

Guía de instalación y administración de NSX Container Plug-in para OpenShift

VMware NSX Container Plug-in 2.4
VMware NSX-T Data Center 2.4



vmware®

Puede encontrar la documentación técnica más actualizada en el sitio web de VMware:

<https://docs.vmware.com/es/>

El sitio web de VMware también ofrece las actualizaciones de producto más recientes.

Si tiene comentarios relacionados con esta documentación, envíelos a:

docfeedback@vmware.com

VMware, Inc.
3401 Hillview Ave.
Palo Alto, CA 94304
www.vmware.com

VMware Spain, S.L.
Calle Rafael Boti 26
2.ª planta
Madrid 28023
Tel.: +34 914125000
www.vmware.com/es

Copyright © 2017–2019 VMware, Inc. Todos los derechos reservados. [Información sobre el copyright y marca comercial](#).

Contenido

Guía de instalación y administración de NSX-T Container Plug-in para OpenShift 4

- 1 Descripción general de NSX-T Container Plug-in 5**
 - [Requisitos de compatibilidad 6](#)
 - [Descripción general de la instalación 6](#)
 - [Actualización de NCP 6](#)
- 2 Configurar recursos de NSX-T 8**
 - [Configurar los recursos de NSX-T 8](#)
- 3 Instalación de NCP 12**
 - [Requisitos del sistema 12](#)
 - [Preparación del archivo de hosts de Ansible 13](#)
- 4 Equilibrio de carga 18**
 - [Configuración del equilibrio de carga 18](#)
- 5 Administrar NSX Container Plug-in 25**
 - [Administrar los bloques de IP desde la GUI de NSX Manager 25](#)
 - [Ver subredes de bloques de IP desde la GUI de NSX Manager 26](#)
 - [Puertos lógicos conectados a CIF 26](#)
 - [Comandos de la CLI 27](#)
 - [Códigos de error 38](#)

Guía de instalación y administración de NSX-T Container Plug-in para OpenShift

Esta guía describe cómo instalar y administrar NSX Container Plug-in (NCP) para integrar NSX-T Data Center y OpenShift.

Público objetivo

Esta guía está destinada a administradores de red y de sistemas. Se supone que está familiarizado con la instalación y administración de NSX-T Data Center y OpenShift.

Glosario de publicaciones técnicas de VMware

Publicaciones técnicas de VMware proporciona un glosario de términos que podrían resultarle desconocidos. Si desea ver las definiciones de los términos que se utilizan en la documentación técnica de VMware, acceda a la página <http://www.vmware.com/support/pubs>.

Descripción general de NSX-T Container Plug-in

1

NSX Container Plug-in (NCP) integra NSX-T Data Center y orquestadores de contenedores, como Kubernetes, así como también integra NSX-T Data Center y productos de software de PaaS (plataforma como servicio) basados en contenedores, como OpenShift. Esta guía describe cómo configurar NCP con OpenShift.

El componente principal de NCP se ejecuta en un contenedor y se comunica con NSX Manager y con el plano de control de OpenShift. NCP supervisa los cambios de los contenedores y otros recursos, y administra los recursos de redes, como los puertos lógicos, los conmutadores, los enrutadores y los grupos de seguridad de los contenedores realizando llamadas a NSX API.

El complemento CNI de NSX se ejecuta en cada nodo de OpenShift. Supervisa los eventos del ciclo de vida de los contenedores, conecta una interfaz de contenedor al vSwitch invitado y programa este vSwitch para que etiquete y reenvíe el tráfico entre las interfaces de contenedor y la VNIC.

NCP ofrece las siguientes funcionalidades:

- Crea de manera automática una topología lógica de NSX-T para un clúster de OpenShift y crea una red lógica independiente para cada espacio de nombres de OpenShift.
- Conecta pods de OpenShift a la red lógica y asigna direcciones IP y MAC.
- Admite la traducción de direcciones de red (Network Address Translation, NAT) y asigna una dirección IP de SNAT independiente para cada espacio de nombres de OpenShift.

Nota Al configurar NAT, el número total de direcciones IP traducidas no puede ser superior a 1000.

- Implementa directivas de red de OpenShift con el firewall distribuido de NSX-T.
 - Compatibilidad con las directivas de red de entrada y salida.
 - Compatibilidad con el selector IPBlock en las directivas de red.
 - Compatibilidad con `matchLabels` y `matchExpression` al especificar selectores de etiquetas para directivas de redes.
- Implementa la ruta de OpenShift con el equilibrador de carga de capa 7 de NSX-T.
 - Compatibilidad con las rutas de HTTP y de HTTPS con terminación de Edge de TLS.
 - Compatibilidad con rutas con back-ends alternativos y subdominios comodín.

- Crea etiquetas en el puerto de conmutador lógico de NSX-T para el espacio de nombres, el nombre de pod y las etiquetas de un pod, y permite al administrador definir directivas y grupos de seguridad de NSX-T Data Center con base en etiquetas.

En esta versión, NCP solo admite un clúster de OpenShift.

Este capítulo incluye los siguientes temas:

- [Requisitos de compatibilidad](#)
- [Descripción general de la instalación](#)
- [Actualización de NCP](#)

Requisitos de compatibilidad

NSX Container Plug-in (NCP) tiene los siguientes requisitos de compatibilidad.

Productos de software	Versión
NSX-T Data Center	2.3, 2.4
Hipervisor para las máquinas virtuales de los hosts del contenedor	<ul style="list-style-type: none"> ■ Versión de vSphere admitida ■ RHEL KVM 7.4, 7.5, 7.6
Sistema operativo del host del contenedor	RHEL 7.4, 7.5, 7.6
Plataforma como servicio	OpenShift 3.10, 3.11
Open vSwitch del host del contenedor	2.10.2 (se incluye con NSX-T Data Center 2.4)

Descripción general de la instalación

La instalación y configuración de NCP incluye los siguientes pasos. Para realizar los pasos correctamente, debe estar familiarizado con la instalación y administración de NSX-T Data Center y OpenShift.

- 1 Instale NSX-T Data Center.
- 2 Crear una zona de transporte superpuesta.
- 3 Crear un conmutador lógico superpuesto y conectar los nodos al conmutador.
- 4 Crear un enrutador lógico de nivel 0.
- 5 Crear bloques de IP para los pods.
- 6 Crear grupos de IP para la traducción de direcciones de red de origen (Source Network Address Translation, SNAT).
- 7 Preparar el archivo de hosts de Ansible.
- 8 Instale NCP y OpenShift mediante un solo playbook.

Actualización de NCP

Esta sección describe cómo actualizar NCP a la versión 2.4.0.

Procedimiento

- 1 Actualice el paquete RPM de CNI, el DaemonSet del agente de nodo de NSX y el ReplicationController de NCP.
- 2 Prepare el archivo de hosts de Ansible.

Cada nodo debe tener el parámetro `openshift_node_group_name` especificado. Por ejemplo,

```
[nodes]
config-master.example.com openshift_hostname=config-master.example.com
openshift_node_group_name=config-master
```

- 3 (opcional) Configure el equilibrio de carga.

Agregue un paso a fin de especificar un grupo de direcciones IP diferente para las direcciones IP externas para el servicio de equilibrador de carga. Por ejemplo,

```
external_ip_pools_lb = <nsx ip pool name>
```

Configurar recursos de NSX-T

Se deben crear recursos de NSX-T Data Center para proporcionar redes a nodos de OpenShift.

Configurar los recursos de NSX-T

Los recursos de NSX-T Data Center que necesita configurar incluyen una zona de transporte de superposición, un enrutador lógico de nivel 0, un conmutador lógico para conectar las máquinas virtuales del nodo, bloques de IP para los nodos de Kubernetes y un grupo de IP para SNAT.

Importante Si ejecuta NSX-T Data Center 2.4 o una versión posterior, debe configurar los recursos de NSX-T mediante la pestaña **Opciones avanzadas de redes y seguridad**.

En el archivo de configuración de NCP `ncp.ini`, los recursos de NSX-T Data Center se especifican mediante sus UUID o nombres.

Zona de transporte superpuesta

Inicie sesión en NSX Manager y busque la zona de transporte superpuesta que se usa para las redes de los contenedores o cree una nueva.

Especifique una zona de transporte superpuesta de un clúster configurando la opción `overlay_tz` en la sección `[nsx_v3]` de `ncp.ini`. Este paso es opcional. Si no se configura `overlay_tz`, NCP recuperará automáticamente el identificador de zona de transporte superpuesta del enrutador de nivel 0.

Enrutamiento lógico de nivel 0

Inicie sesión en NSX Manager y busque el enrutador que se usa para las redes de los contenedores o cree uno nuevo.

Especifique un enrutador lógico de nivel 0 de un clúster configurando la opción `tier0_router` en la sección `[nsx_v3]` de `ncp.ini`.

Nota El enrutador se debe crear en modo activo-en espera.

