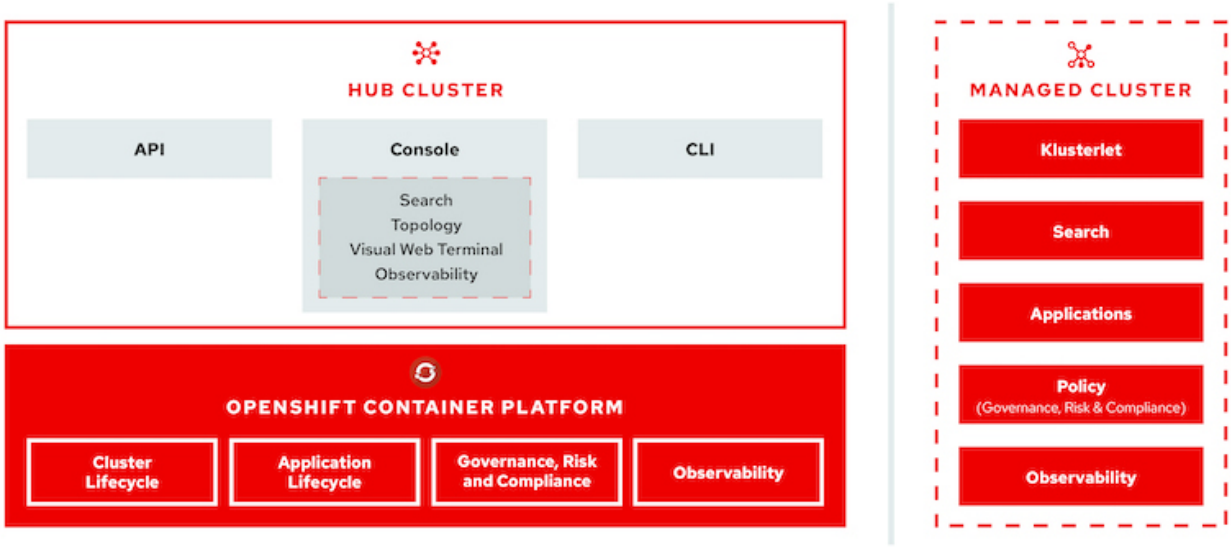
Gestione avanzata dei cluster per Kubernetes: Red Hat OpenShift con NetApp

Poiché un'applicazione containerizzata passa dallo sviluppo alla produzione, molte organizzazioni richiedono più cluster Red Hat OpenShift per supportare il test e l'implementazione di tale applicazione. In combinazione con questo, le organizzazioni generalmente ospitano più applicazioni o carichi di lavoro su cluster OpenShift. Pertanto, ogni organizzazione finisce per gestire un insieme di cluster e gli amministratori di OpenShift devono quindi affrontare la sfida aggiunta di gestire e mantenere più cluster in una gamma di ambienti che si estendono a più data center on-premise e cloud pubblici. Per affrontare queste sfide, Red Hat ha introdotto la gestione avanzata dei cluster per Kubernetes.

Red Hat Advanced Cluster Management per Kubernetes consente di eseguire le seguenti operazioni:

1. Crea, importa e gestisci più cluster tra data center e cloud pubblici
2. Implementa e gestisci applicazioni o carichi di lavoro su più cluster da una singola console
3. Monitorare e analizzare lo stato e lo stato delle diverse risorse del cluster
4. Monitorare e applicare la conformità alla sicurezza in più cluster

Red Hat Advanced Cluster Management per Kubernetes viene installato come add-on in un cluster Red Hat OpenShift e utilizza questo cluster come controller centrale per tutte le operazioni. Questo cluster è noto come cluster di hub ed espone un piano di gestione per consentire agli utenti di connettersi a Advanced Cluster Management. Tutti gli altri cluster OpenShift importati o creati tramite la console Advanced Cluster Management sono gestiti dal cluster hub e sono denominati cluster gestiti. Installa un agente chiamato Klusterlet sui cluster gestiti per connetterli al cluster hub e soddisfare le richieste di attività diverse correlate alla gestione del ciclo di vita del cluster, alla gestione del ciclo di vita delle applicazioni, all'osservabilità e alla conformità alla sicurezza.



Per ulteriori informazioni, consultare la documentazione "qui".

Implementazione

Implementare Advanced Cluster Management per Kubernetes

Prerequisiti

1. Un cluster Red Hat OpenShift (superiore alla versione 4.5) per il cluster hub
2. Cluster Red Hat OpenShift (superiori alla versione 4.4.3) per cluster gestiti
3. Accesso cluster-admin al cluster Red Hat OpenShift
4. Un abbonamento Red Hat per Advanced Cluster Management per Kubernetes

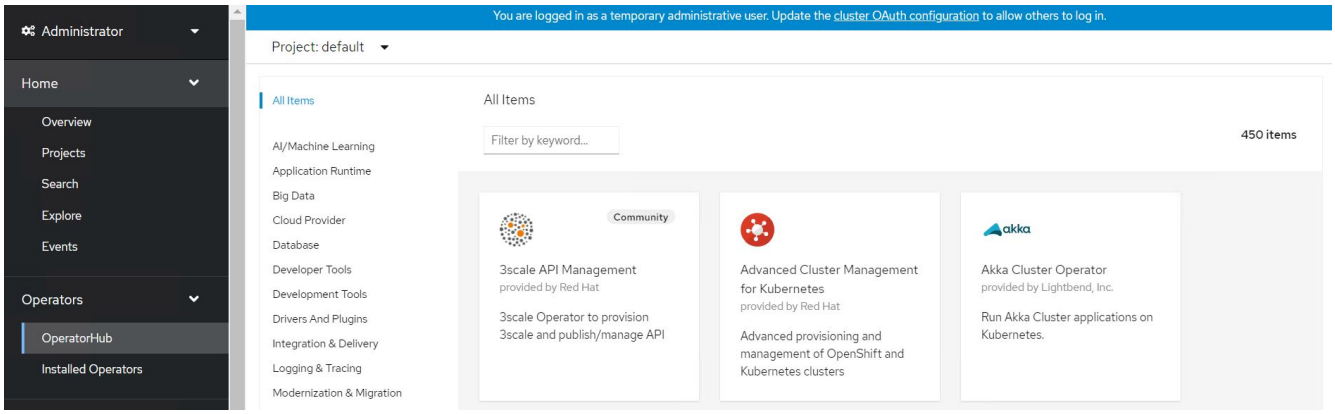
Advanced Cluster Management è un add-on per il cluster OpenShift, pertanto esistono determinati requisiti e restrizioni sulle risorse hardware in base alle funzionalità utilizzate nell'hub e nei cluster gestiti. È necessario tenere conto di questi problemi durante il dimensionamento dei cluster. Consultare la documentazione "qui" per ulteriori dettagli.

Se il cluster hub dispone di nodi dedicati per l'hosting dei componenti dell'infrastruttura e si desidera installare risorse di Advanced Cluster Management solo su tali nodi, è necessario aggiungere di conseguenza tolleranze e selettori a tali nodi. Per ulteriori informazioni, consultare la documentazione "qui".

Implementare Advanced Cluster Management per Kubernetes

Per installare Advanced Cluster Management per Kubernetes su un cluster OpenShift, attenersi alla seguente procedura:

1. Scegliere un cluster OpenShift come cluster hub e accedervi con privilegi di amministratore del cluster.
2. Accedere a Operators > Operators Hub e cercare Advanced Cluster Management for Kubernetes.



3. Selezionare Advanced Cluster Management for Kubernetes (Gestione avanzata cluster per Kubernetes) e fare clic su Install (Installa).

4. Nella schermata Install Operator (operatore di installazione), fornire i dettagli necessari (NetApp consiglia di conservare i parametri predefiniti) e fare clic su Install (Installa).

