Brocade vADC Device Driver for OpenStack Neutron LBaaS

Deployment Guide

This guide explains how to deploy the Brocade vADC device driver for OpenStack Neutron LBaaS. It applies specifically to the OpenStack Kilo release and the Neutron LBaaS API version 2.

Prerequisites for using this guide are:

- A configured OpenStack environment, consisting of at least the Keystone, Neutron, Nova and Glance services. For HTTPS decryption, the Barbican service is also required.

- A working understanding of the above OpenStack services, or the related documentation (available from http://docs.openstack.org) to work from.

- Suitable licenses for the Brocade products you are going to use. These could be:
    - None, if you are using the Developer Edition of vTM in the "central cluster" deployment model for testing purposes.
    - One or more perpetual vTM licenses if you are using the "central cluster" deployment model in production.
    - A Brocade Services Director license, either of the Cloud Services Provider type, or the Enterprise type with an associated bandwidth pack.

- The necessary Brocade software packages, downloadable from the Brocade website:
    - A vTM Virtual Appliance image for the hypervisor you are using for OpenStack Nova. Must be version 10.1 or higher.
    - Optionally, the Brocade Services Director Virtual Appliance (2.1 or higher) for VMWare, or the Brocade Services Director software package if using a different hypervisor.

- A working understanding of the above Brocade products, or the related documentation (available from http://www.brocade.com) to work from.

The Brocade vADC driver supports various deployment options that are described in the first two sections of this guide. The steps for deploying the driver are included in the third section. Please make sure you have planned your deployment, selected the options you will use and configured the necessary prerequisite services before creating your Brocade LBaaS configuration file (see section 3) to ensure you have all the required components and settings in place.

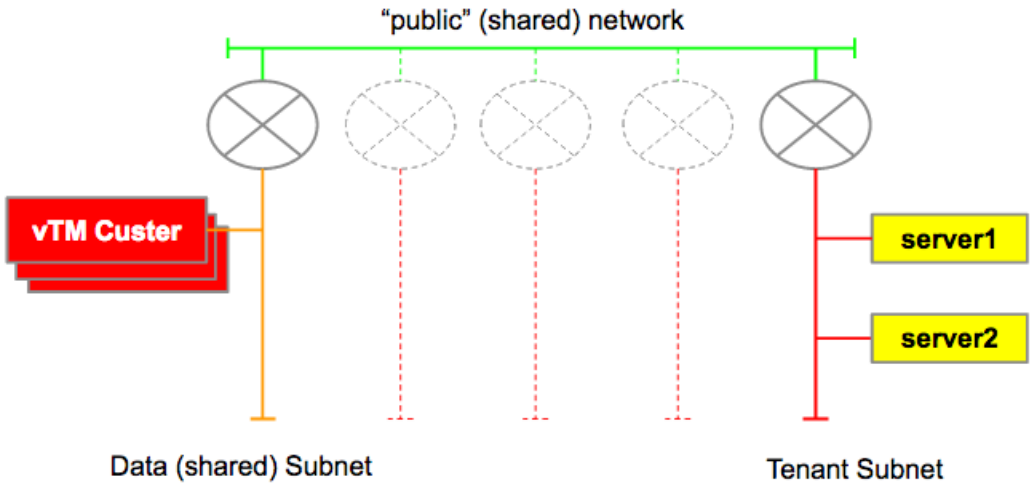
1. Deployment Models

The Brocade vADC device driver can employ Brocade vTMs to provide LBaaS services in one of the following ways:

1.1 Central Cluster of vTMs

This configuration ***does not*** require Brocade Services Director and is most suited to organizations that wish to use perpetual vTM license keys.

A central cluster of vTMs (up to 64) is shared between all OpenStack tenants. A shared network on which all LBaaS "loadbalancer" IP addresses are raised is required, and there is no resource isolation between tenants' services.
Although the load-balanced services use an IP address on a shared network, the back-end servers themselves remain within the tenants' own subnets.



N.B. All vTM configuration objects that are created on the cluster by the LBaaS driver will have the Neutron name of the object and the Keystone ID of the tenant included in the object's "Note" field.

"Loadbalancer" Objects
LBaaS "loadbalancer" objects are implemented as vTM Traffic IP Groups.
Each group contains the "Loadbalancer" IP address, one primary vTM on which to host the IP during normal operation, and a configurable number of backup vTMs to use in the event of a failure of the primary. The driver will automatically distribute Traffic IP Groups across the cluster to providing scalability and approximately even load distribution (assuming all "loadbalancers" process roughly the same amount of traffic). To facilitate Traffic IP failover, each Traffic IP address is added to the "`allowed_address_pair`" field of the Neutron port associated with each vTM.

