Opzioni di configurazione avanzate

NetApp Solutions

NetApp
September 26, 2024

Sommario

Opzioni di configurazione avanzate 1
Esplorazione delle opzioni di bilanciamento del carico 1
Creazione di registri immagine privati 21

Opzioni di configurazione avanzate

Esplorazione delle opzioni di bilanciamento del carico

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.

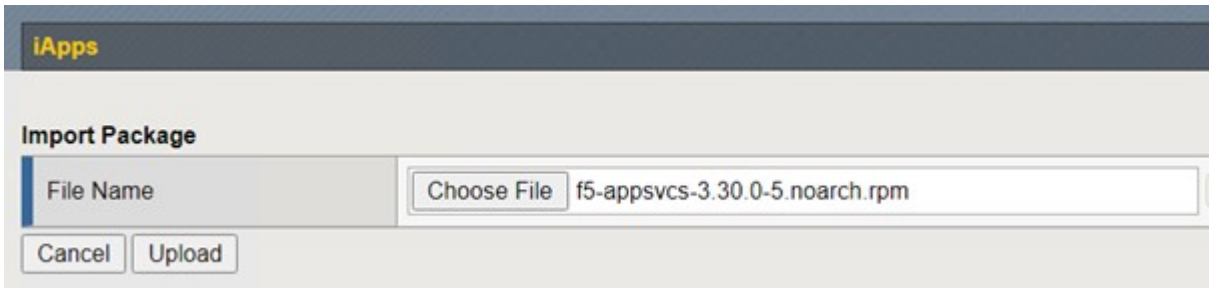
Un sistema F5 BIG-IP può essere implementato su hardware dedicato, nel cloud o come appliance virtuale on-premise con versioni superiori alla 12.x per l'integrazione con F5 CIS. Ai fini di questo documento, il sistema F5 BIG-IP è stato validato come appliance virtuale, ad esempio utilizzando L'edizione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

Release validate

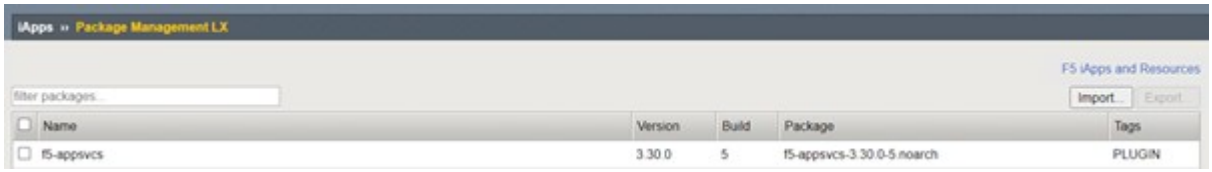
Tecnologia	Versione del software
Red Hat OpenShift	4.6 EUS, 4.7
F5 BIG-IP VE EDITION	16.1.0
F5 Container Ingress Service	2.5.1
F5 Controller IPAM	0.1.4
F5 AS3	3.30.0

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).