OperatorHub > Operator Installation

Install Operator

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

Update channel *

- release-2.0
- release-2.1
- release-2.2

Installation mode *

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

Installed Namespace *

- Operator recommended Namespace: PR open-cluster-management

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

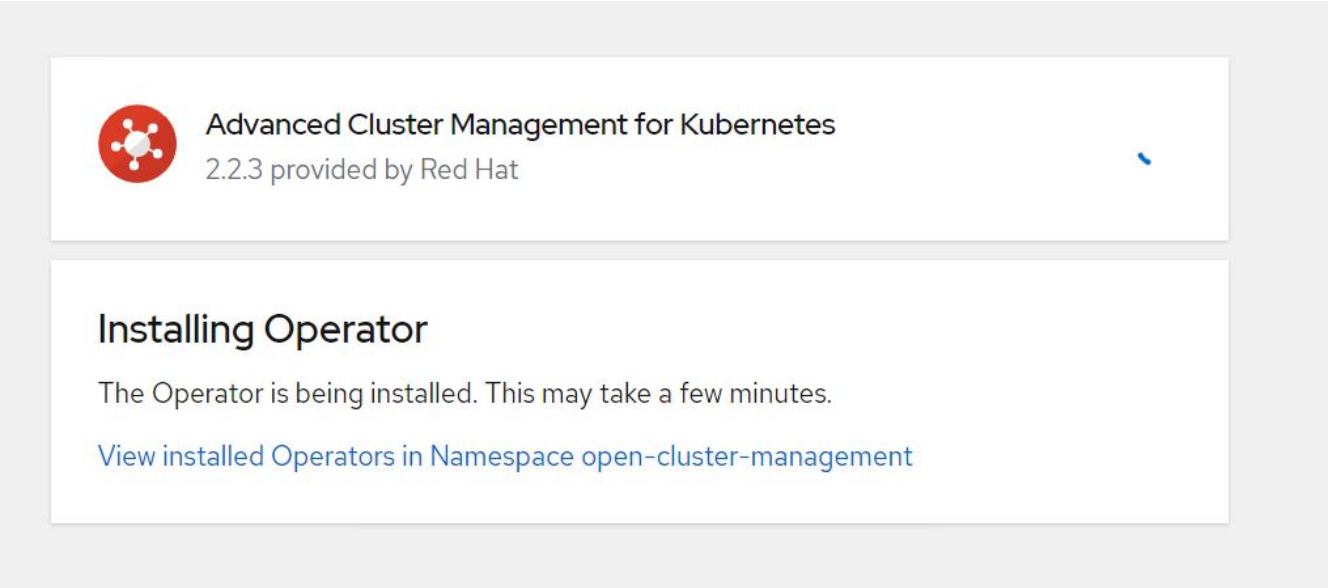
- Select a Namespace

Approval strategy *

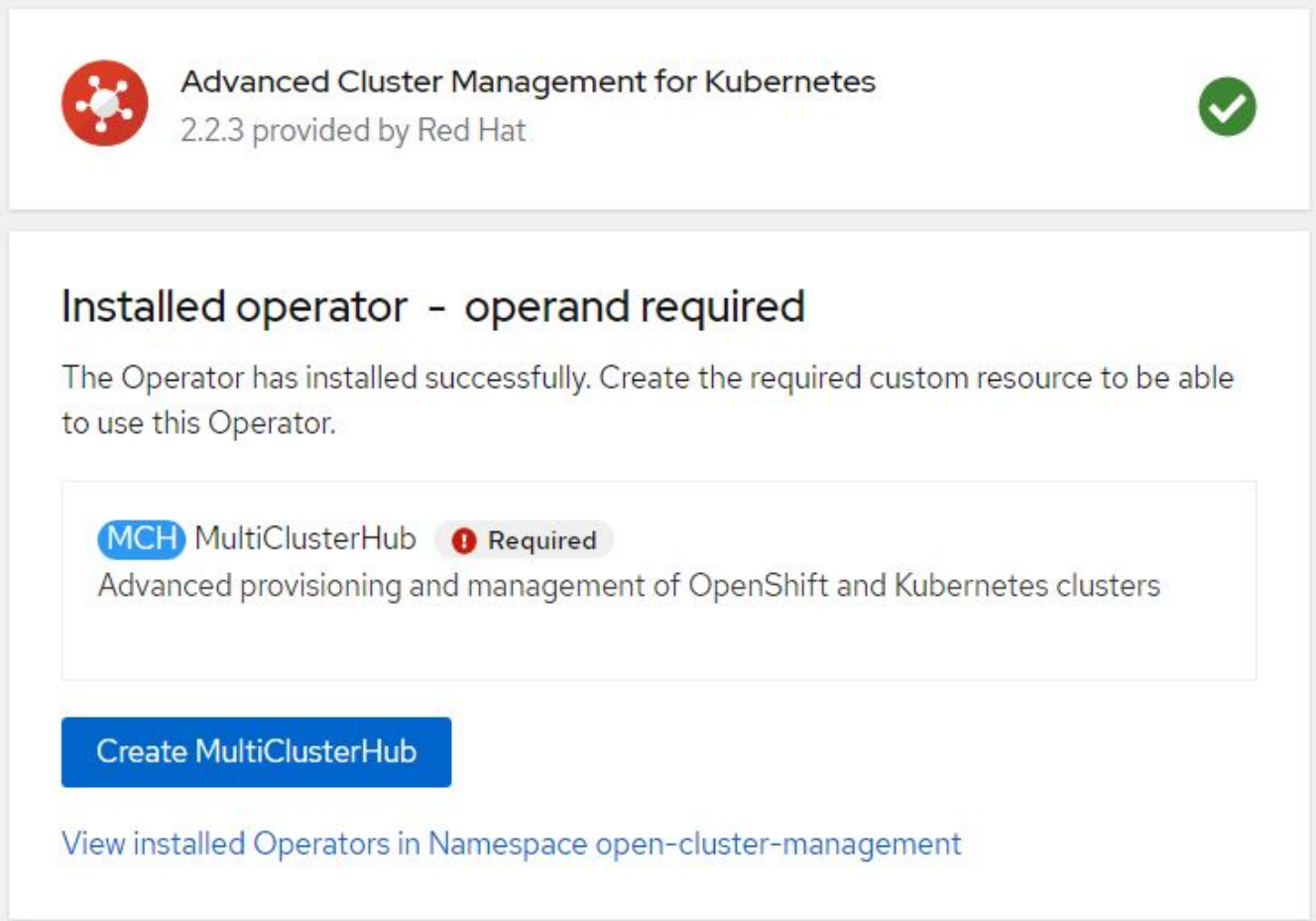
- Automatic
- Manual

Install Cancel

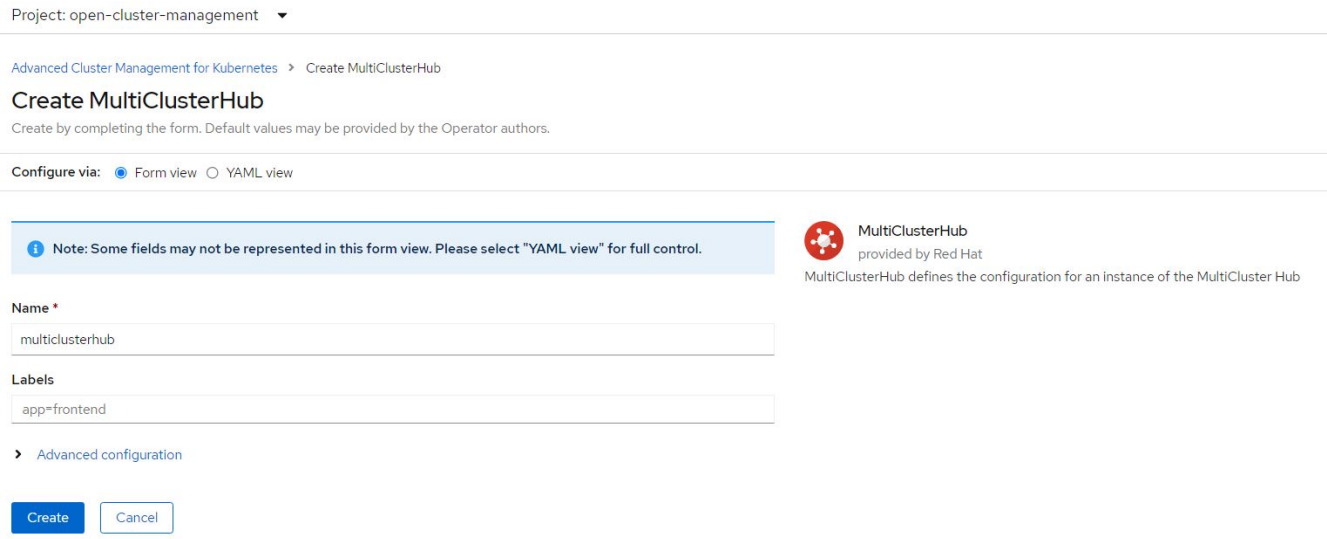
5. Attendere il completamento dell'installazione da parte dell'operatore.



