# OPENSTACK NEUTRON AND SOFTWARE-DEFINED NETWORKING

Red Hat OpenStack Platform

## SUMMARY AT A GLANCE

- Take advantage of OpenStack's pluggable networking architecture.
- Choose Red Hat Certified networking plug-ins based on your unique requirements and needs.
- Add OpenDaylight and the ODL plug-in if needed.

## TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 THE NETWORK STACK</td>
<td>2</td>
</tr>
<tr>
<td>2 DEFAULT NETWORKING</td>
<td>2</td>
</tr>
<tr>
<td>3 ADDITIONAL PLUG-INS</td>
<td>3</td>
</tr>
<tr>
<td>4 CHOOSING THE RIGHT NETWORKING PLUG-IN</td>
<td>4</td>
</tr>
<tr>
<td>5 RED HAT'S CERTIFIED COMMERCIAL PLUG-INS</td>
<td>5</td>
</tr>
<tr>
<td>6 RED HAT AND OPENDAYLIGHT</td>
<td>5</td>
</tr>
<tr>
<td>7 SUMMARY</td>
<td>6</td>
</tr>
</tbody>
</table>
THE NETWORK STACK

The OpenStack® project keeps gaining in popularity, as users and enterprises alike explore the wide variety of opportunities it represents. Integrating the project’s dozens of components can be tricky, especially with its rapid release cycle. As such, many companies are finding it useful to partner with an experienced OpenStack organization that has the expertise needed to make implementation run more smoothly.

OpenStack’s Networking component (Neutron) has a pluggable architecture in which different plug-ins can offer different back-end implementation of the APIs. Red Hat® OpenStack Platform ships with open source plug-ins, which are fully supported and tested by Red Hat. Based on their requirements and needs, customers have the option to deploy other plug-ins from our technology partners, which are certified against the Red Hat OpenStack Platform product, and are jointly supported by Red Hat and the plug-in vendor.

The aim of this document is to provide guidance and messaging around our software-defined networking (SDN) offering, with the ultimate goal of helping our customers and partners to pick the right networking solution for them.

DEFAULT NETWORKING

Red Hat OpenStack Platform 8 (based on the upstream Kilo release) defaults to the following open source plug-ins and back-end implementation, which are directly supported by Red Hat:

- **Core Neutron plug-in**: Modular Layer 2 (ML2)
- **ML2 mechanism driver**: Open vSwitch
- **Supplementary Open vSwitch-based solution**, including ovs-agent, dhcp-agent (based on dnsmasq), routing (based on Kernel forwarding and radvd for IPv6), network address translation (based on iptables), as well as metadata agent and proxy to enable cloud-init support for virtual machine (VM) instances
- **Advanced services plug-ins**
  - Load-Balancing-as-a-Service (LBaaS): with HAProxy
  - Firewall-as-a-Service (FWaaS): with iptables (technology preview)
  - Virtual-Private-Network-as-a-Service (VPNaaS): with Libreswan (technology preview)

---

1 https://access.redhat.com/support/offerings/techpreview
With this solution, Neutron can be used in two major architectures:

1. **Provider networks**, where an administrator creates logical Neutron networks that are directly mapped to existing networks in the physical datacenter, usually realized using 802.1Q VLANs. Tenants can then bridge their VMs into these networks. With this architecture, Neutron provides only basic L2 connectivity, as well as security-groups and optional DHCP service. L3 routing and additional advanced network services are placed on the physical network and not managed by OpenStack.

2. **Tenant networks**, in which Neutron provides all networking aspects, including routing, NAT, and optional advanced L4-L7 services. Tenant networks can use 802.1Q VLANs, but a more common approach is to use overlay tunnels (e.g., with GRE or VXLAN) on an underlying L2/L3 fabric. To route packets between tenant networks or from a tenant network and an external network, the traffic needs to traverse a virtual router handled in software by the l3-agent.

It’s important to emphasize that while a Neutron server has knowledge about the logical network’s topology—specifically the placement of the different endpoint ports (and their corresponding MAC and IP addresses)—Neutron itself is not an SDN controller and should not be considered as such. The information available in Neutron is being used as if it were a controller to some extent (e.g., for the L2 Population feature that optimizes the data traffic flow based on topology information from Neutron server), but those are exceptions and not the design principle of Neutron or the Open vSwitch-based solution. Additionally, with this default offering, Neutron cannot manage or influence any physical network devices (e.g., with protocols such as OpenFlow or OVSDB).

**ADDITIONAL PLUG-INS**

There are additional open source plug-ins available for Neutron that are shipped and supported by Red Hat OpenStack Platform 8, including:

- **The Linux® bridge mechanism driver.** This can be used as an alternative to Open vSwitch. Red Hat strongly recommends that customers use the Open vSwitch driver because this is where our engineering, quality assurance, and testing efforts are focused. That said, Red Hat still provides support for customers who would like to use the Linux bridge driver.