Conmutador lógico

Las vNIC que usan el nodo para el tráfico de datos deben estar conectadas a un conmutador lógico superpuesto. No es obligatorio que la interfaz de administración del nodo esté conectada a NSX-T Data Center, aunque hacerlo facilitaría la configuración. Puede crear un conmutador lógico iniciando sesión en NSX Manager. En el conmutador, cree puertos lógicos y asocie las vNIC del nodo a ellos. Los puertos lógicos deben tener las siguientes etiquetas:

- etiqueta: <cluster_name>, ámbito: ncp/cluster
- etiqueta: <node_name>, ámbito: ncp/node_name

El valor de <cluster_name> debe coincidir con el valor de la opción cluster de la sección [coe] de ncp.ini.

Bloques de IP para los pods de Kubernetes

Inicie sesión en NSX Manager y cree uno o varios bloques de direcciones IP. Especifique el bloque de IP en formato CIDR

Especifique bloques de IP de pods de Kubernetes configurando la opción container_ip_blocks en la sección [nsx_v3] de ncp.ini.

También puede crear bloques de IP específicamente para espacios de nombres que no sean SNAT.

Especifique bloques de IP que no sean SNAT configurando la opción no_snat_ip_blocks en la sección [nsx_v3] de ncp.ini.

Si crea bloques de IP que no sean SNAT mientras NCP se ejecuta, debe reiniciar NCP. De lo contrario, NCP seguirá usando los bloques de IP compartidos hasta que se agoten.

Nota Cuando cree un bloque de IP, el prefijo no debe tener una longitud superior al valor del parámetro subnet_prefix en el archivo de configuración de NCP ncp.ini.

Grupo de IP de SNAT

El grupo de IP se emplea para asignar las direcciones IP que se usarán para traducir las IP de los pods mediante reglas SNAT y para exponer las controladoras de entrada a través de las reglas SNAT/DNAT, igual que las IP flotantes de OpenStack. Estas direcciones IP también se denominan "IP externas".

Varios clústeres de Kubernetes usan el mismo grupo de IP externas. Cada instancia de NCP usa un subgrupo de este grupo para el clúster de Kubernetes que administra. De forma predeterminada, se usará el mismo prefijo de subred para subredes de pods. Para usar un tamaño de subred diferente, actualice la opción external_subnet_prefix en la sección [nsx_v3] de ncp.ini.

Inicie sesión en NSX Manager y cree un grupo o busque uno ya existente.

Especifique grupos de IP para SNAT configurando la opción external_ip_pools en la sección [nsx_v3] de ncp.ini.

También puede configurar SNAT para un determinado servicio agregando una anotación en el servicio. Por ejemplo,

```
apiVersion: v1
kind: Service
metadata:
  name: svc-example
  annotations:
    ncp/snat_pool: <external IP pool ID or name>
  selector:
    app: example
...
```

NCP configurará la regla SNAT para este servicio. La IP de origen de la regla es el conjunto de pods de back-end. La IP de destino es la IP SNAT asignada desde el grupo de IP externo especificado. Tenga en cuenta lo siguiente:

- El grupo de direcciones IP que `ncp/snat_pool` especifica ya debe existir en NSX-T Data Center antes de configurar el servicio. El grupo de direcciones IP debe tener la etiqueta `{"ncp/owner": cluster:<cluster>}`.
- En NSX-T Data Center, la prioridad de la regla SNAT para el servicio es mayor que la del proyecto.
- Si se configura un pod con varias reglas SNAT, solo funcionará una.

Puede especificar el espacio de nombres al que es posible asignar direcciones IP del grupo de direcciones IP de SNAT agregando la siguiente etiqueta al grupo de direcciones IP.

- En el ámbito `ncp/owner`, la etiqueta debe ser `ns:<namespace_UUID>`.

Puede obtener el UUID de espacio de nombres con uno de los siguientes comandos:

```
oc get ns -o yaml
```

Tenga en cuenta lo siguiente:

- Cada etiqueta debe especificar un UUID. Puede crear varias etiquetas para el mismo grupo.
- Si se cambian las etiquetas después de que se asignen direcciones IP a algunos espacios de nombres en función de las etiquetas anteriores, no se recuperarán dichas direcciones IP hasta que se cambien los ajustes de SNAT de los servicios o se reinicie NCP.
- La etiqueta de propietario de espacio de nombres es opcional. Sin esta etiqueta, se pueden asignar direcciones IP del grupo de direcciones IP de SNAT a cualquier espacio de nombres.

(Opcional) Secciones del marcador de firewall

Para que el administrador pueda crear reglas de firewall que no interfieran con las secciones del firewall creadas por NCP en función de las directivas de red, inicie sesión en NSX Manager y cree dos secciones de firewall.

Especifique secciones de firewall del marcador configurando las opciones

`bottom_firewall_section_marker` y `top_firewall_section_marker` en la sección `[nsx_v3]` de `ncp.ini`.

La sección del firewall inferior debe estar bajo la sección del firewall superior. Con estas secciones del firewall, todas las secciones del firewall creadas por NCP para el aislamiento se crearán sobre la sección del firewall inferior, y todas las secciones del firewall creadas por NCP para la directiva se crearán bajo la sección del firewall superior. Si no se crean estas secciones de marcador, todas las reglas de aislamiento se crearán en la parte inferior, y todas las secciones de la directiva se crearán en la parte superior. No puede haber varias secciones del firewall de marcador con el mismo valor en cada clúster, ya que se producirá un error.

Instalación de NCP

NCP está completamente integrado con OpenShift. Cuando se agregan los parámetros necesarios en el archivo de hosts de Ansible y se instala OpenShift, NCP se instala automáticamente.

Este capítulo incluye los siguientes temas:

- [Requisitos del sistema](#)
- [Preparación del archivo de hosts de Ansible](#)

Requisitos del sistema

Antes de instalar OpenShift, asegúrese de que el entorno cumpla con ciertos requisitos.

Requisitos generales

- Ansible 2.4 o posterior.

Requisitos de máquina virtual

Las máquinas virtuales del nodo de OpenShift deben tener dos vNIC:

- Una vNIC de administración conectada al conmutador lógico que tiene un vínculo superior al enrutador de nivel 1 de administración.
- La segunda vNIC en todas las máquinas virtuales debe tener las siguientes etiquetas en NSX-T, de modo que NCP sepa qué puerto se utiliza como VIF principal para todos los pods que se ejecutan en el nodo OpenShift específico.

```
{'ncp/node_name': '<node_name>'}  
{'ncp/cluster': '<cluster_name>'}
```

Requisitos de máquina sin sistema operativo

- Los nodos de OpenShift deben ser nodos de transporte de NSX-T y las etiquetas mencionadas anteriormente deben aplicarse en los nodos de transporte en lugar de las VIF.
- El archivo de hosts de Ansible debe tener esta opción: `nsx_node_type='BAREMETAL'`.

Requisitos de NSX-T

- Un enrutador de nivel 0.
- Una zona de transporte superpuesta.
- Un bloque de direcciones IP para las redes de pods.
- (Opcional) Un bloque de direcciones IP para redes de pods enrutadas (no NAT).
- Un grupo de direcciones IP para SNAT. De forma predeterminada, el bloque de direcciones IP para las redes de pods solo se puede enrutar dentro de NSX-T. NCP utiliza este grupo de direcciones IP para proporcionar conectividad al exterior.
- (Opcional) Secciones de firewall superior e inferior. NCP colocará las reglas de directivas de red de Kubernetes entre estas dos secciones.
- Open vSwitch y los RPM del complemento CNI deben alojarse en un servidor HTTP accesible desde las máquinas virtuales del nodo de OpenShift.

Imagen de Docker de NCP

Actualmente, la imagen de Docker de NCP no está disponible públicamente. La imagen `nsx-ncp` debe encontrarse en un registro privado local; de lo contrario, haga lo siguiente:

```
ansible-playbook [-i /path/to/inventory] playbooks/prerequisites.yml
```

En todos los nodos:

```
docker load -i nsx-ncp-rhel-xxx.yyyyyyyy.tar
docker image tag registry.local/xxx.yyyyyyyy/nsx-ncp-rhel nsx-ncp
ansible-playbook [-i /path/to/inventory] playbooks/deploy_cluster.yml
```

Preparación del archivo de hosts de Ansible

Para que NCP se integre con OpenShift, debe especificar los parámetros de NCP en el archivo de hosts de Ansible.

Después de especificar los siguientes parámetros en el archivo de hosts de Ansible, NCP se instalará automáticamente cuando instale OpenShift.

- `openshift_use_nsx=True`
- `openshift_use_openshift_sdn=False`
- `os_sdn_network_plugin_name='cni'`
- `nsx_openshift_cluster_name='ocp-cluster1'`

(Obligatorio) Esto es obligatorio porque varios clústeres de OpenShift/Kubernetes pueden conectarse al mismo NSX Manager.

- `nsx_api_managers='10.10.10.10'`

(Obligatorio) Dirección IP de NSX Manager. Para un clúster de NSX Manager, especifique las direcciones IP separadas por comas.

- `nsx_tier0_router='MyT0Router'`

(Obligatorio) Nombre o UUID del enrutador de nivel 0 al que se conectarán los enrutadores de nivel 1 del proyecto.

- `nsx_overlay_transport_zone='my_overlay_tz'`

(Obligatorio) Nombre o UUID de la zona de transporte superpuesta que se utilizará para crear conmutadores lógicos.