6. Una volta installato l'operatore, fare clic su Create MultiClusterHub (Crea MultiClusterHub).



7. Nella schermata Create MultiClusterHub (Crea MultiClusterHub), fare clic su Create (Crea) dopo aver inserito i dettagli. In questo modo viene avviata l'installazione di un hub multi-cluster.



8. Dopo che tutti i pod sono stati spostati nello stato in esecuzione nello spazio dei nomi di gestione del cluster aperto e l'operatore passa allo stato riuscito, viene installata la funzione Advanced Cluster Management per Kubernetes.

Project: open-cluster-management

Installed Operators

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

Name | Search by name...

Name	Managed Namespaces	Status	Provided APIs
Advanced Cluster Management for Kubernetes 2.2.3 provided by Red Hat	NS open-cluster-management	Succeeded Up to date	MultiClusterHub ClusterManager ClusterDeployment ClusterState View 25 more...

9. Il completamento dell'installazione dell'hub richiede un po' di tempo e, una volta completata, l'hub MultiCluster passa allo stato di esecuzione.

Installed Operators > Operator details

Advanced Cluster Management for Kubernetes
2.2.3 provided by Red Hat

Actions

Details | YAML | Subscription | Events | All instances | MultiClusterHub | ClusterManager | ClusterDeployment | ClusterSt

MultiClusterHubs

Create MultiClusterHub

Name | Search by name...

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

10. Crea un percorso nello spazio dei nomi di gestione del cluster aperto. Connettersi all'URL nel percorso per accedere alla console Advanced Cluster Management.

Project: open-cluster-management

Routes

Create Route

Filter | Name | mul

Name mul ✕ Clear all filters

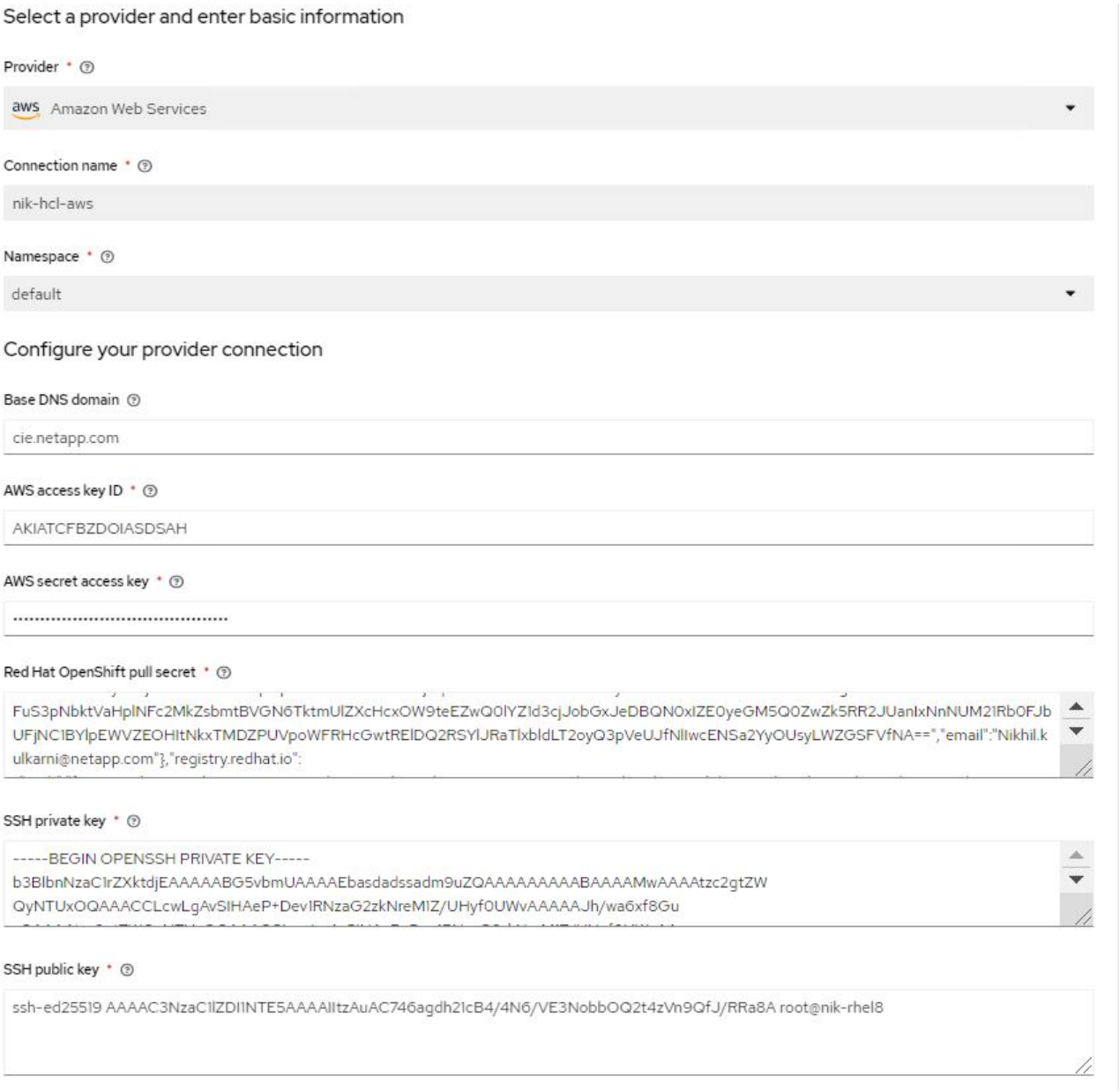
Name	Status	Location	Service
RT multicloud-console	Accepted	https://multicloud-console.apps.ocp-vmware2.cie.netapp.com	S management-ingress

Funzionalità: Gestione avanzata dei cluster per Kubernetes su Red Hat OpenShift con NetApp

