Installation Runbook for

F5 Networks BIG-IP LBaaS Plugin for OpenStack Kilo

Application Version	F5 BIG-IP TMOS 11.6
MOS Version	7.0
OpenStack Version	Openstack Kilo
Application Type	**Validation of LBaaS Plugin for BIG-IP**

Content

Document History

Version	Revision Date	Description
0.1	11-12-2015	Initial Version

1 Introduction

This document is to serve as a detailed Deployment Guide for F5 Networks' OpenStack LBaaSv1 plug-in. F5 offers an LBaaS plug-in compatible with the OpenStack Neutron LBaaS v1.0 service. This document describes the reference architecture, installation steps for certified Mirantis Openstack (MOS) and the F5 OpenStack LBaaSv1 plug-in. PWe will also highlight any limitations in the current LBaaS v1.0 specification.

1.1 Target Audience

This guide is aimed at Mirantis Fuel Openstack Architects and System Administrators who wish to use the F5 OpenStack LBaaSv1 Plug-in to manage BIG-IP hardware or Virtual Editions in their cloud environments. This guide assumes that the reader is familiar with BIG-IP Hardware and Virtual Editions as well as basic layer 2-3 networking. For additional information on the F5 product line please visit www.f5.com.

2 Application overview

The F5 OpenStack LBaaSv1 Plug-in allows OpenStack operators to utilize BIG-IP hardware or Virtual Edition in MOS.

F5's OpenStack LBaaSv1 Plug-in provides management of BIG-IP load balancing services, including:

- Local load balancing;
- Support for multi-tenant deployments;
- Orchestration integration via F5's iControl®
- Single-arm or dual-arm topologies
- Integrated Neutron network services;Multi-service extensibility;Full virtual device onboarding;Support for physical or virtual BIG-IP devices*;SDN gateway functionality with Open vSwitch GRE and VXLAN;* Requires TMOS 11.5 or higher.

3 Joint Reference Architecture

When Neutron LBaaSv1 API calls are issued to your Neutron controller, the community LBaaSv1 plugin will attempt provision LBaaSv1 resources using either a default or designated service provider driver.
The F5 OpenStack LBaaSv1 service provider drivers, running within the Neutron controller process(es), utilize Neutron RPC messaging queues to issue provisioning tasks to specific F5 OpenStack LBaaSv1 agent processes. Upon starting and successfully communicating with configured TMOS device API endpoints, each agent process registers its own specific named

queue to receive tasks from one or more Neutron controllers.