- `nsx_container_ip_block='ip_block_for_my_ocp_cluster'`

(Obligatorio) Nombre o UUID de un bloque de direcciones IP configurado en NSX-T. Habrá una subred por proyecto fuera de este bloque de direcciones IP. Estas redes estarán detrás de SNAT y no serán enrutables.

- `nsx_ovs_uplink_port='ens224'`

(Obligatorio) Si está en modo HOSTVM. NSX-T necesita una segunda vNIC para las redes de pods en los nodos de OCP que sea distinta de la vNIC de administración. Se recomienda que ambas vNIC estén conectadas a conmutadores lógicos de NSX-T. La segunda vNIC (no de administración) debe proporcionarse aquí. Si no hay sistema operativo, este parámetro no es necesario.

- `nsx_cni_url='http://myserver/nsx-cni.rpm'`

(Obligatorio) Requisito temporal hasta que NCP pueda arrancar los nodos. Es necesario colocar `nsx-cni` en un servidor http.

- `nsx_ovs_url='http://myserver/openvswitch.rpm'`

- `nsx_kmod_ovs_url='http://myserver/kmod-openvswitch.rpm'`

(Obligatorio) Parámetros temporales hasta que NCP pueda arrancar los nodos. Puede omitirse en una instalación sin sistema operativo.

- `nsx_node_type='HOSTVM'`

(Opcional) El valor predeterminado es HOSTVM. Establezca esta opción como BAREMETAL si OpenShift no se ejecuta en las máquinas virtuales.

- `nsx_k8s_api_ip=192.168.10.10`

(Opcional) Si se establece esta opción, NCP se comunicará con esta dirección IP; de lo contrario, lo hará con la dirección IP del servicio de Kubernetes.

- `nsx_k8s_api_port=192.168.10.10`

(Opcional) Utiliza 443 como valor predeterminado para el servicio de Kubernetes. Establézcalo como 8443 si lo utiliza en combinación con `nsx_k8s_api_ip` para especificar la dirección IP del nodo principal.

- `nsx_insecure_ssl=true`

(Opcional) El valor predeterminado es `true`, ya que NSX Manager viene con un certificado que no es de confianza. Si cambió el certificado por otro de confianza, puede establecerlo como `false`.

- `nsx_api_user='admin'`
- `nsx_api_password='super_secret_password'`
- `nsx_subnet_prefix=24`

(Opcional) El valor predeterminado es 24. Este es el tamaño de subred que se dedicará a cada proyecto de OpenShift. Si el número de pods supera el tamaño de subred, se agregará un nuevo conmutador lógico con el mismo tamaño de subred al proyecto.

- `nsx_use_loadbalancer=true`

(Opcional) El valor predeterminado es `true`. Establézcalo en `false` si no desea utilizar equilibradores de carga de NSX-T para las rutas de OpenShift y los servicios de tipo equilibrador de carga.

- `nsx_lb_service_size='SMALL'`

(Opcional) El valor predeterminado es `SMALL`. En función del tamaño de NSX Edge, `MEDIUM` o `LARGE` también son posibles.

- `nsx_no_snat_ip_block='router_ip_block_for_my_ocp_cluster'`

(Opcional) Si la anotación `ncp/no_snat=true` se aplica en un proyecto o un espacio de nombres, la subred se tomará de este bloque de direcciones IP y no habrá ninguna SNAT para ella. Se espera que se pueda enrutar.

- `nsx_external_ip_pool='external_pool_for_snat'`

(Obligatorio) Grupo de direcciones IP para el equilibrador de carga y SNAT si `nsx_external_ip_pool_lb` no está definido.

- `nsx_external_ip_pool_lb='my_ip_pool_for_lb'`

(Opcional) Establezca esta opción si desea un grupo de direcciones IP diferente para Router y `SvcTypeLB`.

- `nsx_top_fw_section='top_section'`

(Opcional) Las reglas de directiva de red de Kubernetes se traducirán a reglas de firewall de NSX-T y se colocarán debajo de esta sección.

- `nsx_bottom_fw_section='bottom_section'`

(Opcional) Las reglas de directiva de red de Kubernetes se traducirán a reglas de firewall de NSX-T y se colocarán encima de esta sección.

- `nsx_api_cert='/path/to/cert/nsx.crt'`
- `nsx_api_private_key='/path/to/key/nsx.key'`

(Opcional) Si se establece esta opción, `nsx_api_user` y `nsx_api_password` se omitirán. El certificado debe cargarse en NSX-T y debe crearse manualmente un usuario de identidad de entidad de seguridad que se autentique con este certificado.

- `nsx_lb_default_cert='/path/to/cert/nsx.crt'`
- `nsx_lb_default_key='/path/to/key/nsx.key'`

(Opcional) El equilibrador de carga de NSX-T requiere un certificado predeterminado para poder crear SNI para rutas basadas en TLS. Este certificado solo se presentará si no hay ninguna ruta configurada. Si no se proporciona, se generará un certificado autofirmado.

Archivo de hosts de Ansible de ejemplo

```
[OSEv3:children]
masters
nodes
etcd

[OSEv3:vars]
ansible_ssh_user=root
openshift_deployment_type=origin

openshift_master_identity_providers=[{'name': 'htpasswd_auth', 'login': 'true', 'challenge': 'true',
'kind': 'HTPasswdPasswordIdentityProvider'}]
openshift_master_htpasswd_users={'yasen' : 'password'}

openshift_master_default_subdomain=demo.corp.local
openshift_use_nsx=true
os_sdn_network_plugin_name=cni
openshift_use_openshift_sdn=false
openshift_node_sdn_mtu=1500

# NSX specific configuration
nsx_openshift_cluster_name='ocp-cluster1'
nsx_api_managers='192.168.110.201'
nsx_api_user='admin'
nsx_api_password='VMware1!'
nsx_tier0_router='DefaultT0Router'
nsx_overlay_transport_zone='overlay-tz'
nsx_container_ip_block='ocp-pod-networking'
nsx_no_snat_ip_block='ocp-nonat-pod-networking'
nsx_external_ip_pool='ocp-external'
nsx_top_fw_section='openshift-top'
nsx_bottom_fw_section='openshift-bottom'
nsx_ovs_uplink_port='ens224'
nsx_cni_url='http://1.1.1.1/nsx-cni-2.3.2.x86_64.rpm'
nsx_ovs_url='http://1.1.1.1/openvswitch-2.9.1.rhel75-1.x86_64.rpm'
nsx_kmod_ovs_url='http://1.1.1.1/kmod-openvswitch-2.9.1.rhel75-1.el7.x86_64.rpm'

[masters]
ocp-master.corp.local
```


[etcd]

ocp-master.corp.local

[nodes]

ocp-master.corp.local ansible_ssh_host=10.1.0.10 openshift_node_group_name='node-config-master'

ocp-node1.corp.local ansible_ssh_host=10.1.0.11 openshift_node_group_name='node-config-infra'

ocp-node2.corp.local ansible_ssh_host=10.1.0.12 openshift_node_group_name='node-config-infra'

ocp-node3.corp.local ansible_ssh_host=10.1.0.13 openshift_node_group_name='node-config-compute'

ocp-node4.corp.local ansible_ssh_host=10.1.0.14 openshift_node_group_name='node-config-compute'

Equilibrio de carga

El equilibrador de carga de NSX-T Data Center está integrado con OpenShift y actúa como enrutador de OpenShift.

NCP supervisa los eventos de endpoint y la ruta de OpenShift, y configura las reglas de equilibrio de carga en el equilibrador de carga en función de la especificación de ruta. A raíz de ello, el equilibrador de carga de NSX-T Data Center reenviará el tráfico entrante de capa 7 a los pods de back-end adecuados según las reglas.

Configuración del equilibrio de carga

La configuración del equilibrio de carga conlleva configurar un servicio de equilibrador de carga de Kubernetes o una ruta de OpenShift. También se debe configurar el controlador de replicación de NCP. El servicio de equilibrador de carga corresponde al tráfico de capa 4, mientras que la ruta de OpenShift corresponde al tráfico de capa 7.

Cuando se configura un servicio de equilibrador de carga de Kubernetes, se le asigna una dirección IP del bloque de IP externo que se haya configurado. El equilibrador de carga se expone en esta dirección IP y en el puerto del servicio. Puede especificar el nombre o el identificador de un grupo de IP usando la especificación `loadBalancerIP` en la definición del equilibrador de carga. La IP del servicio de equilibrador de carga se asignará a partir de este grupo de direcciones IP. Si la especificación `loadBalancerIP` está vacía, la IP se asignará a partir del bloque de direcciones IP externo que configure.

El grupo de direcciones IP que `loadBalancerIP` especifica debe tener la etiqueta `{"ncp/owner": cluster:<cluster>}`.

