



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

Setting up Genesys Multicloud CX Private Edition

Configure VXLAN on OpenShift

Contents

- 1 Set up VXLAN tunnels to the OpenShift cluster
- 2 Create a VXLAN tunnel and profile on the BIG-IPs
- 3 Create new OpenShift host subnets for the BIG-IPs
- 4 Create a self IP address in the VXLAN
- 5 Create a partition on the BIG-IP system

Learn how you can set up VXLAN tunnels, which is the first step in SBC integration with the Genesys Multicloud CX private edition deployment on OpenShift.

Related documentation:

-
-

RSS:

- [For private edition](#)

Disclaimer

Genesys is committed to diversity, equality, and inclusivity. This includes using appropriate terms in our software and documentation. Therefore, Genesys is removing non-inclusive terms. For third-party products leveraged by Genesys that include such terms, Genesys uses the following as replacements.

- For the terms master/slave, Genesys uses “primary” and “secondary” or “primary” and “replica,” with exceptions for their use in third-party commands.
- For the terms blacklist/whitelist, Genesys uses blocklist/allowlist.
- For the term master, when used on its own, Genesys uses main wherever possible.

Set up VXLAN tunnels to the OpenShift cluster

```
bash-3.2$ ./oc get clusternetwork -o json | jq .items[0].network -r  
10.28.0.0/14
```

Create a VXLAN tunnel and profile on the BIG-IPs

1. On bigip-01, create a VXLAN tunnel and profile.

```
(tmsh)# create net tunnels vxlan vxlan-mp flooding-type multipoint  
  
(tmsh)# create net tunnels tunnel openshift_vxlan key 0 profile vxlan-mp local-  
address 10.10.2.60 secondary-address 10.10.2.37
```

2. On bigip-02, create a VXLAN tunnel and profile.

```
(tmos)# create net tunnels vxlan vxlan-mp flooding-type multipoint
```

```
(tmos)# create net tunnels tunnel openshift_vxlan key 0 profile vxlan-mp local-address 10.10.2.60 secondary-address 10.10.2.39
```

- Local-address: Internal floating IP
- Secondary-address: Internal self IP

The screenshot shows a web-based configuration interface for a network device. The breadcrumb navigation at the top reads "Network » Tunnels : Profiles : VXLAN » vxlan-multipoint". Below this, there is a "Properties" tab. The configuration is divided into two main sections: "General Properties" and "Settings".

General Properties	
Name	vxlan-multipoint
Partition / Path	Common
Parent Profile	vxlan
Description	

Settings	
Port	4789
Flooding Type	Multipoint

At the bottom of the settings section, there are two buttons: "Update" and "Delete..."

Network » Tunnels : Tunnel List » openshift_vxlan

Properties Tunnel Static Forwarding Table

Configuration

Name	openshift_vxlan
Partition / Path	Common
Description	
Key	0
Profile	vxlan-mp
Local Address	10. 10.2.60
Secondary Address	Specify... 10. 10.2.37
Remote Address	Any
Mode	Bidirectional
MTU	0
Use PMTU	<input checked="" type="checkbox"/> Enabled
TOS	Preserve
Auto-Last Hop	Default
Traffic Group	/Common/traffic-group-1

Delete

Create new OpenShift host subnets for the BIG-IPs

This process includes creating three manifests, one for each of the BIG-IPs and a third for setting up the floating IP.

1. Create one host subnet for each BIG-IP device.

f5-openshift-hostsubnet-bigip-01.yaml

```
# Setup for F5 bigip-01 hostsubnet
#
apiVersion: v1
kind: HostSubnet
metadata:
  name: bigip-01
  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: bigip-01
# 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: 10.10.2.37

```

f5-openshift-hostsubnet-bigip-02.yaml

```

# Setup for F5 big-ip-02 hostsubnet
#
apiVersion: v1
kind: HostSubnet
metadata:
  name: bigip-02
  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: bigip-02
# 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: 10.10.2.39

```

f5-openshift-hostsubnet-bigip-float.yaml

```

# Setup for F5 big-ip-floating interface hostsubnet
#
apiVersion: v1
kind: HostSubnet
metadata:
  name: big-ip-float
  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: big-ip -float
# 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: 10.10.2.60

```

2. Create the host subnet for the BIG-IPs by applying the preceding manifest files in OpenShift.

```

oc create -f f5-openshift-hostsubnet-bigip-01.yaml
oc create -f f5-openshift-hostsubnet-bigip-02.yaml
oc create -f f5-openshift-hostsubnet-bigip-float.yaml