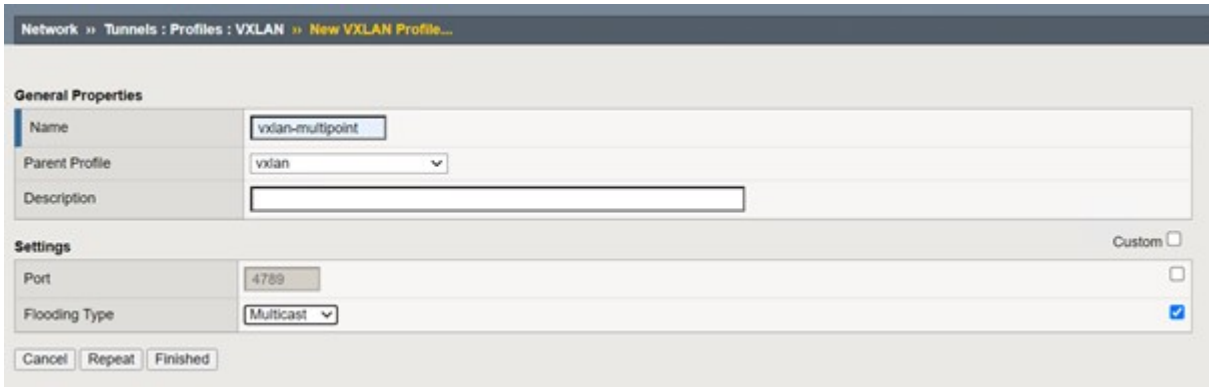


4. Verificare che l'estensione AS3 sia installata correttamente.

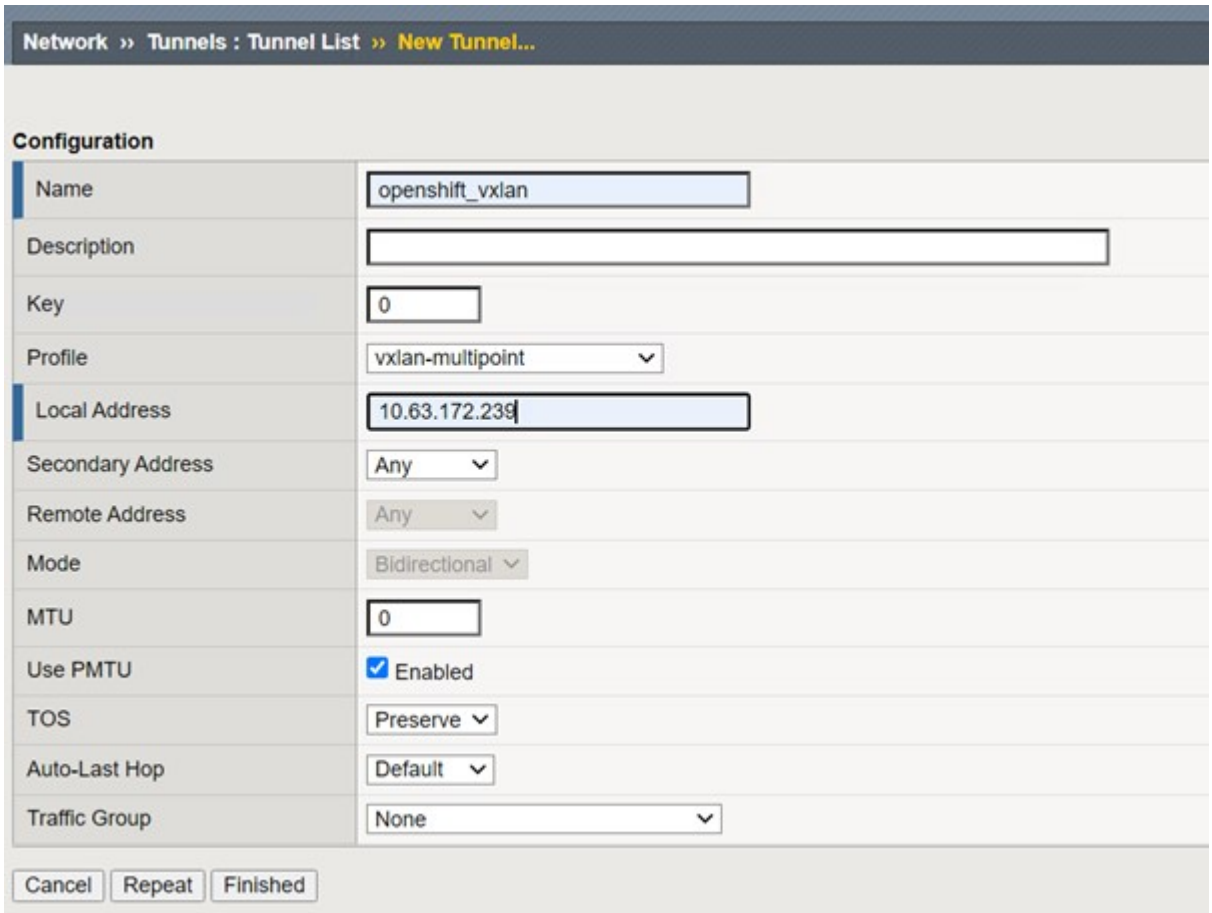


5. Quindi, configurare le risorse necessarie per la comunicazione tra OpenShift e I sistemi BIG-IP. Creare innanzitutto un tunnel tra OpenShift e IL SERVER BIG-IP creando un'interfaccia di tunnel VXLAN sul

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.