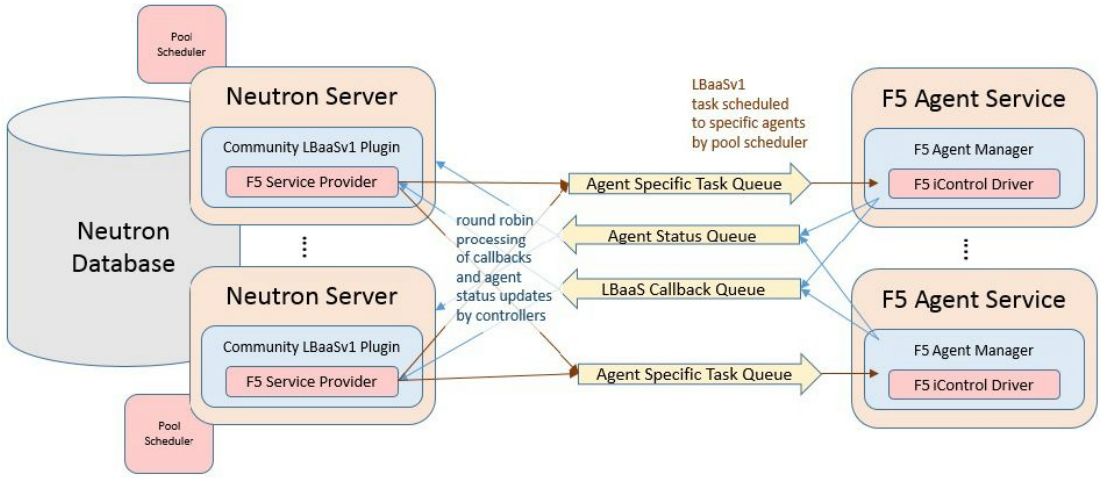


Figure 1: F5 OpenStack LBaaS Deployment

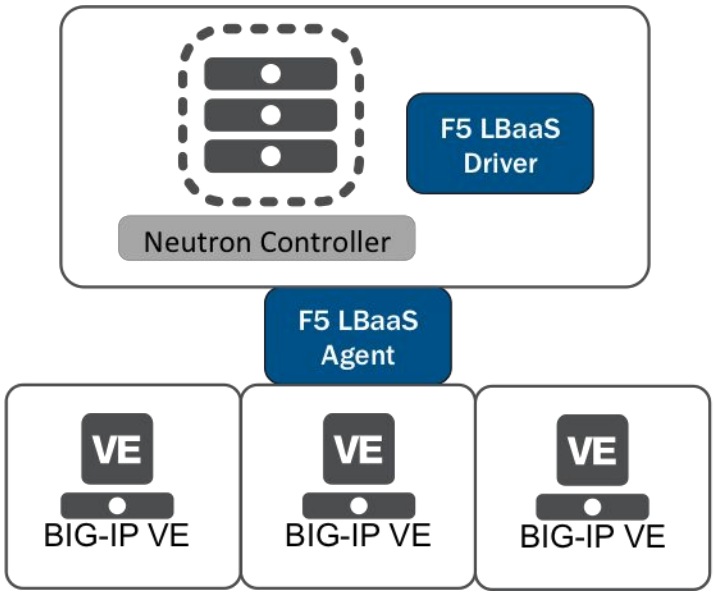
The F5 OpenStack LBaaSv1 agents make callbacks to the service provider drivers to query additional Neutron network, port, and subnet information; allocate Neutron objects like fixed IP address for SelfIP and SNAT translation addresses; and report provisioning and pool status. These callback requests are placed on an RPC message queue which is processed by all listening f5 LBaaSv1 service provider drivers in a round robin fashion.

Since all Neutron controller processes are working off the same backend database, it does not matter which of the available Neutron controller processes handle these callback requests.

4 Physical & Logical Network Topology

You can use the F5 OpenStack Plug-in in any number of ways according to your specific needs. Here, we show a few example configurations. At minimum, you must install one driver and one agent on your Neutron controller.

Figure 2: The F5 OpenStack LBaaS Plug-in agent orchestrating three BIG-IP VE clusters from one host.



The F5 OpenStack LBaaS Plug-in orchestrating one BIG-IP VE cluster from 2 nodes; using 2 agents for the same environment provides redundancy in LBaaS provisioning.