Para usar el equilibrador de carga de NSX-T Data Center, hay que configurar el equilibrio de carga de NCP. En el archivo `ncp_rc.yml`, haga lo siguiente:

- 1 Establezca `use_native_loadbalancer` como `True`.
- 2 Establezca `pool_algorithm` como `WEIGHTED_ROUND_ROBIN`.
- 3 Establezca `lb_default_cert_path` y `lb_priv_key_path` como los nombres de ruta completa del archivo de certificado firmado por CA y del archivo de clave privada, respectivamente. A continuación se incluye un script de ejemplo que permite generar un certificado firmado por CA. Adicionalmente, monte la clave y el certificado predeterminados en el pod de NCP. Consulte las instrucciones más adelante.

- 4 (Opcional) Especifique una configuración de persistencia con los parámetros `l4_persistence` y `l7_persistence`. La opción disponible para la persistencia de capa 4 es la IP de origen. Las opciones disponibles para la persistencia de capa 7 son IP de origen y cookie. El valor predeterminado es `<None>`. Por ejemplo,

```
# Choice of persistence type for ingress traffic through L7 Loadbalancer.
# Accepted values:
# 'cookie'
# 'source_ip'
l7_persistence = cookie

# Choice of persistence type for ingress traffic through L4 Loadbalancer.
# Accepted values:
# 'source_ip'
l4_persistence = source_ip
```

- 5 (Opcional) Establezca `service_size` como `SMALL`, `MEDIUM` o `LARGE`. El valor predeterminado es `SMALL`.
- 6 Si ejecuta OpenShift 3.11, debe aplicar la siguiente configuración de manera que OpenShift no asigne una dirección IP al servicio de equilibrador de carga.
 - Establezca `ingressIPNetworkCIDR` como `0.0.0.0/32` en `networkConfig` en el archivo `/etc/origin/master/master-config.yaml`.
 - Reinicie los controladores y el servidor de API con los siguientes comandos:

```
master-restart api
master-restart controllers
```

Para los equilibradores de carga de Kubernetes, también puede establecer `sessionAffinity` en la especificación del servicio para configurar su comportamiento de persistencia si la persistencia global de capa 4 está desactivada (es decir, si `l4_persistence` se ha establecido como `<None>`). Si se ha establecido `l4_persistence` como `source_ip`, se puede utilizar `sessionAffinity` en la especificación de servicio para personalizar el tiempo de espera de la persistencia del servicio. El tiempo de espera predeterminado de la persistencia de capa 4 es de 10800 segundos (como se especifica en la

documentación de Kubernetes para servicios (<https://kubernetes.io/docs/concepts/services-networking/service>). Los servicios con un tiempo de espera de persistencia predeterminado compartirán el mismo perfil de persistencia del equilibrador de carga de NSX-T. Se creará un perfil dedicado para cada servicio con un tiempo de espera de persistencia no predeterminado.

Nota Si el servicio back-end de una entrada es un servicio de tipo equilibrador de carga, el servidor virtual de capa 4 para el servicio y el servidor virtual de capa 7 para la entrada no pueden tener configuraciones de persistencia diferentes (por ejemplo, `source_ip` para la capa 4 y `cookie` para la capa 7). En tal caso, la configuración de persistencia para ambos servidores virtuales debe ser la misma (`source_ip`, `cookie` o `None`) o una de ellas debe ser `None` (y la otra configuración puede ser `source_ip` o `cookie`). A continuación, se muestra un ejemplo de este escenario:

```
apiVersion: extensions/v1beta1
kind: Ingress
metadata:
  name: cafe-ingress
spec:
  rules:
  - host: cafe.example.com
    http:
      paths:
      - path: /tea
        backend:
          serviceName: tea-svc
          servicePort: 80

-----
apiVersion: v1
kind: Service
metadata:
  name: tea-svc <==== same as the Ingress backend above
  labels:
    app: tea
spec:
  ports:
  - port: 80
    targetPort: 80
    protocol: TCP
    name: tcp
  selector:
    app: tea
  type: LoadBalancer
```

Particionamiento de enrutadores

NCP siempre procesa la terminación de Edge de TLS y las rutas HTTP, y omite las rutas de acceso directo y las rutas de recifrado de TLS, independientemente de sus espacios de nombres o sus etiquetas de espacios de nombres. Para restringir un enrutador de OpenShift de modo que solo procese las rutas de recifrado y de acceso directo de TLS, debe realizar los siguientes pasos:

- Agregue un selector de etiquetas de espacio de nombres al enrutador de OpenShift.

- Agregue una etiqueta de espacio de nombres al espacio de nombres de destino.
- Cree rutas de acceso directo o de recifrado de TLS en el espacio de nombres de destino.

Por ejemplo, para configurar un enrutador con un selector de etiquetas de espacio de nombres, ejecute el siguiente comando (se asume que el nombre de la cuenta de servicio del enrutador es router):

```
oc set env dc/router NAMESPACE_LABELS="router=r1"
```

El enrutador ahora procesará las rutas de los espacios de nombres seleccionados. Para hacer que este selector coincida con un espacio de nombres, ejecute el siguiente comando (se asume que el espacio de nombres se denomina ns1):

```
oc label namespace ns1 "router=r1"
```

Ejemplo de equilibrador de carga de capa 7

Con el siguiente archivo YAML se configuran dos controladores de replicación (tea-rc y coffee-rc), dos servicios (tea-svc y coffee-svc) y dos rutas (cafe-route-multi y cafe-route) para proporcionar equilibrio de carga de capa 7.

```
# RC
apiVersion: v1
kind: ReplicationController
metadata:
  name: tea-rc
spec:
  replicas: 2
  template:
    metadata:
      labels:
        app: tea
    spec:
      containers:
        - name: tea
          image: nginxdemos/hello
          imagePullPolicy: IfNotPresent
          ports:
            - containerPort: 80
---
apiVersion: v1
kind: ReplicationController
metadata:
  name: coffee-rc
spec:
  replicas: 2
  template:
    metadata:
      labels:
        app: coffee
    spec:
      containers:
        - name: coffee
          image: nginxdemos/hello
```

```

        imagePullPolicy: IfNotPresent
        ports:
          - containerPort: 80
---
# Services
apiVersion: v1
kind: Service
metadata:
  name: tea-svc
  labels:
    app: tea
spec:
  ports:
    - port: 80
      targetPort: 80
      protocol: TCP
      name: http
  selector:
    app: tea
---
apiVersion: v1
kind: Service
metadata:
  name: coffee-svc
  labels:
    app: coffee
spec:
  ports:
    - port: 80
      targetPort: 80
      protocol: TCP
      name: http
  selector:
    app: coffee
---
# Routes
apiVersion: v1
kind: Route
metadata:
  name: cafe-route-multi
spec:
  host: www.cafe.com
  path: /drinks
  to:
    kind: Service
    name: tea-svc
    weight: 1
  alternateBackends:
    - kind: Service
      name: coffee-svc
      weight: 2
---
apiVersion: v1
kind: Route
metadata:

```

```
name: cafe-route
spec:
  host: www.cafe.com
  path: /tea-svc
  to:
    kind: Service
    name: tea-svc
    weight: 1
```

Notas adicionales

- El tráfico HTTPS solo admite la terminación de Edge.
- Se admiten subdominios comodín. Por ejemplo, si `wildcardPolicy` está establecido como **Subdomain** y el nombre de host se establece como **wildcard.example.com**, se atenderá cualquier solicitud a ***.ejemplo.com**.
- Si NCP genera un error al procesar un evento de ruta debido a una configuración errónea, será necesario corregir el archivo YAML de la ruta y, a continuación, eliminar y volver a crear el recurso de ruta.
- NCP no aplica la propiedad de nombre de host por espacios de nombres.
- Se admite un servicio de equilibrador de carga por cada clúster de Kubernetes.
- NSX-T Data Center creará un grupo y un servidor virtual de equilibrador de carga de capa 4 por cada puerto de servicio de equilibrador de carga. Se admiten los protocolos TCP y UDP.
- El equilibrador de carga de NSX-T Data Center viene en diferentes tamaños. Para obtener información sobre la configuración de un equilibrador de carga de NSX-T Data Center, consulte la *Guía de administración de NSX-T Data Center*.

Después de que se crea el equilibrador de carga, no es posible cambiar su tamaño actualizando el archivo de configuración. Se puede cambiar mediante la interfaz de usuario o la API.

- Se admite el ajuste de escala automático del equilibrador de carga de capa 4. Si se crea o se modifica un servicio de equilibrador de carga de Kubernetes de manera que requiera servidores virtuales adicionales y el equilibrador de carga de capa 4 existente no tiene la capacidad, se creará un nuevo equilibrador de carga de capa 4. NCP también eliminará un equilibrador de carga de capa 4 que ya no tenga servidores virtuales asociados. Esta función está habilitada de forma predeterminada. Se puede deshabilitar estableciendo `l4_lb_auto_scaling` como **false** en ConfigMap de NCP.