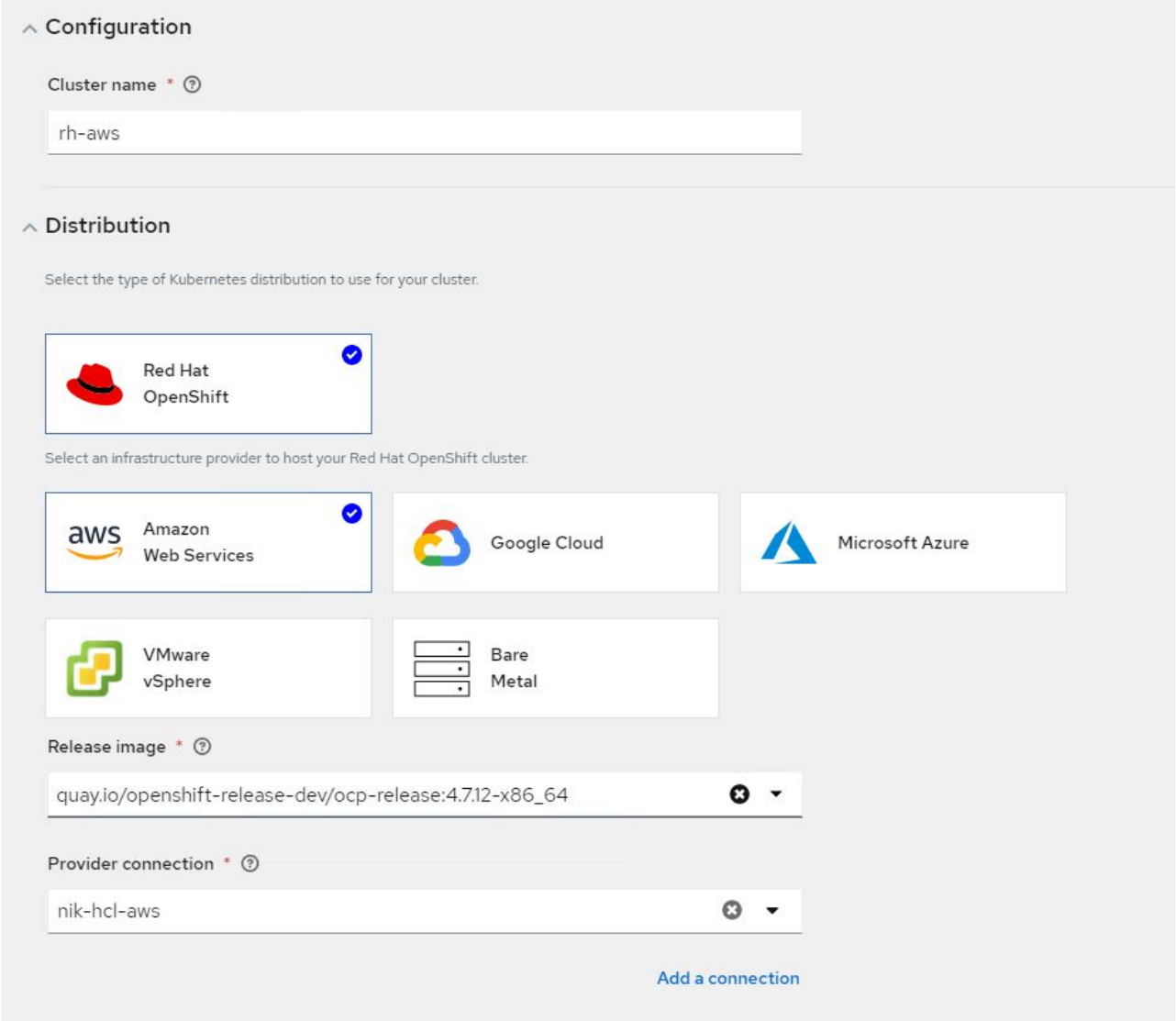
Gestione del ciclo di vita del cluster

Per gestire diversi cluster OpenShift, è possibile crearli o importarli in Advanced Cluster Management.

1. Prima di tutto, automatizza le infrastrutture > Clusters.
2. Per creare un nuovo cluster OpenShift, attenersi alla seguente procedura:
   a. Creare una connessione al provider: Accedere a connessioni provider e fare clic su Aggiungi una connessione, fornire tutti i dettagli corrispondenti al tipo di provider selezionato e fare clic su Aggiungi.

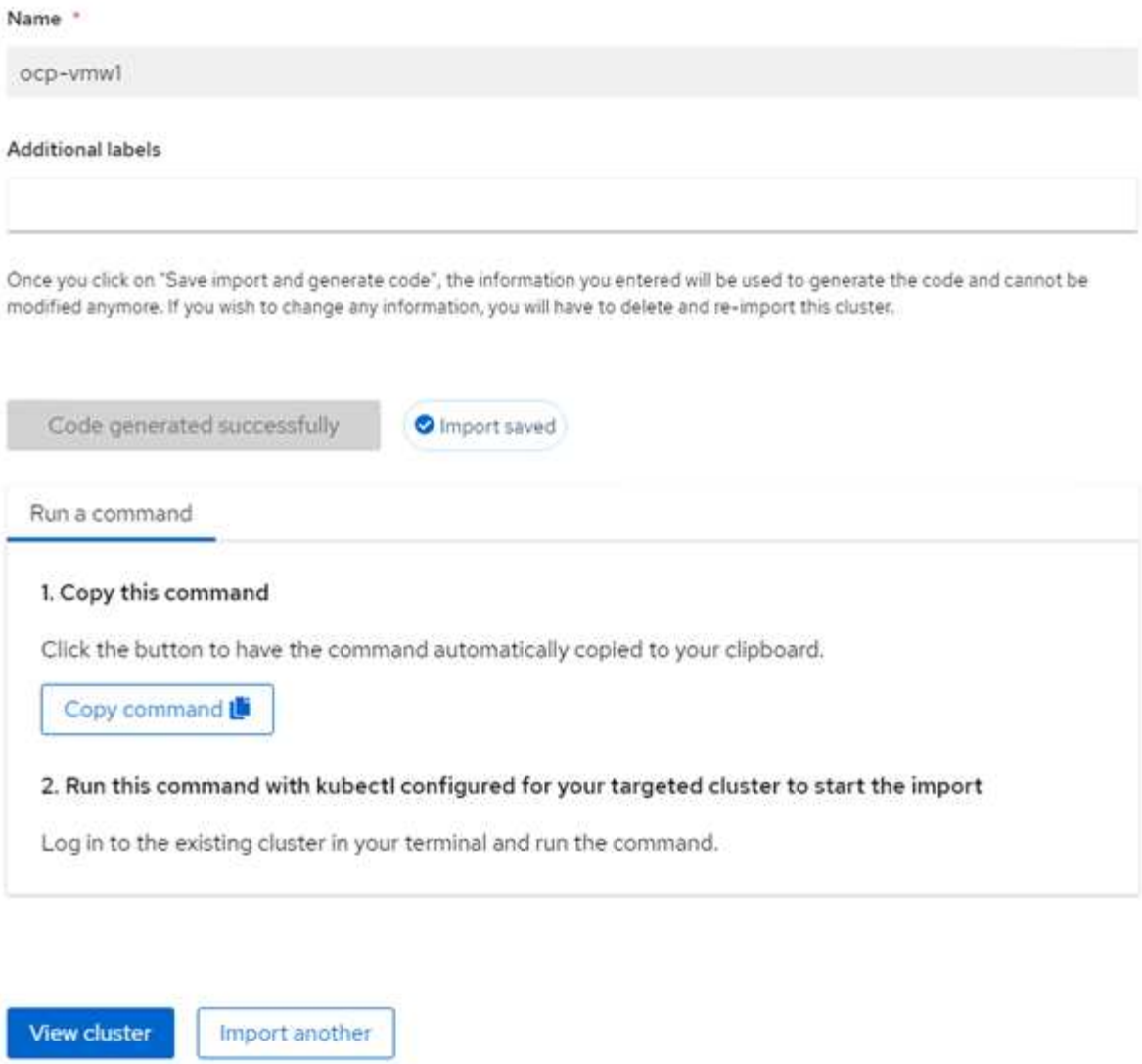


   b. Per creare un nuovo cluster, accedere a Clusters e fare clic su Add a Cluster (Aggiungi cluster) > Create a Cluster (Crea cluster). Fornire i dettagli del cluster e del provider corrispondente, quindi fare clic su Create (Crea).



c. Una volta creato, il cluster viene visualizzato nell'elenco dei cluster con lo stato Ready (Pronto).

3. Per importare un cluster esistente, attenersi alla seguente procedura:
   a. Accedere a Clusters e fare clic su Add a Cluster (Aggiungi cluster) > Import an Existing Cluster (Importa cluster esistente).
   b. Inserire il nome del cluster e fare clic su Save Import and generate Code (Salva importazione e genera codice). Viene visualizzato un comando per aggiungere il cluster esistente.
   c. Fare clic su Copy Command (Copia comando) ed eseguire il comando sul cluster da aggiungere al cluster hub. In questo modo viene avviata l'installazione degli agenti necessari sul cluster e, al termine di questo processo, il cluster viene visualizzato nell'elenco dei cluster con lo stato Ready.