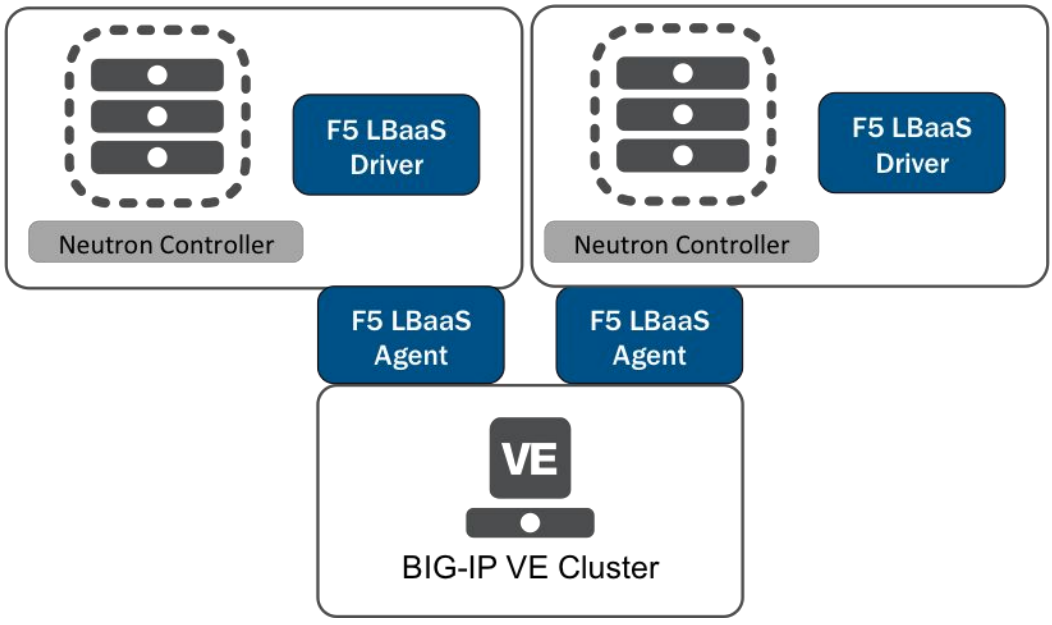


Figure 3: Multiple Neutron Controllers Connecting to a Single BIG-IP Cluster

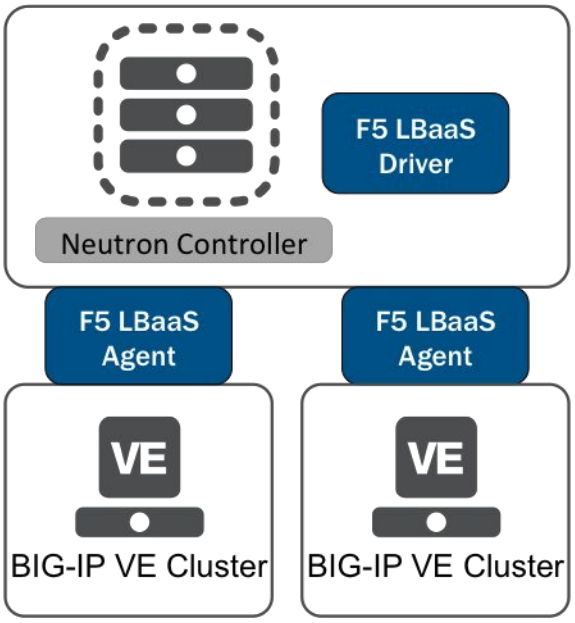


Figure 4: The F5 OpenStack LBaaS Plug-in orchestrating two different BIG-IP VE clusters from the same node..

5 Installation & Configuration

5.1 Environment Preparation

Please follow the documentation for MOS installation (Create a new OpenStack environment..

5.2 Overview of MOS installation steps

- Minimum configuration of 1 Fuel Master node, 1 - MOS Controller node, 3 - MOS compute nodes, Download Mirantis x.x ISO from Mirantis website.
- Follow the installation instructions as specified in Mirantis OpenStack User Guide.

Minimum System Requirements

OS	Mode	HV	Network	Storage
			VLAN	Ceph
CentOS	HA	KVM	x	
Ubuntu	HA	KVM		x

5.3 Overview of F5 OpenStack LBaaSv1 plug-in installation steps

The following section provides detailed instructions for installing the F5 OpenStack LbaaSv1 Plug-in driver and agent. The download package is available from F5's DevCentral.

You will need to log in to your Neutron controller and install the plug-in driver and agent there manually. At least one driver and one agent must be installed on your Neutron controller. In brief, the steps are:

1. Install the driver on the Neutron controller.
2. Install the agent on the Neutron controller.
3. Configure the agent.
4. Configure F5PluginDriver as the load balancing service provider.
5. Define load balancing service and service provider in neutron.conf and neutron_lbaas.conf, respectively.

The initial set-up (one driver, one agent on the same controller) will allow you to provision services for one standalone BIG-IP cluster. To add redundancy or provision additional clusters, you can to install the driver and agent on additional controller nodes according to your specific needs.

Note: All commands included in this section are for Ubuntu 14.04

5.4 Driver and Agent installation steps

Installing the LBaaS plug-in for OpenStack

Before installing the F5 LBaaS plug-in, the following requirements must be met.

- Install and set up an OpenStack host environment.
- Install OpenStack software and configure it appropriately.
- Install the OpenStack Neutron service.