Script de ejemplo para generar un certificado firmado por CA

Con el siguiente script se genera un certificado firmado por CA y una clave privada que se almacenan en los archivos <nombredearchivo>.crt y <nombredearchivo>.key respectivamente. El comando `genrsa` genera una clave de CA. La clave de CA debe estar cifrada. Se puede especificar un método de cifrado con el comando (p. ej., `aes256`).

```
#!/bin/bash
host="www.example.com"
filename=server

openssl genrsa -out ca.key 4096
openssl req -key ca.key -new -x509 -days 365 -sha256 -extensions v3_ca -out ca.crt -subj
"/C=US/ST=CA/L=Palo Alto/O=OS3/OU=Eng/CN=${host}"
openssl req -out ${filename}.csr -new -newkey rsa:2048 -nodes -keyout ${filename}.key -subj
"/C=US/ST=CA/L=Palo Alto/O=OS3/OU=Eng/CN=${host}"
openssl x509 -req -days 360 -in ${filename}.csr -CA ca.crt -CAkey ca.key -CAcreateserial -out $
{filename}.crt -sha256
```

Montar la clave y el certificado predeterminados en el pod de NCP

Tras generar la clave privada y el certificado, colóquelos en el directorio `/etc/nsx-ujo` en la máquina virtual del host. Suponiendo que los archivos de clave y de certificado se llaman `lb-default.crt` y `lb-default.key` respectivamente, edite `ncp-rc.yaml` para que estos archivos en el host se monten en el pod. Por ejemplo,

```
spec:
  ...
  containers:
  - name: nsx-ncp
    ...
    volumeMounts:
    ...
    - name: lb-default-cert
      # Mount path must match nsx_v3 option "lb_default_cert_path"
      mountPath: /etc/nsx-ujo/lb-default.crt
    - name: lb-priv-key
      # Mount path must match nsx_v3 option "lb_priv_key_path"
      mountPath: /etc/nsx-ujo/lb-default.key
  volumes:
  ...
  - name: lb-default-cert
    hostPath:
      path: /etc/nsx-ujo/lb-default.crt
  - name: lb-priv-key
    hostPath:
      path: /etc/nsx-ujo/lb-default.key
```


Administrar NSX Container Plug-in

5

Puede administrar NSX Container Plug-in desde la GUI de NSX Manager o desde la interfaz de la línea de comandos (command-line interface, CLI).

Nota Si una máquina virtual de host del contenedor se ejecuta en ESXi 6.5 y se migra mediante vMotion a otro host ESXi 6.5, los contenedores que se ejecuten en el host del contenedor perderán la conectividad con los contenedores que se ejecuten en otros hosts del contenedor. Para resolver el problema, desconecte y conecte la vNIC del host del contenedor. Este problema no ocurre con ESXi 6.5 Update 1 o versiones posteriores.

HyperBus reserva el identificador de VLAN 4094 en el hipervisor para la configuración de PVLAN. Este identificador no se puede cambiar. Para evitar conflictos de VLAN, no configure los conmutadores lógicos de VLAN ni las vmknics de VTEP con el mismo identificador de VLAN.

Este capítulo incluye los siguientes temas:

- [Administrar los bloques de IP desde la GUI de NSX Manager](#)
- [Ver subredes de bloques de IP desde la GUI de NSX Manager](#)
- [Puertos lógicos conectados a CIF](#)
- [Comandos de la CLI](#)
- [Códigos de error](#)

Administrar los bloques de IP desde la GUI de NSX Manager

Puede agregar, eliminar, editar y administrar las etiquetas de un bloque de IP, así como consultar su información, en la GUI de NSX Manager.

Procedimiento

- 1 En un explorador, inicie sesión en NSX Manager en `https://<dirección-ip-o-nombre-dominio-nsx-manager>`.
- 2 Desplácese hasta **Redes > IPAM**.
Aparece una lista de bloques de IP.

3 Realice una de las siguientes acciones.

Opción	Acción
Agregar un bloque de IP	Haga clic en AGREGAR .
Eliminar uno o varios bloques de IP	Seleccione uno o varios bloques de IP y haga clic en ELIMINAR .
Editar un bloque de IP	Seleccione un bloque de IP y haga clic en EDITAR .
Ver información de un bloque de IP	Haga clic en el nombre del bloque de IP. Haga clic en la pestaña Información general para consultar la información general. Haga clic en la pestaña Subredes para consultar las subredes de este bloque de IP.
Administrar etiquetas de un bloque de IP	Seleccione un bloque de IP y haga clic en ACCIONES > Administrar etiquetas .

No puede eliminar un bloque de IP que tenga subredes asignadas.

Ver subredes de bloques de IP desde la GUI de NSX Manager

Puede ver las subredes de un bloque de IP desde la GUI de NSX Manager. No se recomienda agregar o eliminar subredes de bloques de IP cuando NCP está instalado y en ejecución.

Procedimiento

- 1 En un explorador, inicie sesión en NSX Manager en <https://<dirección-ip-o-nombre-dominio-nsx-manager>>.
- 2 Desplácese hasta **Redes > IPAM**.
Aparece una lista de bloques de IP.
- 3 Haga clic en un nombre de un bloque de IP.
- 4 Haga clic en la pestaña **Subredes**.

Puertos lógicos conectados a CIF

Las CIF (interfaces de los contenedores) son interfaces de red de contenedores conectados a los puertos lógicos de un conmutador. Estos puertos se denominan puertos lógicos conectados a CIF.

Puede administrar los puertos lógicos conectados a CIF desde la GUI de NSX Manager.

Administrar los puertos lógicos conectados a CIF

Desplácese hasta **Redes > Conmutación > Puertos** para ver todos los puertos lógicos, incluidos los puertos lógicos conectados a CIF. Haga clic en el vínculo de conexión de un puerto lógico conectado a CIF para ver la información de la conexión. Haga clic en el vínculo del puerto lógico para abrir un panel de ventana con cuatro pestañas: Información general, Supervisor, Administrar y Relacionado. Al hacer clic en **Relacionado > Puertos lógicos**, aparecen los puertos lógicos relacionados de un conmutador de enlace ascendente. Para obtener más información sobre los puertos de conmutación, consulte la *Guía de administración de NSX-T*.

Herramientas para supervisar la red

Las siguientes herramientas admiten puertos lógicos con conexión a CIF. Para obtener más información sobre estas herramientas, consulte la *Guía de administración de NSX-T*.

- Traceflow
- Conexión de puertos
- IPFIX
- Se admite la creación de reflejo de puertos remotos con la encapsulación GRE de un puerto de conmutador lógico que se conecta a un contenedor. Para obtener más información, consulte el apartado "Información sobre el perfil de conmutación de creación de reflejo del puerto" de la *Guía de administración de NSX-T*. Sin embargo, la creación de reflejo de puerto del puerto CIF a VIF no se admite mediante la interfaz de usuario de administrador.

Comandos de la CLI

Para ejecutar comandos de la CLI, inicie sesión en el contenedor de NSX Container Plug-in, abra un terminal y ejecute el comando `nsxcli`.

También puede obtener avisos de la CLI ejecutando el siguiente comando en un nodo:

```
kubectl exec -it <pod name> nsxcli
```

Tabla 5-1. Comandos de la CLI para el contenedor de NCP

Tipo	Comando
Estado	<code>get ncp-master status</code>
Estado	<code>get ncp-nsx status</code>
Estado	<code>get ncp-watcher <nombre-monitor></code>
Estado	<code>get ncp-watchers</code>
Estado	<code>get ncp-k8s-api-server status</code>
Estado	<code>check projects</code>
Estado	<code>check project <nombre-proyecto></code>
Caché	<code>get project-cache <nombre-proyecto></code>
Caché	<code>get project-caches</code>
Caché	<code>get namespace-cache <nombre-espaciodenombres></code>
Caché	<code>get namespace-caches</code>
Caché	<code>get pod-cache <nombre-pod></code>
Caché	<code>get pod-caches</code>
Caché	<code>get ingress-caches</code>
Caché	<code>get ingress-cache <ingress-name></code>

Tabla 5-1. Comandos de la CLI para el contenedor de NCP (Continuación)

Tipo	Comando
Caché	get ingress-controllers
Caché	get ingress-controller <nombre-controlador-entrada>
Caché	get network-policy-caches
Caché	get network-policy-cache <nombre-pod>
Soporte técnico	get ncp-log file <nombredearchivo>
Soporte técnico	get ncp-log-level
Soporte técnico	set ncp-log-level <nivel-registro>
Soporte técnico	get support-bundle file <nombredearchivo>
Soporte técnico	get node-agent-log file <nombredearchivo>
Soporte técnico	get node-agent-log file <nombredearchivo> <nombre-nodo>

Tabla 5-2. Comandos de la CLI para el contenedor de agentes del nodo de NSX

Tipo	Comando
Estado	get node-agent-hyperbus status
Caché	get container-cache <nombre-contenedor>
Caché	get container-caches

Tabla 5-3. Comandos de la CLI para el contenedor de Kube Proxy de NSX

Tipo	Comando
Estado	get ncp-k8s-api-server status
Estado	get kube-proxy-watcher <nombre-monitor>
Estado	get kube-proxy-watchers
Estado	dump ovs-flows

Comandos de estado para el contenedor de NCP

- Mostrar el estado del maestro de NCP

```
get ncp-master status
```

Ejemplo:

```
kubenode> get ncp-master status
This instance is not the NCP master
Current NCP Master id is a4h83eh1-b8dd-4e74-c71c-cbb7cc9c4c1c
Last master update at Wed Oct 25 22:46:40 2017
```

- Mostrar el estado de la conexión entre NCP y NSX Manager

```
get ncp-nsx status
```