- **The L2 Population (l2population) mechanism driver.** This can be enabled when necessary if VXLAN or GRE overlays are used. This driver allows for a more efficient, on-demand setup of overlay tunnels between the nodes, eliminating the default full-mesh that is deployed by default. In addition, it’s also possible to enable the local address resolution protocol (ARP) responder feature, which provides response for ARP requests generated by VMs from the local vSwitch on each compute node. This optimization saves ARP traffic from traversing the network fabric.

- **The SR-IOV (sriovnicsswitch) mechanism driver.** This driver allows for the assignment of SR-IOV virtual functions (VFs) directly into VM instances, so that the VM is communicating directly with the NIC controller, effectively bypassing the vSwitch. This driver requires SR-IOV-enabled network adapter on the compute node.²

² For a list of supported capable NICs please refer to https://access.redhat.com/articles/1390483
CHOOSING THE RIGHT NETWORKING PLUG-IN

Like any network design, the default ML2/Open vSwitch-based solution has both pros and cons. Red Hat is actively working together with our partners, customers, and the OpenStack community to stabilize and enhance this solution, and we are fully committed to supporting it in production. We have strong engineering and quality assurance teams focusing on this solution and are actively involved in the upstream community.

SDN is many things to many people. It’s important to understand what this solution can and cannot provide, and most importantly, the requirements and needs. Some customers are just looking for a more traditional design and would benefit from a provider network configuration that is considered simple and stable. Others are looking for an overlay-based solution managed by OpenStack which, if properly designed, is available and recommended as well. But some are looking for a controller to program their existing network switches or for some form of tight integration with the physical fabric, which is not available with this solution.

When it comes to choosing a networking plug-in for OpenStack, there are several business and technological factors that should be taken into consideration. It is not the purpose of this document to list or cover all of them, but here are some points that can be used as a reference:

APPLICATIONS

• What type of applications you will run in the cloud is a crucial consideration. Different applications may have different design requirements that can impact the networking, security, and high-availability setup. Some applications may also require advanced networking features such as IPv6, dynamic routing, or IP multicast.

SERVICES

• It’s important to define what network services (e.g., IPAM, routing, NAT, load balancing) are expected in your cloud, who should provide each service, and if there are existing external services to be used.

INTERFACING WITH THE NETWORKING IMPLEMENTATION

• Different solutions offer different ways of interfacing with the network, such as an API, a graphical user interface (GUI), or specific protocols. This can significantly impact the way you operate and monitor your cloud.

EXISTING/PLANNED PHYSICAL NETWORK

• The virtual network solution must also work well with your physical (underlay) network. Some customers expect a high level of interaction between the physical network and OpenStack, and the ability to control the physical fabric via Neutron.
RED HAT’S CERTIFIED COMMERCIAL PLUG-INS

Red Hat offers a certification service for third-party software as a benefit of our Technology Partner program. The current certification for Neutron covers core plug-ins, ML2 drivers, and service plug-ins for LBaaS from selected leading partners in our ecosystem. Red Hat is constantly working to expand this list to give our customers maximum flexibility and choice when designing their Red Hat OpenStack Platform environment. Any certified plug-in that meets the customer requirements and needs can be discussed and evaluated.

The official catalog of certified Neutron plug-ins by Red Hat OpenStack Platform version is externally visible and can be found in the Red Hat Customer Portal. Neutron plug-ins are listed under “Certified Software” – “Red Hat OpenStack Platform.”

RED HAT AND OPENDAYLIGHT

Red Hat is a platinum founding member of OpenDaylight, a collaborative open source project hosted by The Linux Foundation that aims to accelerate the adoption of software-defined networking (SDN) and network functions virtualization (NFV).

In OpenDaylight, Red Hat is developing the features necessary to support the network virtualization requirements of an OpenStack deployment. This architecture features OpenDaylight as an SDN controller managing Open vSwitch instances in each compute node using OVSDB and OpenFlow protocols to build GRE or VXLAN overlays. This design also enables managing top of rack (ToR) switch ports as tunnel endpoints for use as gateways or hardware offloaded tunneling.

Starting with the OpenStack Icehouse release, there has been an ML2 mechanism driver in Neutron to enable communication between Neutron and OpenDaylight. The initial solution available in Icehouse, based on the Hydrogen version of OpenDaylight, was mostly focused around core L2 connectivity and still requires Neutron DHCP and L3 agents to handle routing and network services. Going forward, Red Hat is working to enhance this solution so we can handle more network operations through OpenDaylight.

The ODL ML2 driver ships starting with Red Hat OpenStack Platform 5 as a technology preview and is not officially supported by Red Hat at this point. While we do ship the Neutron driver that lets you start using the OpenDaylight integration, the user has to build or otherwise obtain an OpenDaylight controller, which we do not provide.

3 https://access.redhat.com/certifications
4 https://access.redhat.com/support/offerings/techpreview
SUMMARY

If you would like to join our open source community and try out OpenStack for free, visit the RDO site. You’ll find plenty of documentation to help get you started, forums where you can connect with other users, and community-supported packages of the most up-to-date OpenStack releases available for download.

If you are looking for enterprise-level support, a hardened, production-ready platform, and a certified partner program, then you need Red Hat OpenStack Platform. Red Hat can help you get started with Neutron and find the right networking solution for your OpenStack-based cloud.