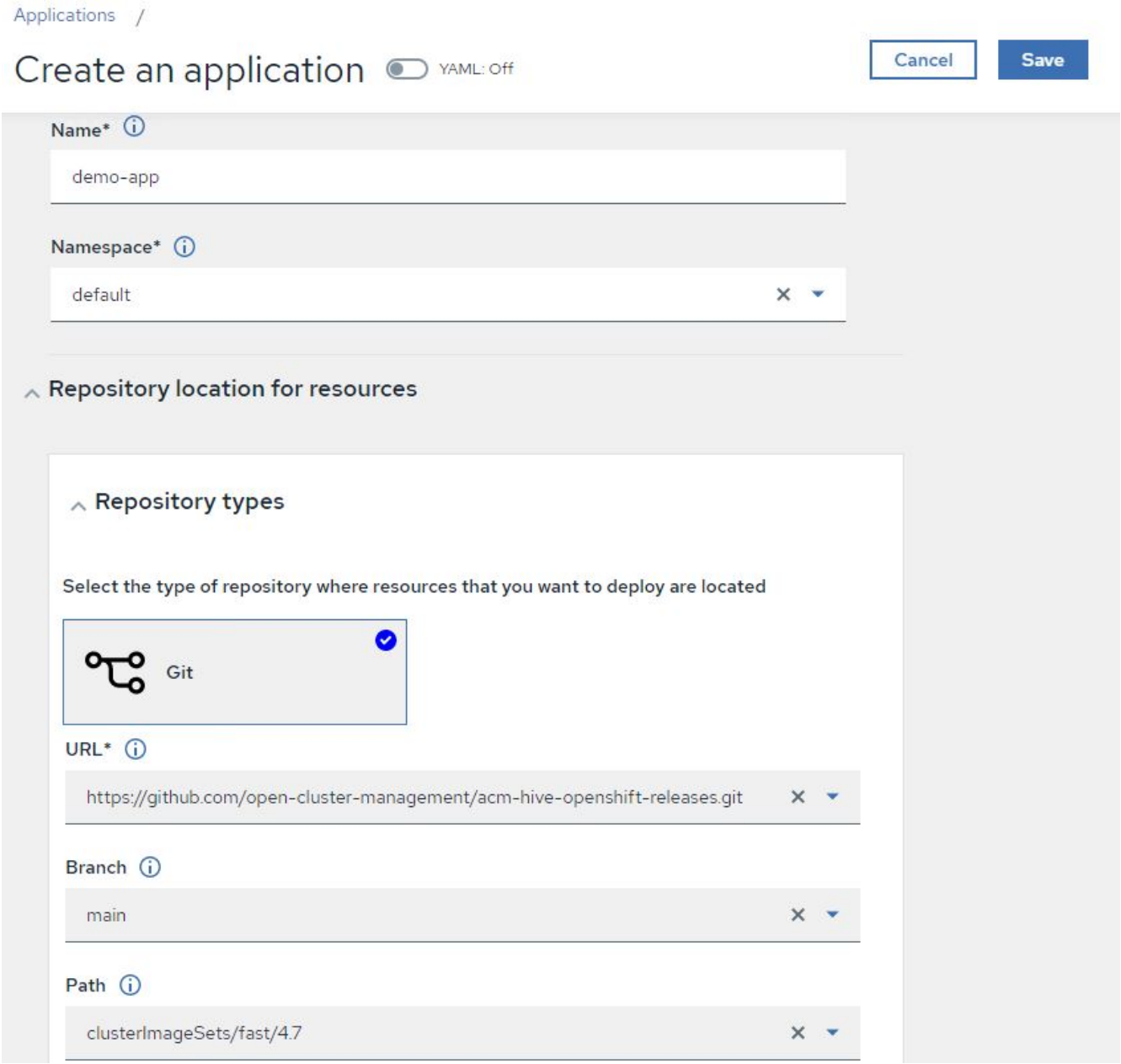
4. Dopo aver creato e importato più cluster, è possibile monitorarli e gestirli da una singola console.

Funzionalità: Gestione avanzata dei cluster per Kubernetes su Red Hat OpenShift con NetApp

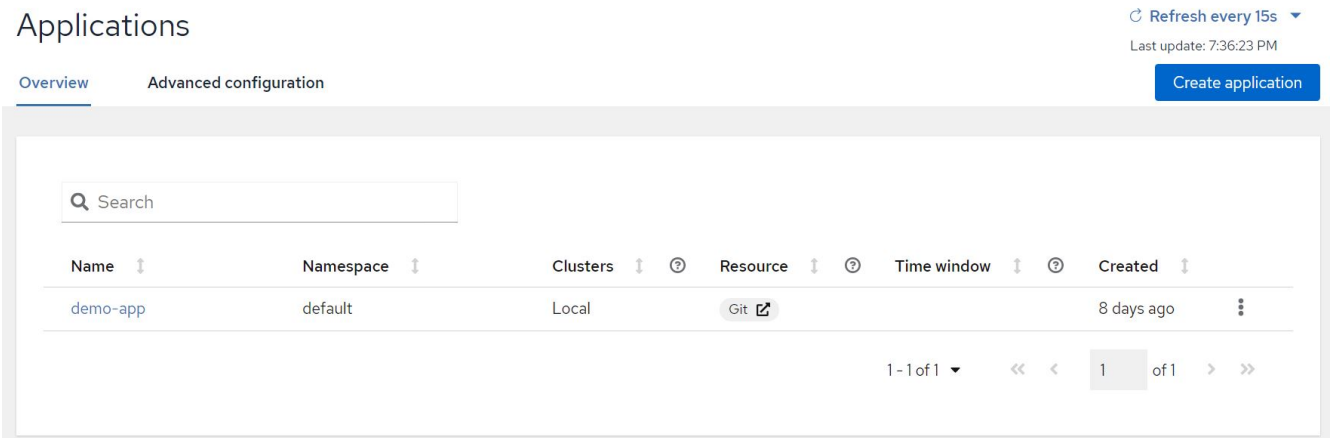
Gestione del ciclo di vita dell'applicazione

Per creare un'applicazione e gestirla in un insieme di cluster,

1. Accedere a Manage Applications (Gestisci applicazioni) dalla barra laterale e fare clic su Create Application (Crea applicazione). Fornire i dettagli dell'applicazione che si desidera creare e fare clic su Save (Salva).



2. Una volta installati i componenti dell'applicazione, l'applicazione viene visualizzata nell'elenco.



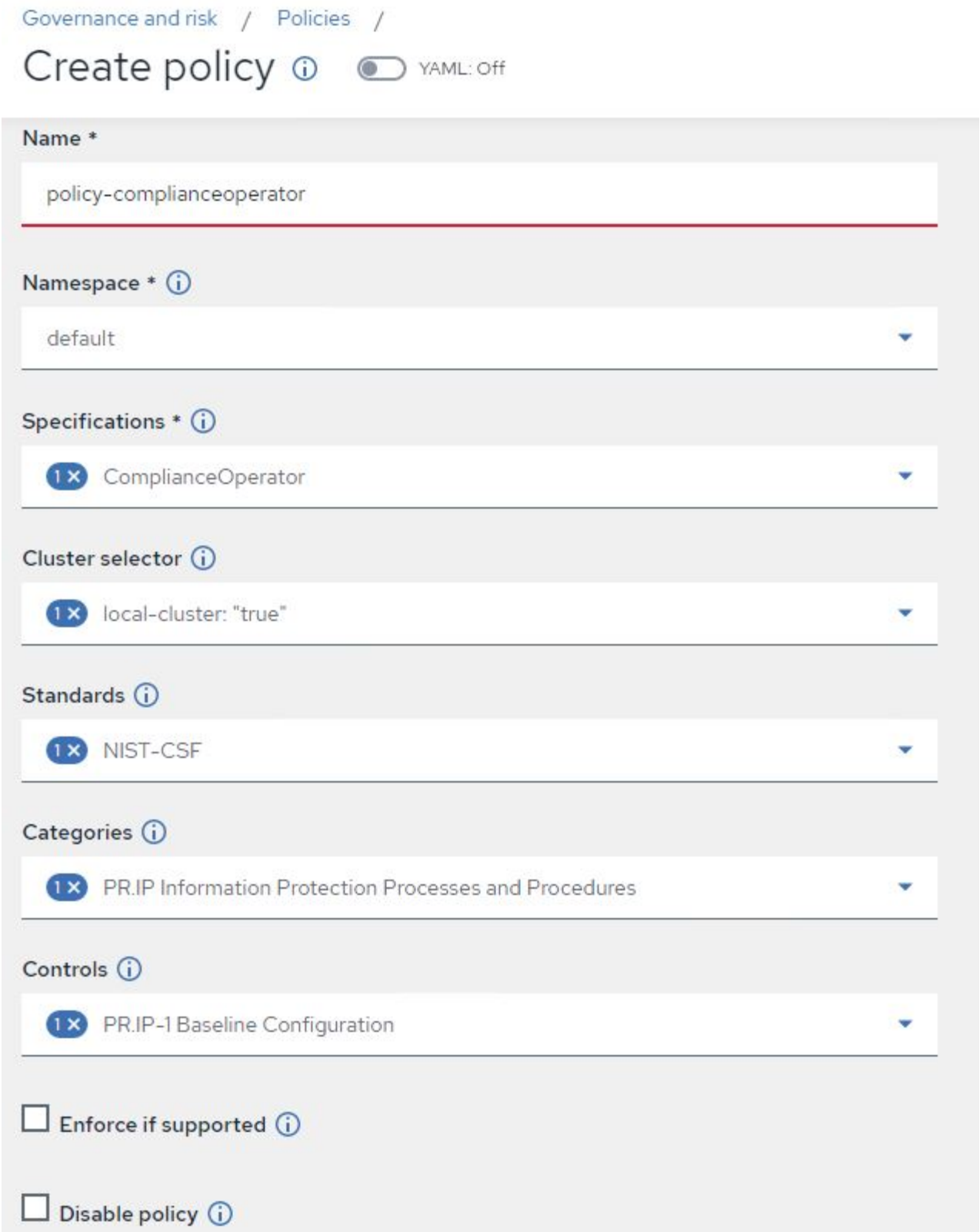
3. L'applicazione può ora essere monitorata e gestita dalla console.