Ejemplo:

```
kubecall> get ncp-nsx status
NSX Manager status: Healthy
```

- Mostrar el estado del monitor para la entrada, el espacio de nombres, el pod y el servicio

```
get ncp-watcher <watcher-name>
get ncp-watchers
```

Ejemplo 1:

```
kubecall> get ncp-watcher pod
Average event processing time: 1174 msec (in past 3600-sec window)
Current watcher started time: Mar 02 2017 10:47:35 PST
Number of events processed: 1 (in past 3600-sec window)
Total events processed by current watcher: 1
Total events processed since watcher thread created: 1
Total watcher recycle count: 0
Watcher thread created time: Mar 02 2017 10:47:35 PST
Watcher thread status: Up
```

Ejemplo 2:

```
kubecall> get ncp-watchers
pod:
Average event processing time: 1145 msec (in past 3600-sec window)
Current watcher started time: Mar 02 2017 10:51:37 PST
Number of events processed: 1 (in past 3600-sec window)
Total events processed by current watcher: 1
Total events processed since watcher thread created: 1
Total watcher recycle count: 0
Watcher thread created time: Mar 02 2017 10:51:37 PST
Watcher thread status: Up

namespace:
Average event processing time: 68 msec (in past 3600-sec window)
Current watcher started time: Mar 02 2017 10:51:37 PST
Number of events processed: 2 (in past 3600-sec window)
Total events processed by current watcher: 2
Total events processed since watcher thread created: 2
Total watcher recycle count: 0
Watcher thread created time: Mar 02 2017 10:51:37 PST
Watcher thread status: Up

ingress:
Average event processing time: 0 msec (in past 3600-sec window)
Current watcher started time: Mar 02 2017 10:51:37 PST
```

```

Number of events processed: 0 (in past 3600-sec window)
Total events processed by current watcher: 0
Total events processed since watcher thread created: 0
Total watcher recycle count: 0
Watcher thread created time: Mar 02 2017 10:51:37 PST
Watcher thread status: Up

```

service:

```

Average event processing time: 3 msec (in past 3600-sec window)
Current watcher started time: Mar 02 2017 10:51:37 PST
Number of events processed: 1 (in past 3600-sec window)
Total events processed by current watcher: 1
Total events processed since watcher thread created: 1
Total watcher recycle count: 0
Watcher thread created time: Mar 02 2017 10:51:37 PST
Watcher thread status: Up

```

- Mostrar el estado de conexión entre el servidor de API de NCP y de Kubernetes

```
get ncp-k8s-api-server status
```

Ejemplo:

```

kubenode> get ncp-k8s-api-server status
Kubernetes ApiServer status: Healthy

```

- Comprobar todos los proyectos o uno específico

```

check projects
check project <project-name>

```

Ejemplo:

```

kubenode> check projects
default:
  Tier-1 link port for router 1b90a61f-0f2c-4768-9eb6-ea8954b4f327 is missing
  Switch 40a6829d-c3aa-4e17-ae8a-7f7910fdf2c6 is missing

ns1:
  Router 8accc9cd-9883-45f6-81b3-0d1fb2583180 is missing

kubenode> check project default
Tier-1 link port for router 1b90a61f-0f2c-4768-9eb6-ea8954b4f327 is missing
Switch 40a6829d-c3aa-4e17-ae8a-7f7910fdf2c6 is missing

```

Comandos de caché para el contenedor de NCP

- Obtener la caché interna para los proyectos o los espacios de nombres

```
get project-cache <project-name>
get project-caches
get namespace-cache <namespace-name>
get namespace-caches
```

Ejemplo:

```
kubecall> get project-caches
default:
  logical-router: 8accc9cd-9883-45f6-81b3-0d1fb2583180
  logical-switch:
    id: 9d7da647-27b6-47cf-9cdb-6e4f4d5a356d
    ip_pool_id: 519ff57f-061f-4009-8d92-3e6526e7c17e
    subnet: 10.0.0.0/24
    subnet_id: f75fd64c-c7b0-4b42-9681-fc656ae5e435

kube-system:
  logical-router: 5032b299-acad-448e-a521-19d272a08c46
  logical-switch:
    id: 85233651-602d-445d-ab10-1c84096cc22a
    ip_pool_id: ab1c5b09-7004-4206-ac56-85d9d94bffa2
    subnet: 10.0.1.0/24
    subnet_id: 73e450af-b4b8-4a61-a6e3-c7ddd15ce751

testns:
  ext_pool_id: 346a0f36-7b5a-4ecc-ad32-338dcb92316f
  labels:
    ns: myns
    project: myproject
  logical-router: 4dc8f8a9-69b4-4ff7-8fb7-d2625dc77efa
  logical-switch:
    id: 6111a99a-6e06-4faa-a131-649f10f7c815
    ip_pool_id: 51ca058d-c3dc-41fd-8f2d-e69006ab1b3d
    subnet: 50.0.2.0/24
    subnet_id: 34f79811-bd29-4048-a67d-67ceac97eb98
  project_nsgroup: 9606afee-6348-4780-9dbe-91abfd23e475
  snat_ip: 4.4.0.3

kubecall> get project-cache default
logical-router: 8accc9cd-9883-45f6-81b3-0d1fb2583180
logical-switch:
  id: 9d7da647-27b6-47cf-9cdb-6e4f4d5a356d
  ip_pool_id: 519ff57f-061f-4009-8d92-3e6526e7c17e
  subnet: 10.0.0.0/24
  subnet_id: f75fd64c-c7b0-4b42-9681-fc656ae5e435

kubecall> get namespace-caches
default:
  logical-router: 8accc9cd-9883-45f6-81b3-0d1fb2583180
```

```

logical-switch:
  id: 9d7da647-27b6-47cf-9cdb-6e4f4d5a356d
  ip_pool_id: 519ff57f-061f-4009-8d92-3e6526e7c17e
  subnet: 10.0.0.0/24
  subnet_id: f75fd64c-c7b0-4b42-9681-fc656ae5e435

kube-system:
  logical-router: 5032b299-acad-448e-a521-19d272a08c46
  logical-switch:
    id: 85233651-602d-445d-ab10-1c84096cc22a
    ip_pool_id: ab1c5b09-7004-4206-ac56-85d9d94bffa2
    subnet: 10.0.1.0/24
    subnet_id: 73e450af-b4b8-4a61-a6e3-c7ddd15ce751

testns:
  ext_pool_id: 346a0f36-7b5a-4ecc-ad32-338dcb92316f
  labels:
    ns: myns
    project: myproject
  logical-router: 4dc8f8a9-69b4-4ff7-8fb7-d2625dc77efa
  logical-switch:
    id: 6111a99a-6e06-4faa-a131-649f10f7c815
    ip_pool_id: 51ca058d-c3dc-41fd-8f2d-e69006ab1b3d
    subnet: 50.0.2.0/24
    subnet_id: 34f79811-bd29-4048-a67d-67ceac97eb98
  project_nsgroup: 9606afee-6348-4780-9dbe-91abfd23e475
  snat_ip: 4.4.0.3

kubenode> get namespace-cache default
logical-router: 8accc9cd-9883-45f6-81b3-0d1fb2583180
logical-switch:
  id: 9d7da647-27b6-47cf-9cdb-6e4f4d5a356d
  ip_pool_id: 519ff57f-061f-4009-8d92-3e6526e7c17e
  subnet: 10.0.0.0/24
  subnet_id: f75fd64c-c7b0-4b42-9681-fc656ae5e435

```

■ Obtener la caché interna para los pods

```

get pod-cache <pod-name>
get pod-caches

```

Ejemplo:

```

kubenode> get pod-caches
nsx.default.nginx-rc-uq2lv:
  cif_id: 2af9f734-37b1-4072-ba88-abbf935bf3d4
  gateway_ip: 10.0.0.1
  host_vif: d6210773-5c07-4817-98db-451bd1f01937
  id: 1c8b5c52-3795-11e8-ab42-005056b198fb
  ingress_controller: False
  ip: 10.0.0.2/24
  labels:
    app: nginx
  mac: 02:50:56:00:08:00

```



```

port_id: d52c833a-f531-4bdf-bfa2-e8a084a8d41b
vlan: 1

nsx.testns.web-pod-1:
  cif_id: ce134f21-6be5-43fe-afbf-aaca8c06b5cf
  gateway_ip: 50.0.2.1
  host_vif: d6210773-5c07-4817-98db-451bd1f01937
  id: 3180b521-270e-11e8-ab42-005056b198fb
  ingress_controller: False
  ip: 50.0.2.3/24
  labels:
    app: nginx-new
    role: db
    tier: cache
  mac: 02:50:56:00:20:02
  port_id: 81bc2b8e-d902-4cad-9fc1-aabdc32ecaf8
  vlan: 3

kubens> get pod-cache nsx.default.nginx-rc-uj2lv
  cif_id: 2af9f734-37b1-4072-ba88-abbf935bf3d4
  gateway_ip: 10.0.0.1
  host_vif: d6210773-5c07-4817-98db-451bd1f01937
  id: 1c8b5c52-3795-11e8-ab42-005056b198fb
  ingress_controller: False
  ip: 10.0.0.2/24
  labels:
    app: nginx
  mac: 02:50:56:00:08:00
  port_id: d52c833a-f531-4bdf-bfa2-e8a084a8d41b
  vlan: 1