"Listener" Objects
LBaaS "listener" objects are implemented as vTM Virtual Servers.
HTTP, HTTPS pass-through, HTTPS off-load, generic TCP (client-first) and UDP protocols are supported.
Certificates for HTTPS off-loading must be managed through the OpenStack Barbican tool.
The LBaaS "listener" "connection_limit" setting is implemented using a vTM Rate Class and a corresponding TrafficScript request rule to apply it. These are tied to the Virtual Server and will be deleted automatically along with it when the "listener" object is deleted.

"Pool" Objects
LBaaS "pool" objects are implemented as vTM Pools.
The LBaaS "pool" "session_persistence" setting is implemented using a vTM Session Persistence Class applied to the Pool. This is tied to the Pool and will be deleted automatically along with it when the "pool" object is deleted.

"Member" Objects
LBaaS "member" objects are implemented as nodes in the corresponding vTM Pool.

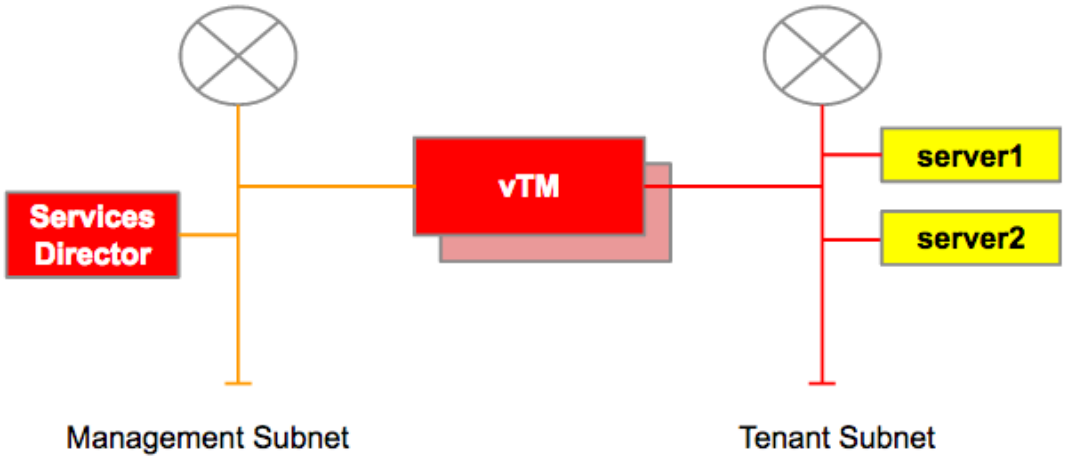
"Healthmonitor" Objects
LBaaS "healthmonitor" objects are implemented as vTM Monitors.
Whilst there is a 1:1 relationship between "pools" and "healthmonitors", the vTM Monitor object is not automatically deleted when the "pool" is deleted, allowing it to be reused by another "pool".

1.2 Private vTM Per-Tenant

This configuration ***does*** require Brocade Services Director and is most suitable for organizations that wish to provide their users with network-isolated LBaaS capability whilst keeping hardware resource use to a minimum.

A vTM instance or HA-pair (this is decided globally by the administrator) is automatically created as a Nova virtual machine for each tenant the first time they create an LBaaS "loadbalancer" object. All subsequent LBaaS services the tenant creates are hosted on the same instance/HA-pair. The virtual machine(s) resides within the tenant's environment and is connected directly to their own subnet. This deployment provides network isolation from other tenants, but does not provide resource isolation between LBaaS services.
The vTM instance(s) are locked to prevent accidental deletion and should not be touched by the tenants. To delete the instance(s), simply delete all LBaaS "loadbalancer" objects.

(The diagram below shows the vTM instance(s) connected to a management network; this is one available option for management traffic, please see section 2.1 for more details.)



The relationship between LBaaS objects and vTM configuration objects is the same as for the "Central Cluster of vTMs" - see section1.1.

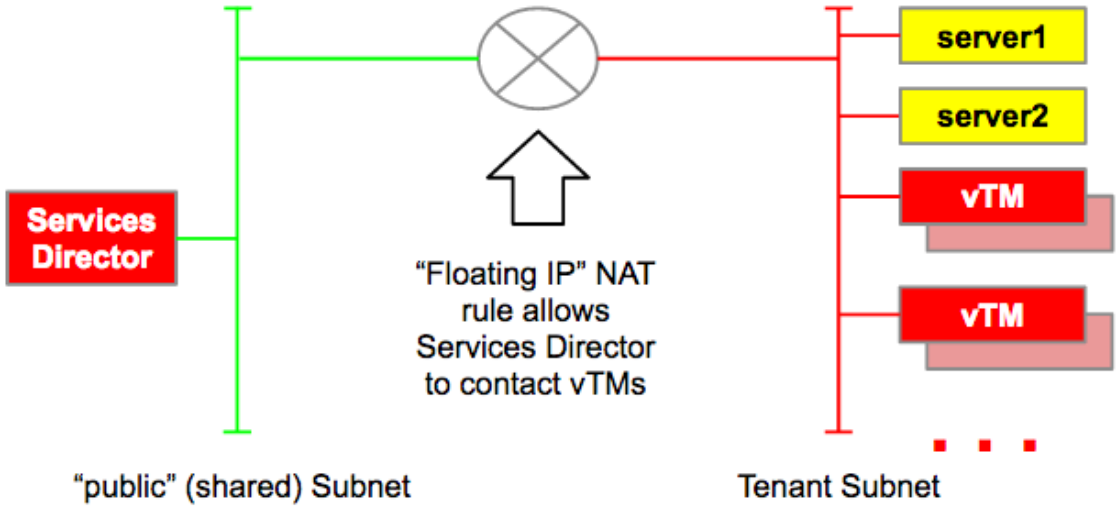
1.3 Private vTM Per-"loadbalancer"