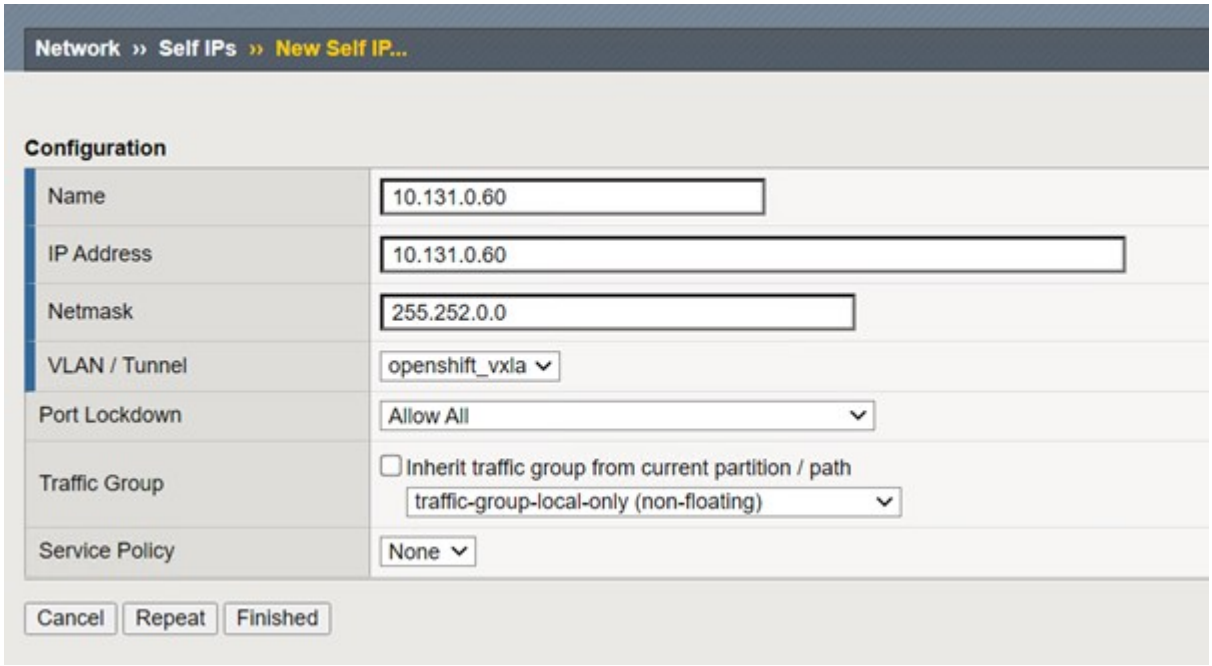
10. Creare la risorsa HostSubnet.