```

■ Obtener cachés de directiva de red o una específica

```

get network-policy caches
get network-policy-cache <network-policy-name>

```

Ejemplo:

```

kubens> get network-policy-caches
nsx.testns.allow-tcp-80:
  dest_labels: None
  dest_pods:
    50.0.2.3
  match_expressions:
    key: tier
    operator: In
    values:
      cache
  name: allow-tcp-80
  np_dest_ip_set_ids:
    22f82d76-004f-4d12-9504-ce1cb9c8aa00
  np_except_ip_set_ids:
  np_ip_set_ids:
    14f7f825-f1a0-408f-bbd9-bb2f75d44666

```

```

np_isol_section_id: c8d93597-9066-42e3-991c-c550c46b2270
np_section_id: 04693136-7925-44f2-8616-d809d02cd2a9
ns_name: testns
src_egress_rules: None
src_egress_rules_hash: 97d170e1550eee4afc0af065b78cda302a97674c
src_pods:
  50.0.2.0/24
src_rules:
  from:
    namespaceSelector:
      matchExpressions:
        key: tier
        operator: DoesNotExist
      matchLabels:
        ns: myns
    ports:
      port: 80
      protocol: TCP
src_rules_hash: e4ea7b8d91c1e722670a59f971f8fcc1a5ac51f1

```

```

kubenset> get network-policy-cache nsx.testns.allow-tcp-80
dest_labels: None
dest_pods:
  50.0.2.3
match_expressions:
  key: tier
  operator: In
  values:
    cache
name: allow-tcp-80
np_dest_ip_set_ids:
  22f82d76-004f-4d12-9504-ce1cb9c8aa00
np_except_ip_set_ids:
np_ip_set_ids:
  14f7f825-f1a0-408f-bbd9-bb2f75d44666
np_isol_section_id: c8d93597-9066-42e3-991c-c550c46b2270
np_section_id: 04693136-7925-44f2-8616-d809d02cd2a9
ns_name: testns
src_egress_rules: None
src_egress_rules_hash: 97d170e1550eee4afc0af065b78cda302a97674c
src_pods:
  50.0.2.0/24
src_rules:
  from:
    namespaceSelector:
      matchExpressions:
        key: tier
        operator: DoesNotExist
      matchLabels:
        ns: myns

```

```
ports:
  port: 80
  protocol: TCP
src_rules_hash: e4ea7b8d91c1e722670a59f971f8fcc1a5ac51f1
```

Comandos de soporte para el contenedor de NCP

- Guardar el paquete de soporte técnico de NCP en el almacén de archivos

El paquete de soporte técnico consta de los archivos de registro de todos los contenedores de los pods con la etiqueta **tier:nsx-networking**. El archivo de paquete tiene formato tgz y se guarda en el directorio del almacén de archivos predeterminado de la CLI `/var/vmware/nsx/file-store`. Puede utilizar el comando del almacén de archivos de la CLI para copiar el archivo de paquete a un sitio remoto.

```
get support-bundle file <filename>
```

Ejemplo:

```
kubenode>get support-bundle file foo
Bundle file foo created in tgz format
kubenode>copy file foo url scp://nicira@10.0.0.1:/tmp
```

- Guardar los registros de NCP en el almacén de archivos

El archivo de registro se guarda en formato tgz y en el directorio del almacén de archivos predeterminado de la CLI `/var/vmware/nsx/file-store`. Puede utilizar el comando del almacén de archivos de la CLI para copiar el archivo de paquete a un sitio remoto.

```
get ncp-log file <filename>
```

Ejemplo:

```
kubenode>get ncp-log file foo
Log file foo created in tgz format
```

- Guardar los registros del agente del nodo en el almacén de archivos

Guarde los registros del agente del nodo de uno nodo o de todos. Los registros se guardan en formato tgz y en el directorio del almacén de archivos predeterminado de la CLI `/var/vmware/nsx/file-store`. Puede utilizar el comando del almacén de archivos de la CLI para copiar el archivo de paquete a un sitio remoto.

```
get node-agent-log file <filename>
get node-agent-log file <filename> <node-name>
```

Ejemplo:

```
kubecode>get node-agent-log file foo
Log file foo created in tgz format
```

- Obtener y establecer el nivel de registro

Los niveles de registro disponibles son NOTSET, DEBUG, INFO, WARNING, ERROR y CRITICAL.

```
get ncp-log-level
set ncp-log-level <log level>
```

Ejemplo:

```
kubecode>get ncp-log-level
NCP log level is INFO

kubecode>set ncp-log-level DEBUG
NCP log level is changed to DEBUG
```

Comandos de estado para el contenedor de agentes del nodo de NSX

- Mostrar el estado de conexión entre HyperBus y el agente de este nodo.

```
get node-agent-hyperbus status
```

Ejemplo:

```
kubecode> get node-agent-hyperbus status
HyperBus status: Healthy
```

Comandos de caché para el contenedor de agentes del nodo de NSX

- Obtener la caché interna para contenedores de agentes del nodo de NSX.

```
get container-cache <container-name>
get container-caches
```

Ejemplo 1:

```
kubecode> get container-cache cif104
ip: 192.168.0.14/32
mac: 50:01:01:01:01:14
gateway_ip: 169.254.1.254/16
vlan_id: 104
```

Ejemplo 2:

```
kubecfg> get container-caches
cif104:
  ip: 192.168.0.14/32
  mac: 50:01:01:01:01:14
  gateway_ip: 169.254.1.254/16
  vlan_id: 104
```

Comandos de estado para el contenedor del proxy de NSX Kube

- Mostrar el estado de conexión entre el servidor de Kube Proxy y de Kubernetes

```
get ncp-k8s-api-server status
```

Ejemplo:

```
kubecfg> get kube-proxy-k8s-api-server status
Kubernetes ApiServer status: Healthy
```

- Mostrar el estado del monitor de Kube Proxy

```
get kube-proxy-watcher <watcher-name>
get kube-proxy-watchers
```

Ejemplo 1:

```
kubecfg> get kube-proxy-watcher endpoint
Average event processing time: 15 msec (in past 3600-sec window)
Current watcher started time: May 01 2017 15:06:24 PDT
Number of events processed: 90 (in past 3600-sec window)
Total events processed by current watcher: 90
Total events processed since watcher thread created: 90
Total watcher recycle count: 0
Watcher thread created time: May 01 2017 15:06:24 PDT
Watcher thread status: Up
```

Ejemplo 2:

```
kubecfg> get kube-proxy-watchers
endpoint:
  Average event processing time: 15 msec (in past 3600-sec window)
  Current watcher started time: May 01 2017 15:06:24 PDT
  Number of events processed: 90 (in past 3600-sec window)
  Total events processed by current watcher: 90
  Total events processed since watcher thread created: 90
  Total watcher recycle count: 0
  Watcher thread created time: May 01 2017 15:06:24 PDT
  Watcher thread status: Up

service:
```

```

Average event processing time: 8 msec (in past 3600-sec window)
Current watcher started time: May 01 2017 15:06:24 PDT
Number of events processed: 2 (in past 3600-sec window)
Total events processed by current watcher: 2
Total events processed since watcher thread created: 2
Total watcher recycle count: 0
Watcher thread created time: May 01 2017 15:06:24 PDT
Watcher thread status: Up

```

■ Volcar los flujos OVS en un nodo

```
dump ovs-flows
```

Ejemplo:

```

kubenode> dump ovs-flows
NXST_FLOW reply (xid=0x4):
  cookie=0x0, duration=8.876s, table=0, n_packets=0, n_bytes=0, idle_age=8, priority=100,ip
  actions=ct(table=1)
    cookie=0x0, duration=8.898s, table=0, n_packets=0, n_bytes=0, idle_age=8, priority=0
    actions=NORMAL
      cookie=0x0, duration=8.759s, table=1, n_packets=0, n_bytes=0, idle_age=8,
      priority=100,tcp,nw_dst=10.96.0.1,tp_dst=443 actions=mod_tp_dst:443
        cookie=0x0, duration=8.719s, table=1, n_packets=0, n_bytes=0, idle_age=8,
        priority=100,ip,nw_dst=10.96.0.10 actions=drop
          cookie=0x0, duration=8.819s, table=1, n_packets=0, n_bytes=0, idle_age=8,
          priority=90,ip,in_port=1 actions=ct(table=2,nat)
            cookie=0x0, duration=8.799s, table=1, n_packets=0, n_bytes=0, idle_age=8, priority=80,ip
            actions=NORMAL
              cookie=0x0, duration=8.856s, table=2, n_packets=0, n_bytes=0, idle_age=8, actions=NORMAL

```

Códigos de error

En esta sección se muestran los códigos de error que generan los diferentes componentes.