Documentation for how to install and configure OpenStack components can be found at: http://docs.openstack.org/.

The system administrator must install the F5 Load Balancing as a Service (LBaaS) plug-in so that you can then provision a device on OpenStack.

1. Download the F5 LBaaS plug-in from Github. Install the plug-in driver on your OpenStack control host. Choose the driver that corresponds to your OpenStack control host operating system:
    - For a Red Hat derivative, install the plug-in driver by running the following command: `$ rpm -i f5-lbaas-driver.rpm`.
    - For a Debian derivative, install the plug-in driver by running the following command: `$ dpkg -i f5-lbaas-driver.deb`.

    Tip: The .rpm and .deb package names may differ slightly from the examples provided here.

Important: You may need increased privileges to install the driver.

2. Install the plug-in driver on your OpenStack network host. Choose the driver that corresponds to your OpenStack network host operating system:
    - For a Red Hat derivative, install the plug-in driver by running the following command: `$ rpm -i f5-lbaas-driver.rpm`.
    - For a Debian derivative, install the plug-in driver by running the following command: `$ dpkg -i f5-lbaas-driver.deb`.

    Tip: The .rpm and .deb package names may differ slightly from the examples provided here.
    Important: You may need increased privileges to install the driver.
3. Install the plug-in agent on your OpenStack network host based on your OpenStack network host operating system:
    - For a Red Hat derivative, install the agent by running the following command: `$ rpm -i f5-lbaas-agent.rpm`
    - For a Debian derivative, install the agent by running the following command: `$ dpkg -i f5-lbaas-agent.deb`

    Tip: The .rpm and .deb package names may differ slightly from the examples provided here.
    Important: You may need increased privileges to install the driver.

Once the plug-in driver and agent are successfully installed, you need to configure the plug-in.

Configuring the LBaaS plug-in for OpenStack

You must install the F5 LBaaS plug-in before you can configure it.

You configure the F5 Load Balancing as a service (LBaaS) plug-in so that you can then provision OpenStack LBaaS resources (such as virtual IP addresses, pools, pool members, or health monitors).

1. On your OpenStack control host, configure the OpenStack Neutron service to use the F5 LBaaS plug-in.
    a. Use a text editor to open the OpenStack Neutron service configuration file neutron.conf. On the OpenStack network host, the configuration file is located at /etc/neutron/.
    b. Locate the section where you can configure OpenStack Neutron extension services. You can find this section by searching for [service_providers].
    c. Add an entry for the F5 LBaaS plug-in similar to the following: `service_provider=LOADBALANCER:f5:neutron.services.loadbalancer.drivers.f5.plugin_driver.F5PluginDriver`. If you want the F5 LBaaS plug-in to provide service for OpenStack LBaaS by default, add `:default` to the end of the entry.

2. On your OpenStack network host, use a text editor to revise the plug-in agent configuration file f5-device-lbaas-agent.ini. On the OpenStack network host, the configuration file is located at /etc/neutron/.
    a. Set the value of the use_unsupported_community_plugin_configuration field to `False`.
    b. When you complete all of the revisions, save and close the configuration file.
3. On the OpenStack control host, edit the OpenStack Horizon service configuration file to configure the OpenStack Horizon service to display the LBaaS service in the OpenStack user interface.

    Tip: The configuration file is located at /etc/openstack-dashboard/local_settings.
    a. In the configuration file, locate the OpenStack Neutron GUI settings, by searching for OPENSTACK_NEUTRON_NETWORK.
    b. Set the value of the enable_lb key to `True`.
    c. When you complete this revision, save and close the configuration file.
4. On the OpenStack compute host, edit the OpenStack Nova policy file to allow for statistics to be collected for OpenStack users.

    Tip: The OpenStack Nova policy file is named policy.json. This file is located at /etc/nova/.
    a. In the policy file, locate the statistics collection section, by searching for compute_extension:server_diagnostics.
    b. Set the value of the compute_extension:server_diagnostics key to `rule:admin_or_owner`.
    c. When you complete this revision, save and close the policy file.
5. To reload the configuration changes just completed, restart the following services: neutron-server, f5-bigip-lbaas-agent, and httpd.