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

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

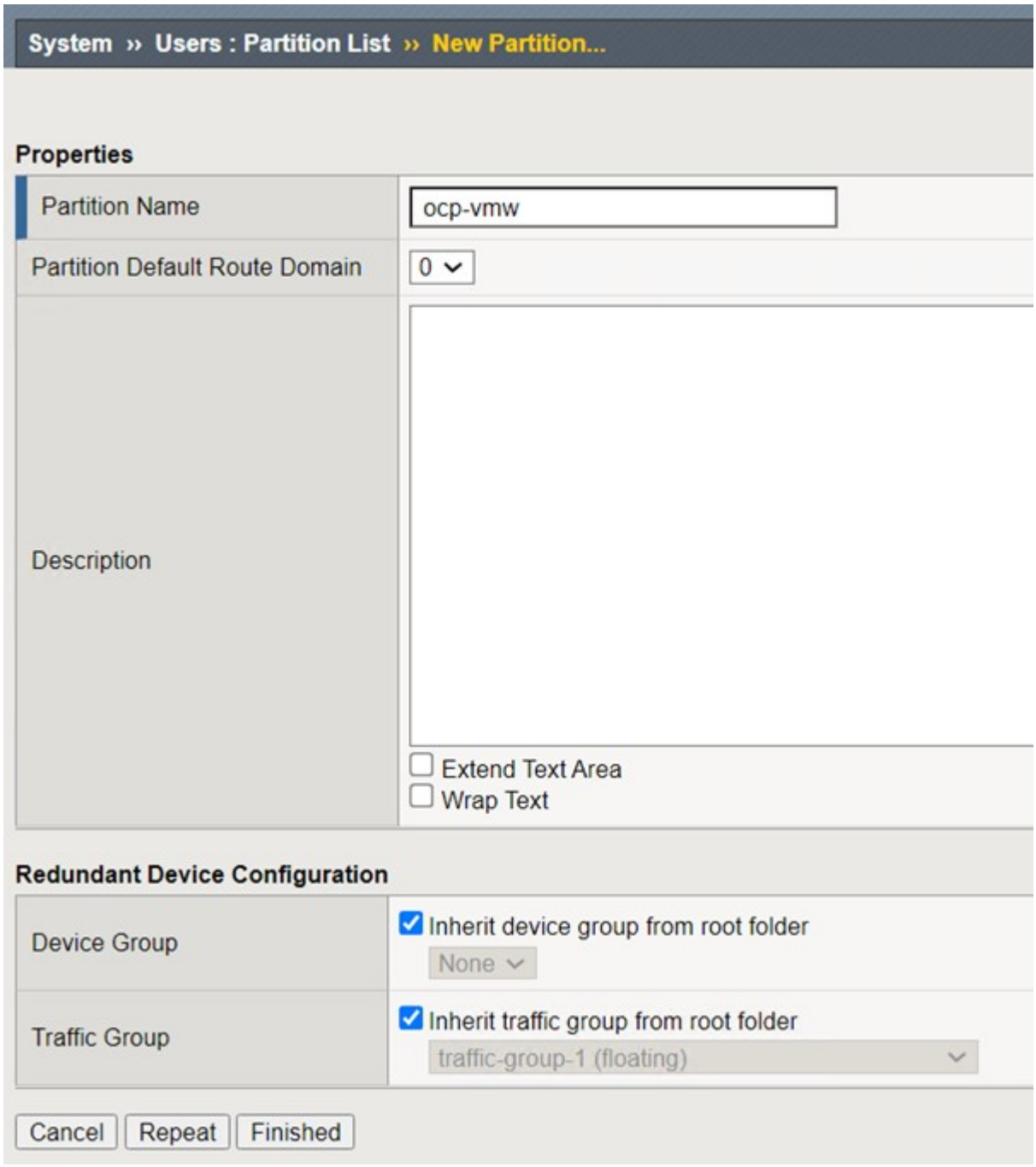
11. Ottenere l'intervallo di subnet IP del cluster per la subnet host creata per il server Big-IP F5.

```
[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.

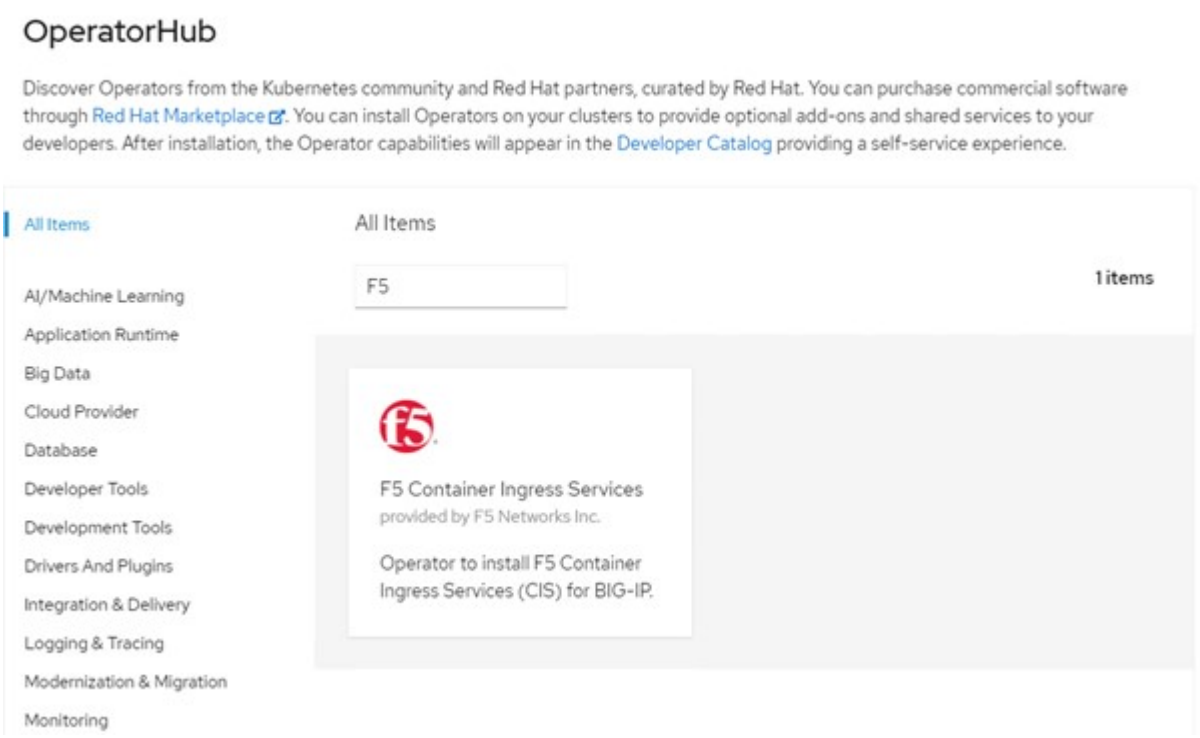
14. Installare F5 BIG-IP CIS utilizzando l'operatore di OperatorHub. Accedi al cluster Red Hat OpenShift con privilegi di amministrazione del cluster e crea un segreto con le credenziali di accesso del sistema F5 BIG-IP, un prerequisito per l'operatore.

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

secret/bigip-login created
```