Funzionalità: Gestione avanzata dei cluster per Kubernetes su Red Hat OpenShift con NetApp

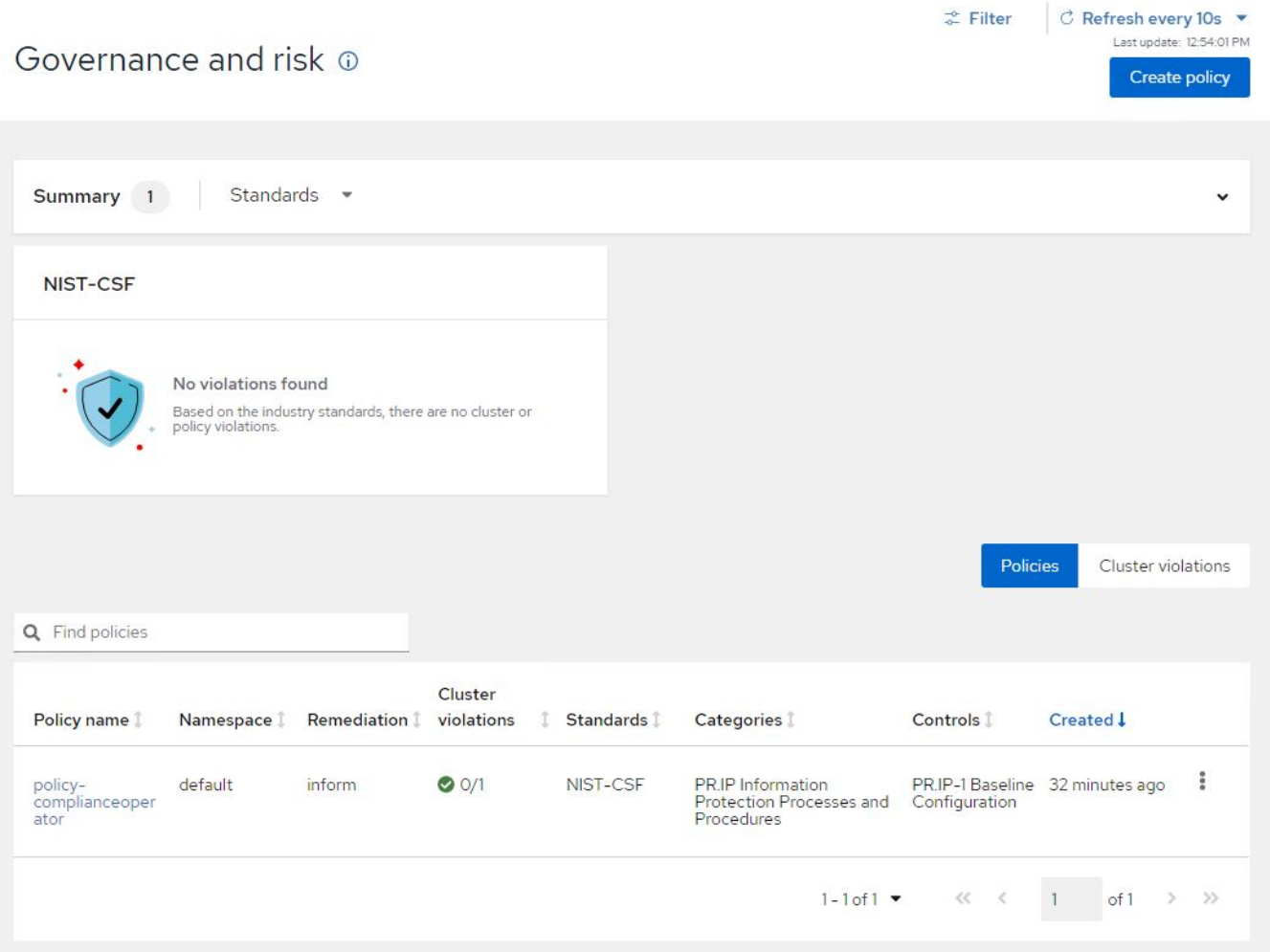
Governance e rischi

Questa funzionalità consente di definire le policy di conformità per diversi cluster e di assicurarsi che i cluster aderiscano ad esso. È possibile configurare le policy per informare o correggere eventuali deviazioni o violazioni delle regole.

1. Accedere a Governance and Risk (Governance e rischi) dalla barra laterale.
2. Per creare policy di compliance, fare clic su Create Policy (Crea policy), inserire i dettagli degli standard dei policy e selezionare i cluster che devono aderire a tali policy. Se si desidera correggere automaticamente le violazioni di questa policy, selezionare la casella di controllo Applica se supportato e fare clic su Crea.



3. Dopo aver configurato tutti i criteri richiesti, è possibile monitorare e correggere eventuali violazioni di policy o cluster da Advanced Cluster Management.