```

3. Verify creation of the host subnets.

```

bash-3.2$ ./oc get hostsubnets
NAME
HOST                                HOST IP      SUBNET      EGRESS
CIDRS  EGRESS IPS
arol-eastus2-cpe-master-0          arol-eastus2-cpe-master-0
10.10.0.9    10.29.0.0/23
arol-eastus2-cpe-master-1          arol-eastus2-cpe-master-1
10.10.0.8    10.30.0.0/23

```

arol-eastus2-cpe-master-2 10.10.0.7 10.28.0.0/23	arol-eastus2-cpe-master-2
arol-eastus2-cpe-worker-eastus21-2bwqk 10.10.1.5 10.29.12.0/23	arol-eastus2-cpe-worker-eastus21-2bwqk
arol-eastus2-cpe-worker-eastus21-5v8hx 10.10.1.7 10.30.10.0/23	arol-eastus2-cpe-worker-eastus21-5v8hx
arol-eastus2-cpe-worker-eastus22-5mpvv 10.10.1.11 10.28.12.0/23	arol-eastus2-cpe-worker-eastus22-5mpvv
bigip-float float	bigip- 10.10.2.60 10.30.2.0/23
bigip-01	
bigip-01	10.10.2.37 10.31.0.0/23
bigip-02	
bigip-02	10.10.2.39 10.29.2.0/23

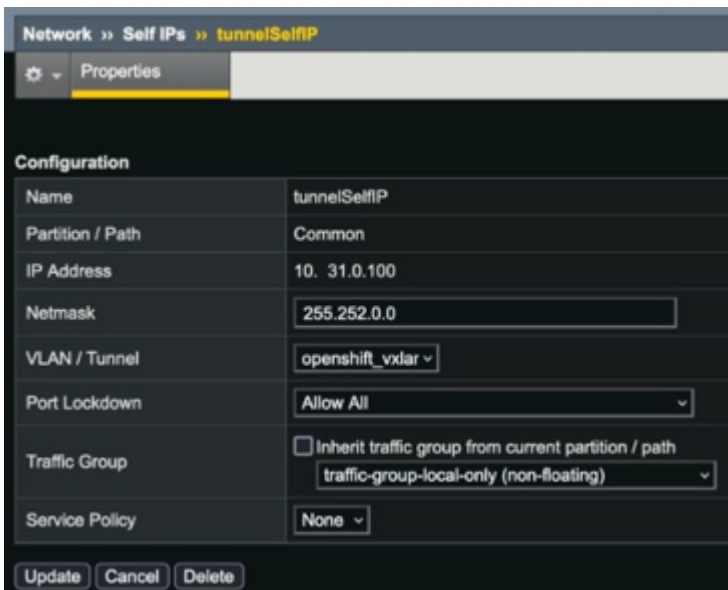
Create a self IP address in the VXLAN

When you create a self IP address in the VXLAN for each BIG-IP, you must ensure the following:

- The subnet mask you assign to the self IP must match the one that the OpenShift SDN assigns to nodes. By default, it is a /14.
- Specify a floating traffic group (for example, traffic-group-1). Otherwise, the self IP uses the BIG-IP system's default.

1. On bigip-01, create the self IP from host subnets bigip-01

```
(tmos)# create net self tunnelSelfIP address 10.31.0.100/14 allow-service all vlan openshift_vxlan
```



2. On bigip-02, create the self IP from host subnets bigip-02

```
(tmos)# create net self tunnelSelfIP address 10.29.2.100/14 allow-service all vlan openshift_vxlan
```

-
3. On the active BIG-IP, create a floating IP address in the subnet assigned by the OpenShift SDN from the host subnets big-ip-float

```
(tmsh)# create net self tunnelSelfIP address 10.30.2.100/14 allow-service all traffic-group traffic-group-1 vlan openshift_vxlan
```

The screenshot shows the configuration page for a Self IP named 'tunnelSelfIP_float'. The configuration is as follows:

Configuration	
Name	tunnelSelfIP_float
Partition / Path	Common
IP Address	10.30.2.100
Netmask	255.252.0.0
VLAN / Tunnel	openshift_vxlan
Port Lockdown	Allow All
Traffic Group	<input type="checkbox"/> Inherit traffic group from current partition / path traffic-group-1 (floating)
Service Policy	None

Buttons: Update, Cancel, Delete

Create a partition on the BIG-IP system

Create a partition on the BIG-IP system using the following command:

```
(tmsh)# create auth partition cisPartition
```