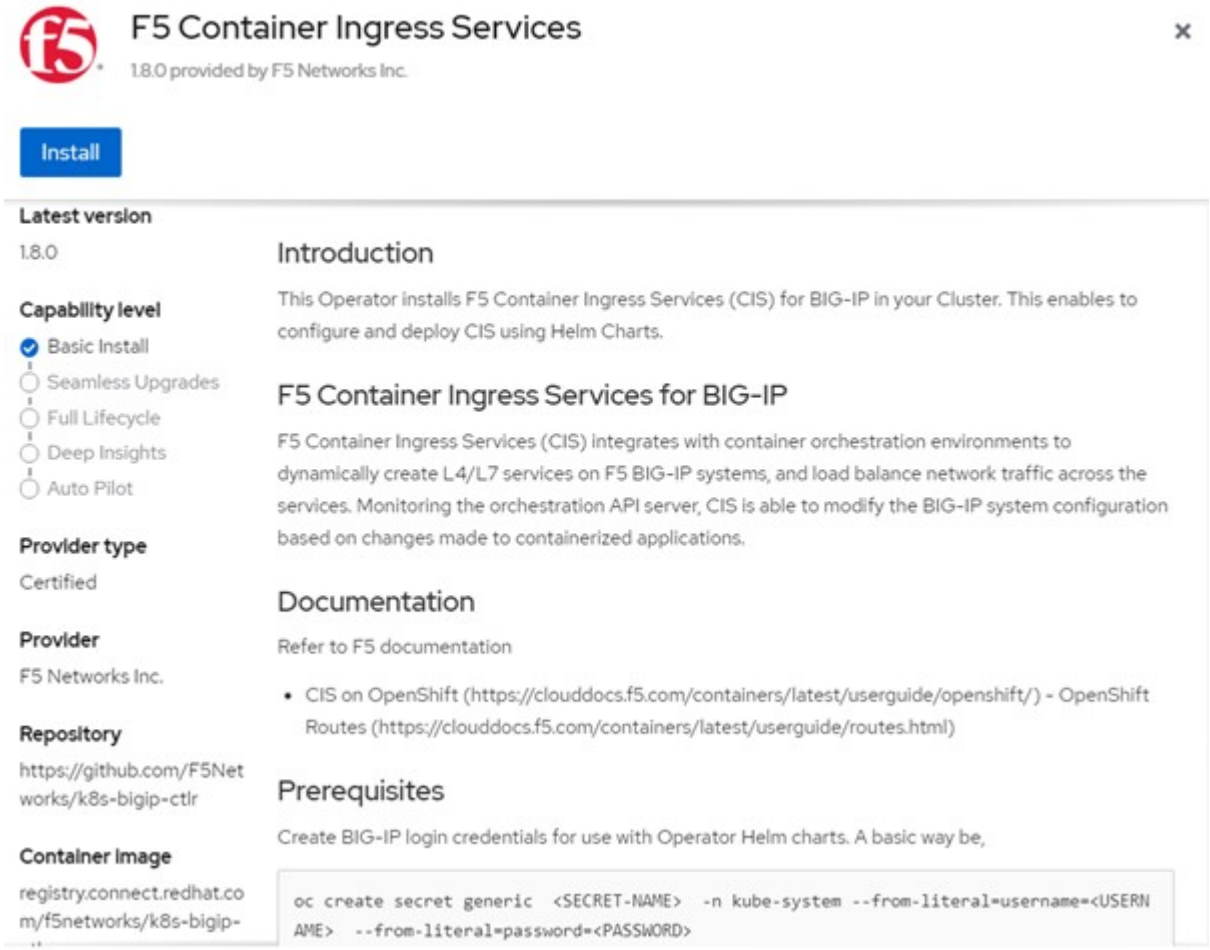
15. Installare F5 CIS CRD.

```
[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
```