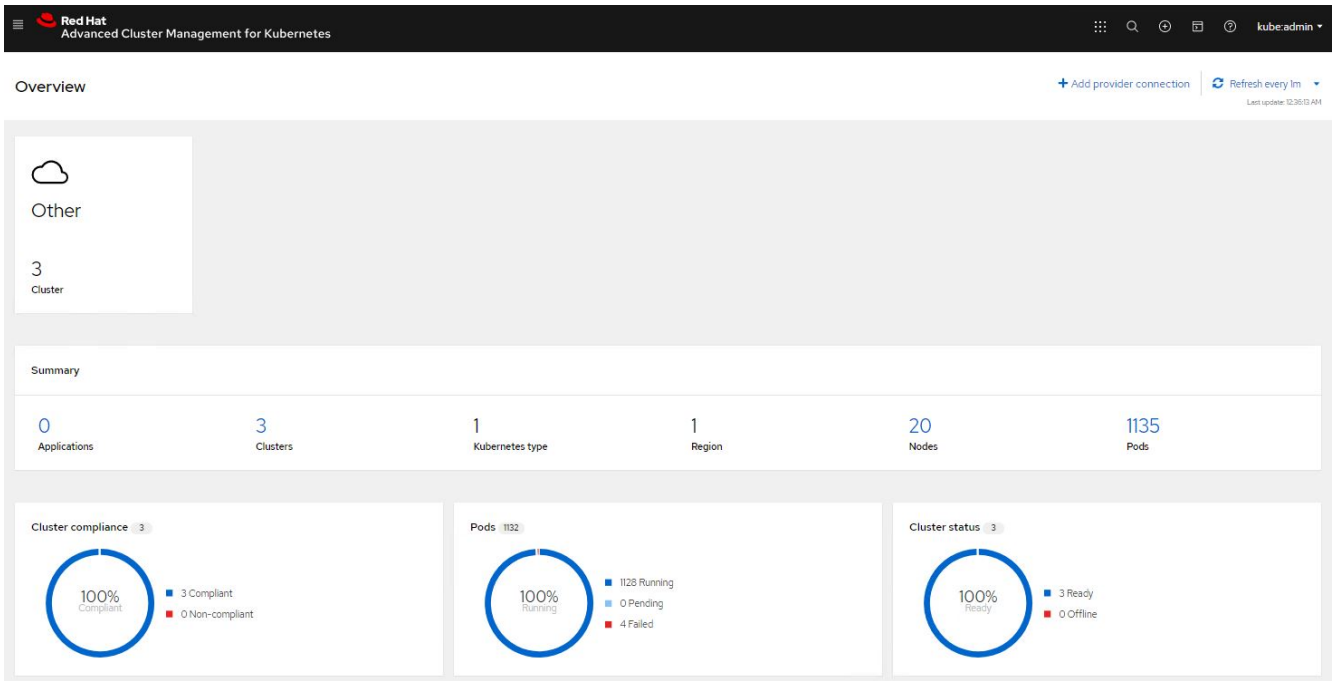


Funzionalità: Gestione avanzata dei cluster per Kubernetes su Red Hat OpenShift con NetApp

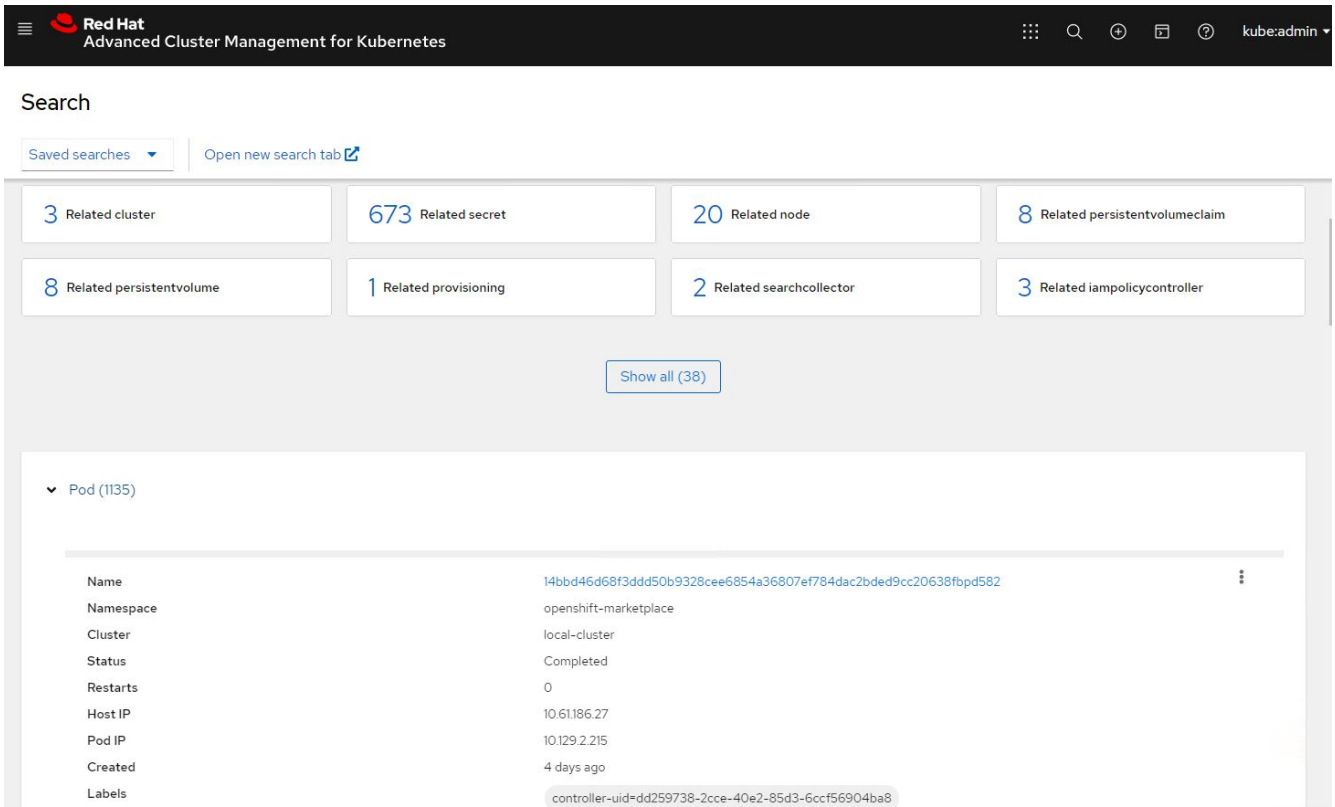
Osservabilità

La gestione avanzata dei cluster per Kubernetes consente di monitorare nodi, pod, applicazioni e carichi di lavoro in tutti i cluster.

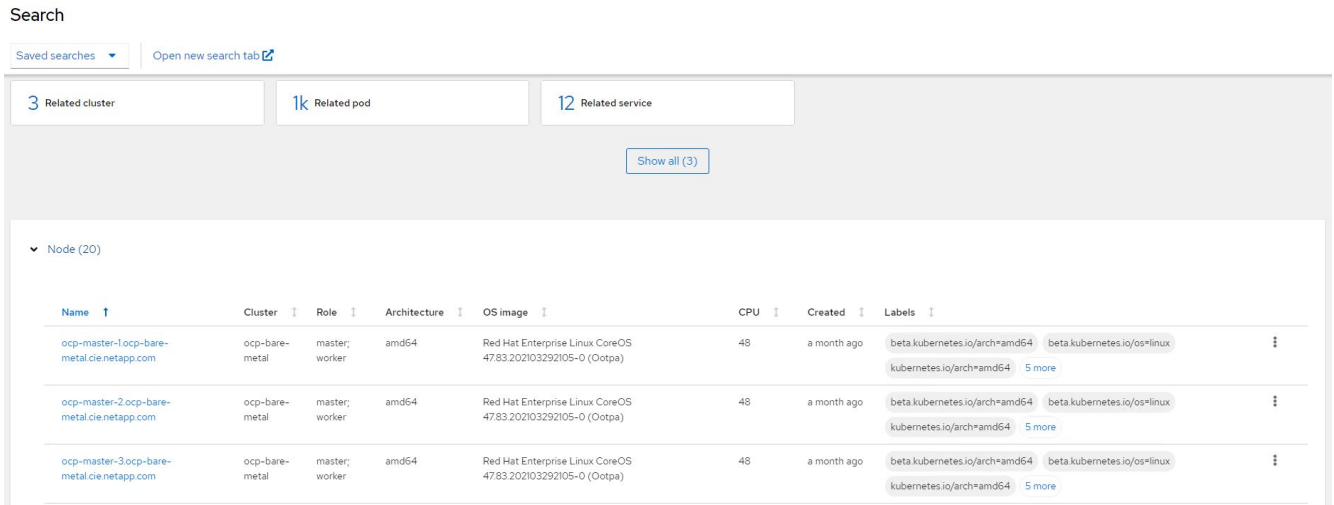
1. Accedere a osservare gli ambienti > Panoramica.