This configuration ***does*** require Brocade Services Director and is most suitable for organizations where resource isolation and performance are a higher priority than hardware resource usage.

A vTM instance or HA-pair (this is decided globally by the administrator) is automatically created for each LBaaS "loadbalancer" object the tenant creates. This deployment provides full resource isolation between services but also uses up the most resources (CPU, memory, storage).

The vTM instance(s) are locked to prevent accidental deletion and should not be touched by the tenants. To delete the instance(s), simply delete the associated LBaaS "loadbalancer" object.

(The diagram below shows the vTM instance(s) communicating with the Services Director via a Neutron floating IP; this is one available option for management traffic, please see section 2.1 for more details.)



"Loadbalancer" Objects

LBaaS "loadbalancer" objects are implemented as vTM instances or HA-pairs. If single instances are used, no Traffic IP Groups are required and the instance will use the IP address of the Neutron port associated with the "loadbalancer" object for the data interface.

The relationship between all other LBaaS objects and vTM configuration objects is the same as for the "Central Cluster of vTMs" - see section 1.1.

2. Other Deployment Options

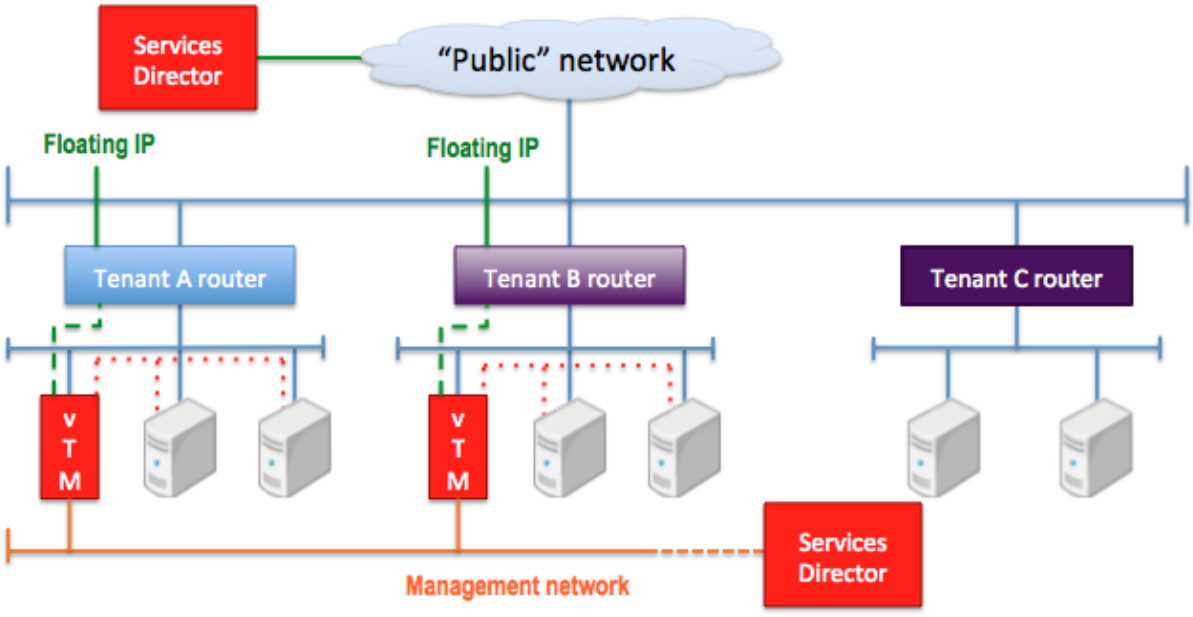
As well as the three deployment models described in section 1, the Brocade vADC device driver supports various options for the following:

2.1 Management Traffic

For the two deployment options that require Brocade Services Director, management communication between the SD and the vTM instances can be configured in one of two ways:

1. Over a dedicated management network (the vTM will have two vNICs – one for management, one for data) - orange in the diagram below

2. Via a Neutron floating IP mapped to the vTM's data vNIC - green in diagram below



To use a management network, the OpenStack administrator must create a suitable shared network and subnet before running the `brocade_lbaas_config_generator` tool (see section 3), as the network ID will be required.
When a management network is used, Neutron security group rules are automatically created and applied to ensure that management traffic can only come from the Services Directors.