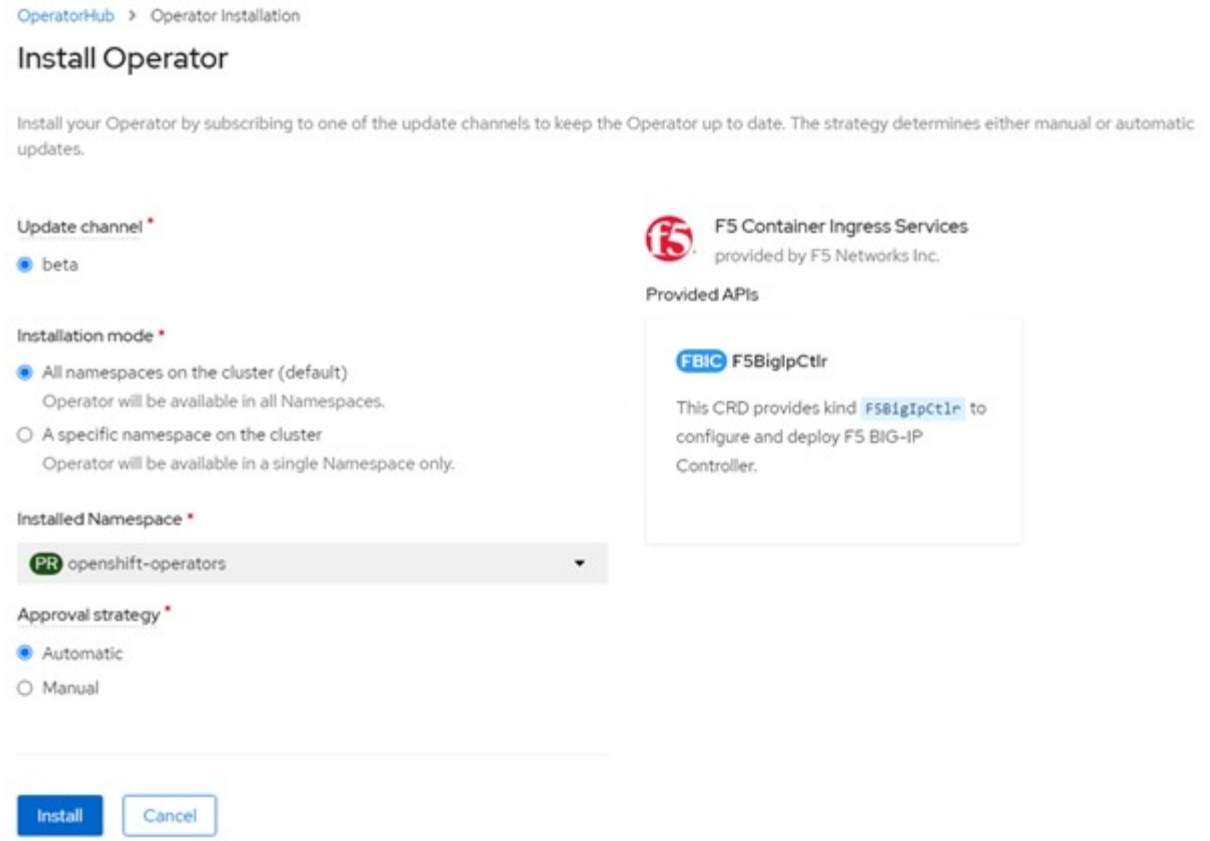
16. Accedere a Operator > OperatorHub, cercare la parola chiave F5 e fare clic sul riquadro F5 Container Ingress Service.