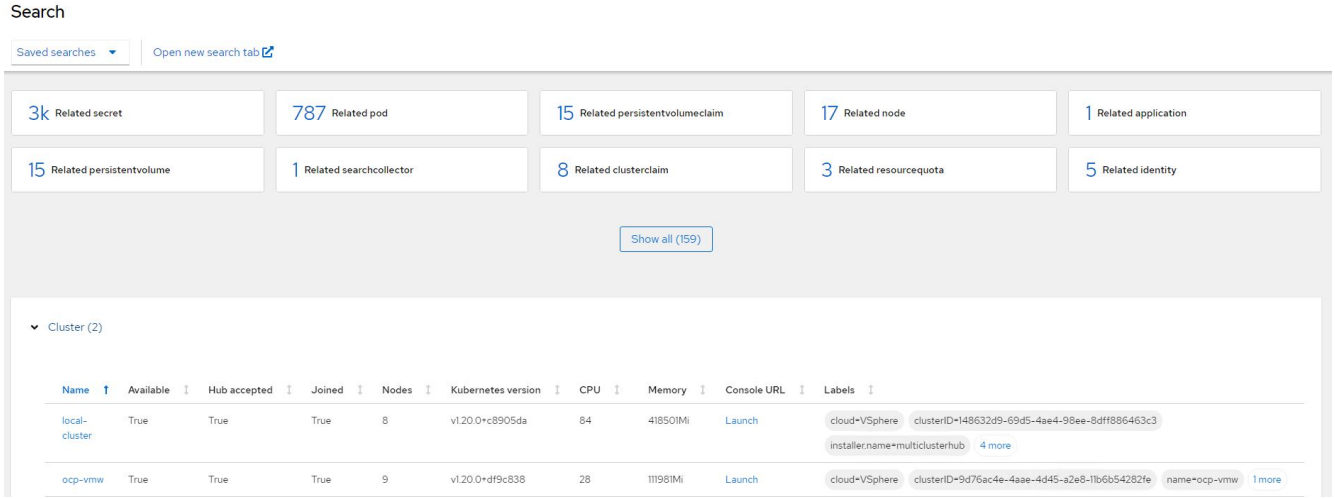
2. Tutti i pod e i carichi di lavoro di tutti i cluster vengono monitorati e ordinati in base a una varietà di filtri. Fare clic su Pod per visualizzare i dati corrispondenti.



3. Tutti i nodi dei cluster vengono monitorati e analizzati in base a una varietà di punti dati. Fare clic su Nodes (nodi) per ulteriori informazioni sui dettagli corrispondenti.



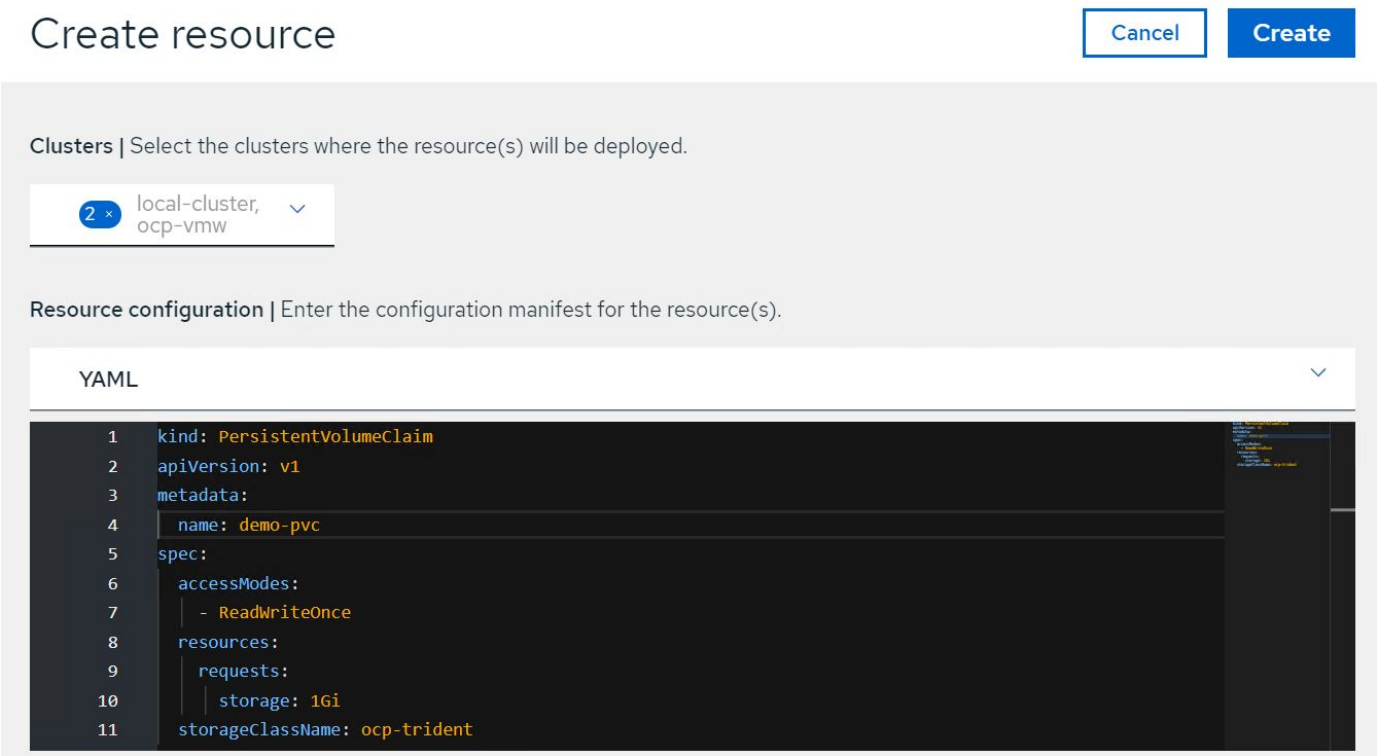
4. Tutti i cluster vengono monitorati e organizzati in base a diverse risorse e parametri del cluster. Fare clic su Clusters (Clusters) per visualizzare i dettagli del cluster.



Funzionalità: Gestione avanzata dei cluster per Kubernetes su Red Hat OpenShift con NetApp

Creare risorse su più cluster

Advanced Cluster Management per Kubernetes consente agli utenti di creare risorse su uno o più cluster gestiti contemporaneamente dalla console. Ad esempio, se si dispone di cluster OpenShift in siti diversi supportati da diversi cluster NetApp ONTAP e si desidera eseguire il provisioning dei PVC in entrambi i siti, è possibile fare clic sul segno (+) nella barra superiore. Quindi selezionare i cluster in cui si desidera creare il PVC, incollare la risorsa YAML e fare clic su Create (Crea).



Video e demo: Red Hat OpenShift con NetApp

I seguenti video mostrano alcune delle funzionalità documentate in questo documento:

Utilizzo di Red Hat MTV per migrare le VM alla virtualizzazione OpenShift con lo storage NetApp ONTAP

Accelera lo sviluppo del software con Astra Control e la tecnologia NetApp FlexClone - Red Hat OpenShift con NetApp

Sfrutta NetApp Astra Control per eseguire l'analisi post-mortem e ripristinare l'applicazione

Data Protection in pipeline ci/CD con Astra Control Center

Migrazione dei workload con Centro di controllo Astra - Red Hat OpenShift con NetApp

Migrazione dei workload - Red Hat OpenShift con NetApp

Installazione della virtualizzazione OpenShift - Red Hat OpenShift con NetApp

Implementazione di una macchina virtuale con virtualizzazione OpenShift - Red Hat OpenShift con NetApp

NetApp HCI per Red Hat OpenShift sulla virtualizzazione Red Hat

Ulteriori informazioni: Red Hat OpenShift con NetApp

Per ulteriori informazioni sulle informazioni descritte in questo documento, visitare i seguenti siti Web:

- Documentazione NetApp

  "https://docs.netapp.com/"

- Documentazione di Astra Trident

  "https://docs.netapp.com/us-en/trident/index.html"

- Documentazione di NetApp Astra Control Center

  "https://docs.netapp.com/us-en/astra-control-center/"

- Documentazione di Red Hat OpenShift

  "https://access.redhat.com/documentation/en-us/openshift_container_platform/4.7/"

- Documentazione sulla piattaforma Red Hat OpenStack

  "https://access.redhat.com/documentation/en-us/red_hat_openstack_platform/16.1/"

- Documentazione sulla virtualizzazione Red Hat

  "https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.4/"

- Documentazione VMware vSphere

  "https://docs.vmware.com/"