Códigos de error de NCP

Código de error	Descripción
NCP00001	Configuración no válida
NCP00002	Error en la inicialización
NCP00003	Estado no válido
NCP00004	Adaptador no válido
NCP00005	Certificado no encontrado
NCP00006	Token no encontrado
NCP00007	Configuración de NSX no válida
NCP00008	Etiqueta de NSX no válida
NCP00009	Error en la conexión de NSX

Código de error	Descripción
NCP00010	Etiqueta de nodo no encontrada
NCP00011	Puerto de conmutador lógico de nodo no válido
NCP00012	Error en la actualización de VIF principal
NCP00013	VLAN agotada
NCP00014	Error en la liberación de VLAN
NCP00015	Grupo de direcciones IP agotado
NCP00016	Error en la liberación de IP
NCP00017	Bloque de direcciones IP agotado
NCP00018	Error en la creación de la subred de direcciones IP
NCP00019	Error en la eliminación de la subred de direcciones IP
NCP00020	Error en la creación del grupo de direcciones IP
NCP00021	Error en la eliminación del grupo de direcciones IP
NCP00022	Error en la creación del enrutador lógico
NCP00023	Error en la actualización del enrutador lógico
NCP00024	Error en la eliminación del enrutador lógico
NCP00025	Error en la creación del conmutador lógico

Código de error	Descripción
NCP00026	Error en la actualización del conmutador lógico
NCP00027	Error en la eliminación del conmutador lógico
NCP00028	Error en la creación del puerto de enrutador lógico
NCP00029	Error en la eliminación del puerto de enrutador lógico
NCP00030	Error en la creación del puerto de conmutador lógico
NCP00031	Error en la actualización del puerto de conmutador lógico
NCP00032	Error en la eliminación del puerto de conmutador lógico
NCP00033	Directiva de red no encontrada
NCP00034	Error en la creación del firewall
NCP00035	Error en la lectura del firewall
NCP00036	Error en la actualización del firewall
NCP00037	Error en la eliminación del firewall
NCP00038	Varias instancias de firewall encontradas
NCP00039	Error en la creación del grupo NSGroup
NCP00040	Error en la eliminación del grupo NSGroup
NCP00041	Error en la creación del conjunto de direcciones IP
NCP00042	Error en la actualización del conjunto de direcciones IP

Código de error	Descripción
NCP00043	Error en la eliminación del conjunto de direcciones IP
NCP00044	Error en la creación de reglas SNAT
NCP00045	Error en la eliminación de reglas SNAT
NCP00046	Error en la conexión de la API de adaptador
NCP00047	Excepción de monitor de adaptador
NCP00048	Error en la eliminación del servicio de equilibrador de carga
NCP00049	Error en la creación del servidor virtual del equilibrador de carga
NCP00050	Error en la actualización del servidor virtual del equilibrador de carga

Código de error	Descripción
NCP00051	Error en la eliminación del servidor virtual del equilibrador de carga
NCP00052	Error en la creación del grupo de equilibradores de carga
NCP00053	Error en la actualización del grupo de equilibradores de carga
NCP00054	Error en la eliminación del grupo de equilibradores de carga
NCP00055	Error en la creación de la regla de equilibrador de carga
NCP00056	Error en la actualización de la regla de equilibrador de carga
NCP00057	Error en la eliminación de la regla de equilibrador de carga
NCP00058	Error en la liberación de IP del grupo de equilibradores de carga
NCP00059	Asociación entre el servicio y el servidor virtual del equilibrador de carga no encontrada
NCP00060	Error en la actualización del grupo NSGroup
NCP00061	Error en la obtención de las reglas de firewall
NCP00062	Grupo NSGroup sin criterios
NCP00063	Máquina virtual de nodo no encontrada
NCP00064	VIF de nodo no encontrado
NCP00065	Error en la importación del certificado
NCP00066	Error en la anulación de la importación del certificado
NCP00067	Error en la actualización del enlace de SSL
NCP00068	Perfil de SSL no encontrado
NCP00069	Grupo de direcciones IP no encontrado
NCP00070	Clúster de Edge T0 no encontrado
NCP00071	Error en la actualización del grupo de direcciones IP
NCP00072	Error en el distribuidor
NCP00073	Error en la eliminación de reglas NAT
NCP00074	Error en la obtención del puerto de enrutador lógico
NCP00075	Error en la validación de la configuración de NSX

Código de error	Descripción
NCP00076	Error en la actualización de reglas SNAT
NCP00077	Regla SNAT superpuesta
NCP00078	Error en la adición de endpoints de equilibrador de carga
NCP00079	Error en la actualización de endpoints de equilibrador de carga
NCP00080	Error en la creación del grupo de reglas de equilibrador de carga
NCP00081	Servidor virtual del equilibrador de carga no encontrado
NCP00082	Error en la lectura del conjunto de direcciones IP
NCP00083	Error en la obtención del grupo SNAT
NCP00084	Error en la creación del servicio de equilibrador de carga
NCP00085	Error en la actualización del servicio de equilibrador de carga
NCP00086	Error en la actualización del puerto de enrutador lógico
NCP00087	Error en la inicialización del equilibrador de carga
NCP00088	El grupo de direcciones IP no es único
NCP00089	Error en la sincronización de la memoria caché del equilibrador de carga de capa 7
NCP00090	Error relativo a la inexistencia del grupo de equilibradores de carga
NCP00091	Error en la inicialización de la memoria caché de la regla de equilibrador de carga
NCP00092	Error en el proceso SNAT
NCP00093	Error en el certificado predeterminado de equilibrador de carga
NCP00094	Error en la eliminación del endpoint de equilibrador de carga
NCP00095	Proyecto no encontrado
NCP00096	Acceso denegado al grupo
NCP00097	No se pudo obtener un servicio de equilibrador de carga
NCP00098	No se pudo crear un servicio de equilibrador de carga
NCP00099	Error en la sincronización de la memoria caché del grupo de equilibradores de carga

Códigos de error del agente de nodo de NSX

Código de error	Descripción
NCP01001	Vínculo superior de OVS no encontrado
NCP01002	Dirección MAC de host no encontrada
NCP01003	Error en la creación del puerto de OVS
NCP01004	Sin configuración de pod
NCP01005	Error en la configuración de pod
NCP01006	Error en la anulación de la configuración de pod
NCP01007	Socket de CNI no encontrado

Código de error	Descripción
NCP01008	Error en la conexión de CNI
NCP01009	Error de coincidencia de la versión de CNI
NCP01010	Error en la recepción del mensaje de CNI
NCP01011	Error en la transmisión del mensaje de CNI
NCP01012	Error en la conexión de HyperBus
NCP01013	Error de coincidencia de la versión de HyperBus
NCP01014	Error en la recepción del mensaje de HyperBus
NCP01015	Error en la transmisión del mensaje de HyperBus
NCP01016	Error en el envío de GARP
NCP01017	Error en la configuración de interfaz

Códigos de error de nsx-kube-proxy

Código de error	Descripción
NCP02001	Puerto de puerta de enlace no válido de proxy
NCP02002	Error en el comando de proxy
NCP02003	Error en la validación de proxy

Códigos de error de la CLI

Código de error	Descripción
NCP03001	Error en el inicio de la CLI
NCP03002	Error en la creación del socket de la CLI
NCP03003	Excepción de socket de la CLI
NCP03004	Solicitud no válida del cliente de la CLI
NCP03005	Error en la transmisión del servidor de la CLI
NCP03006	Error en la recepción del servidor de la CLI
NCP03007	Error en la ejecución del comando de la CLI

Códigos de error de Kubernetes

Código de error	Descripción
NCP05001	Error en la conexión de Kubernetes
NCP05002	Configuración no válida de Kubernetes
NCP05003	Error en la solicitud de Kubernetes
NCP05004	Clave de Kubernetes no encontrada

Código de error	Descripción
NCP05005	Tipo de Kubernetes no encontrado
NCP05006	Excepción de monitor de Kubernetes
NCP05007	Longitud no válida del recurso de Kubernetes
NCP05008	Tipo no válido del recurso de Kubernetes
NCP05009	Error en el identificador del recurso de Kubernetes
NCP05010	Error en el identificador del servicio de Kubernetes
NCP05011	Error en el identificador del endpoint de Kubernetes
NCP05012	Error en el identificador de la entrada de Kubernetes
NCP05013	Error en el identificador de la directiva de red de Kubernetes
NCP05014	Error en el identificador del nodo de Kubernetes
NCP05015	Error en el identificador del espacio de nombres de Kubernetes
NCP05016	Error en el identificador del pod de Kubernetes
NCP05017	Error en el identificador del secreto de Kubernetes
NCP05018	Error en el back-end predeterminado de Kubernetes
NCP05019	Expresión de coincidencia no admitida de Kubernetes
NCP05020	Error en la actualización del estado de Kubernetes
NCP05021	Error en la actualización de la anotación de Kubernetes
NCP05022	Memoria caché de espacio de nombres de Kubernetes no encontrada
NCP05023	Secreto de Kubernetes no encontrado
NCP05024	El back-end predeterminado de Kubernetes está en uso
NCP05025	Error en el identificador del servicio de equilibrador de carga de Kubernetes

Códigos de error de OpenShift

Código de error	Descripción
NCP07001	Error en el identificador de ruta de OC
NCP07002	Error en la actualización del estado de ruta de OC