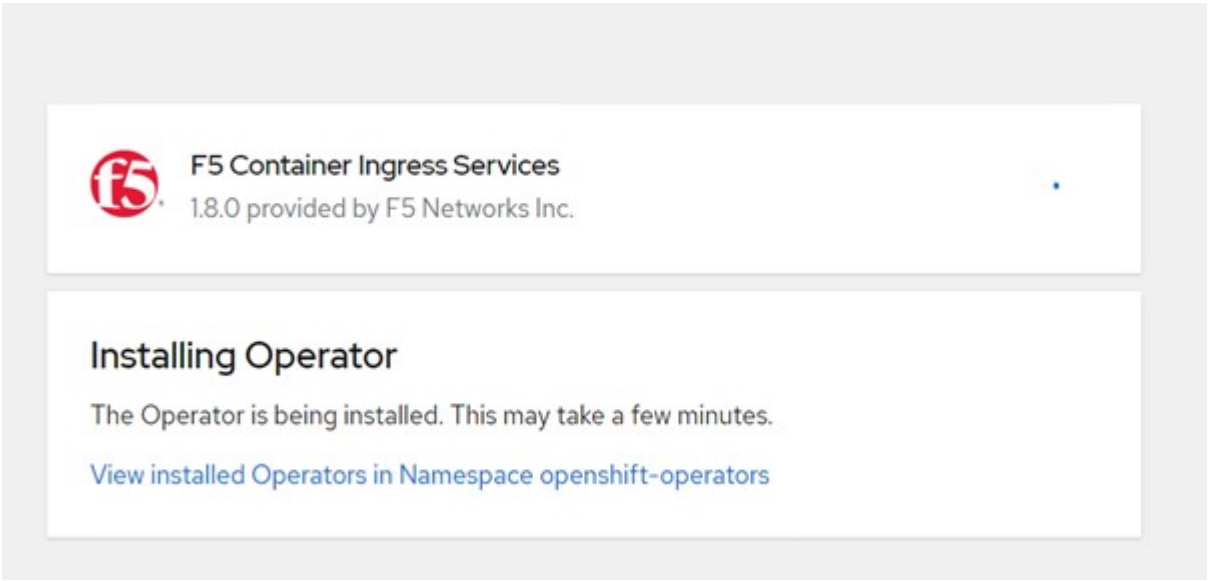
17. Leggere le informazioni dell'operatore e fare clic su Install (Installa).



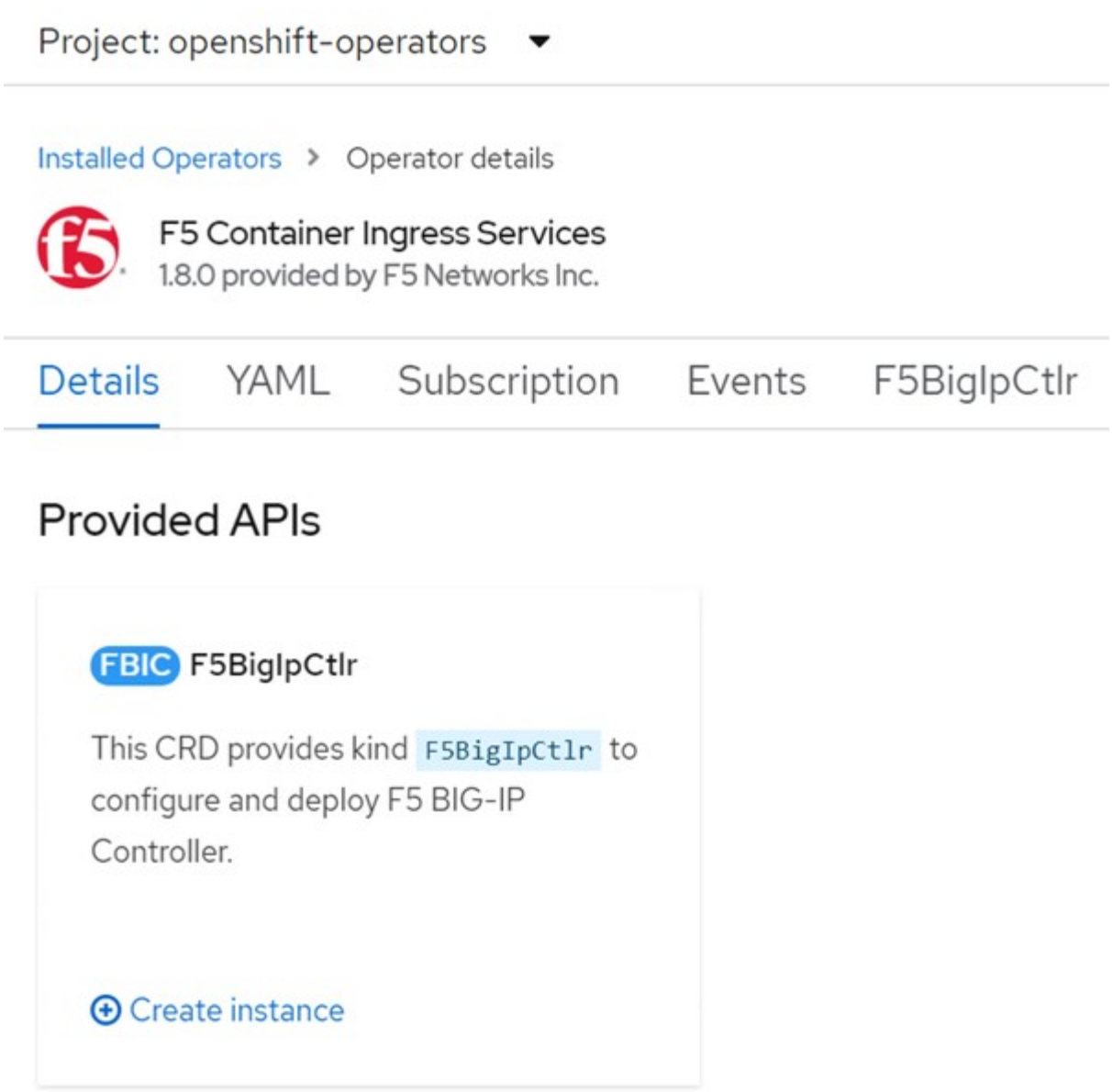
18. Nella schermata Install operator (Installa operatore), lasciare tutti i parametri predefiniti e fare clic su Install (Installa).