Once the plug-in is successfully configured, you need to configure the BIG-IP system instance.

Configuring the BIG-IP instance

You must create a BIG-IP instance in your OpenStack project before you can configure the BIG-IP system for use with the LBaaS plug-in.
Each BIG-IP instance you create must be configured so it can be used as part of an OpenStack LBaaS.
Note: All of the steps you perform in this task, except for one, are documented in the *BIG-IP System: Initial Configuration* guide. The one exception, configuring redundant device options, is documented in the *BIG-IP Device Service Clustering: Administration* guide.

1. Activate the license for the BIG-IP system.
2. Provision the BIG-IP modules you plan to use.
3. Configure the general properties.

4. Specify the password for the Admin account. The password must match the one specified for the bigip_management_password when you configured the LBaaS plug-in.
5. Configure the internal network configuration. The value you use for the internal VLAN must use the value of the fixed IP and subnet for your OpenStack internal network.
6. Configure the external network configuration. The value you use for the external VLAN must use the value of the fixed IP and subnet for your OpenStack external network. Once you specify the external network, you have the first BIG-IP instance configured for LBaaS use.
7. Configure the redundant device options to specify the BIG-IP device peers required for load balancing. Refer to the *BIG-IP Device Service Clustering: Administration* guide for step-by-step instruction.

Provisioning OpenStack LBaaS resources

Before you can provision LBaaS resources on OpenStack, you must have the following elements in place.

- BIG-IP system instance installed and configured.

Using the OpenStack user interface, you create and configure the resources needed to support LBaaS.

1. Log in to the OpenStack user interface and then select Load Balancers in the navigation pane.
2. Create a pool.
    a. Click the Add Pool button to create an OpenStack pool.
    b. Type in a name for the pool.
    c. In the Provider list, select f5.
    d. In the Subnet list, select the subnet in which your OpenStack pool members are located.
    e. In the Protocol list, select the protocol appropriate for your network.
    f. In the Load Balancing Method list, select the type of load balancing appropriate for your network.
3. Create a virtual IP server (VIP) and associate it with the just created pool.
    a. Click the More button for the pool and then select Add VIP.
    b. Type in a name for the VIP.
    c. In the Protocol Port list, select the protocol appropriate for your network.
    d. In the Load Balancing Method list, select the type of load balancing appropriate for your network.
4. Create a health monitor that you can associate with the pool.
    a. Click Monitors > Add Monitors to create the new OpenStack health monitor.

b. Make appropriate selections for Type, Delay, Timeout, and Max Retries.
c. In the Protocol Port list, select the protocol appropriate for your network.
d. In the Load Balancing Method list, select the type of load balancing appropriate for your network.

5. Click Pools > More > Add Health Monitor to associate the health monitor with the pool.
6. Add pool members to the pool.
    a. Click Members > Add Member.
    b. Make appropriate selections for Pool, Member(s), and Protocol Port.

If you want the virtual server (VIP) to be publicly accessible and it is not, then after you create the virtual server you must create a floating IP address and associate it with the OpenStack port of the virtual server. Documentation for this task is available on the OpenStack website at: http://docs.openstack.org/.

Table 1. F5 OpenStack LBaaS Agent Configuration Options

[DEFAULT]	
debug = True	Show debugging output in log (sets DEBUG log level output).
periodic_interval = 10	The LBaaS agent will resync its state with Neutron to recover from any transient notification or rpc errors. The periodic interval is number of seconds between attempts.
service_resync_interval = 500	How often the agent throws away its service cache and resyncs assigned services with the neutron LBaaS plugin.
environment_prefix = uuid	The environmental prefix that the agent applies to BIG-IP objects it creates. The default setting is 'uuid'.
WARNING: **You should set the environmental prefix before creating any BIG-IP objects.** If you change it after you've created objects, those objects will no longer be associated with the agent and will have to be managed manually.	