2.2 Name Resolution

Brocade Services Director currently relies on name resolution. The vADC device driver provides a plugin mechanism for manipulating name resolution records, with the following implementations included:

1. /etc/hosts – The driver will manipulate the /etc/hosts file on the Neutron server. If there are multiple Neutron servers and/or Brocade Services Directors, these should all use the same copy of /etc/hosts, shared via rsync, NFS, etc.

2. Designate – The driver will talk to the Designate REST API to manipulate the name resolution records.

If a different mechanism is required (e.g. To manipulate a BIND server), a suitable plugin driver will need to be written. This driver should be included in the `brocade_neutron_lbaas/vtm/resolver.py` file as a class, should be named <DESCRIPTOR>Resolver (e.g. "`BindResolver`") and should implement the following methods:

```
def add_record(self, hostname, ip)
    """Adds a name resolution record"""

def delete_record(self, hostname)
    """Deletes a name resolution record"""
```

Once included in `resolver.py`, the custom driver can be selected when the `brocade_lbaas_config_generator` tool is run (see section 3), by entering, *in lower case,* the name that was used for <DESCRIPTOR> when prompted for the name resolution method.

2.3 GUI Access

For the two deployment options that create vTM instances dedicated to the tenant, the administrator can optionally (as a global setting) provide the tenant with read-only access to the GUI for monitoring.

If allowed, the GUI can be accessed in a web browser on TCP port 9090 of the IP address assigned to the data interface, from within the tenant's subnet. IP forwarding may be required

The username is "monitor" and the initial password is "password".

3. Installation and Configuration

The following instructions assume you have a fully configured OpenStack Kilo (or above) environment, including all of the required Neutron networks.

- Download the appropriate Brocade vTM Virtual Appliance image (must be vTM 10.1 or above) for your environment and copy the image file to the Glance server.

- As an OpenStack admin user, register the image with Glance:

```
# glance image-create --name Brocade-vTM --is-public True --disk-format
<FORMAT> --container-format bare --file <IMAGE_FILE>
```

NB. You can omit the `--is-public` argument if using the shared central cluster deployment model and you don't wish end-users to have access to the vTM image.

- If a built-in machine flavor does not have the required settings, create one:

```
# nova flavor-create <NAME> <ID> <RAM> <DISK> <VCPUS>
```

A minimum of 2GB RAM and 16GB storage should be allocated.

- If using a shared central cluster of vTMs:
    - Create and configure the cluster (see vTM User Guide or Getting Started Guide) ensuring each cluster member has a vNIC on the shared data network, and on the management network if applicable.
    - Ensure that the REST API is enabled on all cluster members.
- Else if using one of the deployment options that requires Services Director:
    - Create and configure the necessary Services Director instances (see Services Director User Guide)
    - Create the required Version, License and Feature Pack resources to be used by the vTM instances that are created. NB. The instances are unmanaged, so the Version entry can be a dummy.

- Clone the device driver GitHub repository:

```
# git clone https://github.com/brocade-vadc/neutron-lbaas-device-driver
```

- Install the driver:

```
# cd neutron-lbaas-device-driver
# sudo python setup.py install
```

- Configure the driver:

```
# brocade_lbaas_config_generator > \
/etc/neutron/services/loadbalancer/brocade.conf
```

You will have to answer questions about the deployment options you wish to use, service credentials, and identifiers of key components such as network IDs, Services Director hostnames etc.

- Configure the driver:

This will be done on the node that contains the neutron-server. In a multi-node setup, you will perform this configuration on the Controller Node.

In `/etc/neutron/neutron.conf`, under the `[DEFAULT]` section, ensure the "`service_plugins =`" line contains
`neutron_lbaas.services.loadbalancer.plugin.LoadBalancerPluginv2`

In `/etc/neutron/neutron_lbaas.conf`, under the `[service_providers]` section, set

```
service_provider = \
LOADBALANCERV2:brocade:neutron_lbaas.drivers.brocade.driver_v2.BrocadeLoadBalancerDriver:default
```

- Stop the Neutron server (System and installation-specific)

- Start the Neutron server with the additional CLI parameter
`--config-file /etc/neutron/services/loadbalancer/brocade.conf`

At this point, you should be able to create, modify and delete LBaaS services through the Neutron command line tool, the Neutron REST API or the Horizon GUI - see http://docs.openstack.org for information.