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.

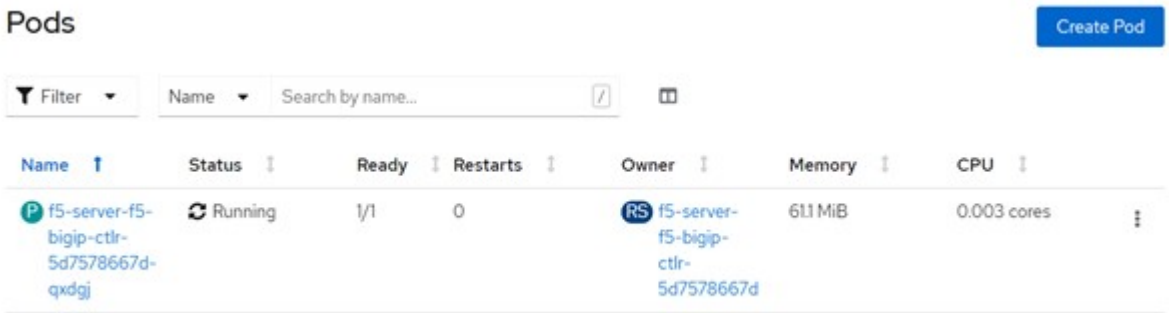


22. Fare clic su YAML View (Visualizza YAML) e incollare il seguente contenuto dopo aver aggiornato i parametri necessari.

> 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.

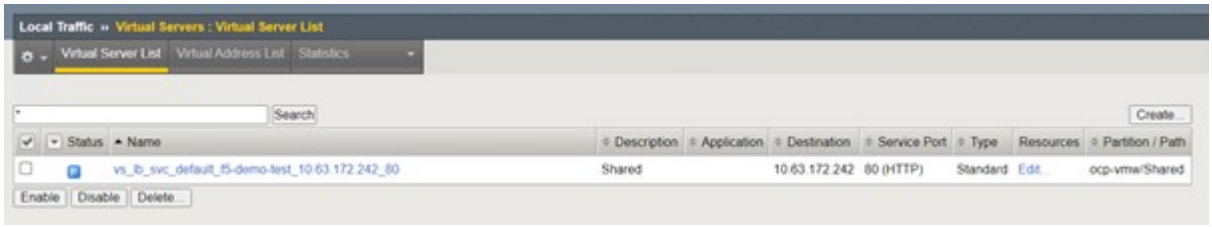
```
[admin@rhel-7 ~]$ vi example_deployment.yaml

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
```

Informazioni sul copyright

Informazioni sul marchio commerciale