static_agent_configuration_data = name1:value1, name1:value2, name3:value3	Static configuration data to send back to the plugin. A single entry, or a comma-separated list of name:value entries, which will be sent in the agent's configuration dictionary to Neutron. This can be used on the plugin side of Neutron to provide agent identification for custom pool-to-agent scheduling.
Device Setting	**Device type for LBaaS**
f5_device_type = external	external - external (hardware or VE)
f5_device_type = guest_admin	guest_admin - VE created under the admin tenant
f5_device_type = guest_tenant	guest_tenant - VE created under the pool tenant
HA model	**Set High Availability (HA) model**
f5_ha_type = standalone	Single device, no HA
f5_ha_type = pair	Active/standby two-device HA
f5_ha_type = scalen	Active device cluster
Note: If the device is external, the devices must be imported into openstack for the appropriate HA mode or else the driver will not provision devices.	
Sync mode	**Set policy sharing across devices**
f5_sync_mode = autosync	Syncable policies configured on one device are synced to the group.
f5_sync_mode = replication	Each device is configured separately.
L2 Segmentation Mode Settings	Configure L2 segmentation for pools or VIPs created on VLAN networks
Device VLAN to interface and tag mapping	A comma-separated list of strings that specifiy to which interface the agent should map a VLAN and state if VLAN tagging should be enforced by the external device. The

	string should use the following format: "**physical_network:interface_name:tagged**".
	'"physical_network" corresponds to "provider:physical_network" attributes
	"interface_name" is the name of an interface or LAG trunk
	"tagged" is a boolean (True or False)
standalone: f5_external_physical_mappings = default:1.1:True	Maps the 'default' physical network to interface '1.1' with tagging enforced.
pair or scalen: f5_external_physical_mappings = default:1.3:True	Maps the 'default' physical network to interface '1.3' with tagging enforced.
Note: If a network does not have a provider:physical_network attribute, or the provider:physical_network attribute does not match in the configured list, the 'default' physical_network setting is applied. At a minimum, you must have a 'default' physical_network setting. The '1.1' and '1.2' interfaces are used for HA purposes.	
Device Tunneling (VTEP) self-ips	A single entry or a comma-separated list of cidr (h/m) format self-ip addresses - one per BIG-IP device - to use for VTEP addresses.
f5_vtep_folder = 'Common'	Sets the VTEP to use the default partition ("Common") on the BIG-IP.
f5_vtep_selfip_name = 'vtep'	Sets the VTEP self-ip name to "vtep".
Tunnel types	A comma-separated list of tunnel types to report as available from this agent and to send to compute nodes via tunnel_sync rpc messages. This should match the ml2 network types on your compute nodes.
advertised_tunnel_types = gre	use only gre tunnels

advertised_tunnel_types = vxlan	use only vxlan tunnels
advertised_tunnel_types = gre,vxlan	use gre and vxlan tunnel networks
advertised_tunnel_types = vlan	use only vlans
Note: If no gre or vxlan tunneling is required, these settings should be commented out or set to None.	
Static ARP population for members on tunnel networks	Creating a static ARP entry allows the agent to avoid having to learn Pool Members' MAC addresses via flooding. Use a boolean "True" or "False" entry to specify whether an entry should be created if a Pool Member IP address is associated with a gre or vxlan tunnel network and a tunnel fdb record is added.
f5_populate_static_arp = True	Sets the agent to create a static arp entry.
Device Tunneling (VTEP) selfips	A boolean "True" or "False" entry which determines if the BIG-IP will use L2 Population service to update its fdb tunnel entries.
l2_population = True	Tells the agent to configure the BIG-IP to use L2 Population service to update fbd tunnel entries.
Note: This needs to be set up in parallel with the tunnel agents. If the BIG-IP and tunnel agent settings don't match, the tunnel setup will not work properly.	

5.4.1 Target use case(s)

1. Provision and deploy a Virtual Server, Load Balancing Pool, Web Application Server and Monitor

5.4.2 Test cases

1. Validate that application is available and user can access web based application via Virtual Server deployed from MOS.

5.4.2 Test Tools

N\A

5.4.3 Test Results

N\A