

OpenStack Network Plugins Development Proposal

1. Vision and Goals Of The Project:

The Openstack Neutron Cisco ASR1K plugin main goal is to integrate OpenStack with Cisco's ASR1K Router. High-Level Goals of the Openstack Neutron Cisco ASR1K plugin include:

- Extend OpenStack Neutron and API's support to the Cisco ASR1K router
- Ability to add custom configuration to the ASR1K router
- Potential Firewall support

Users/Persons Of The Project:

The Openstack Neutron Cisco ASR1K plugin is currently being used many major customers of Cisco. We will understand how customers are using the plugin and will review enhancement possibilities. This plugin will not be used by individuals who do not have access to a Cisco ASR1K router.

2. Scope and Features Of The Project:

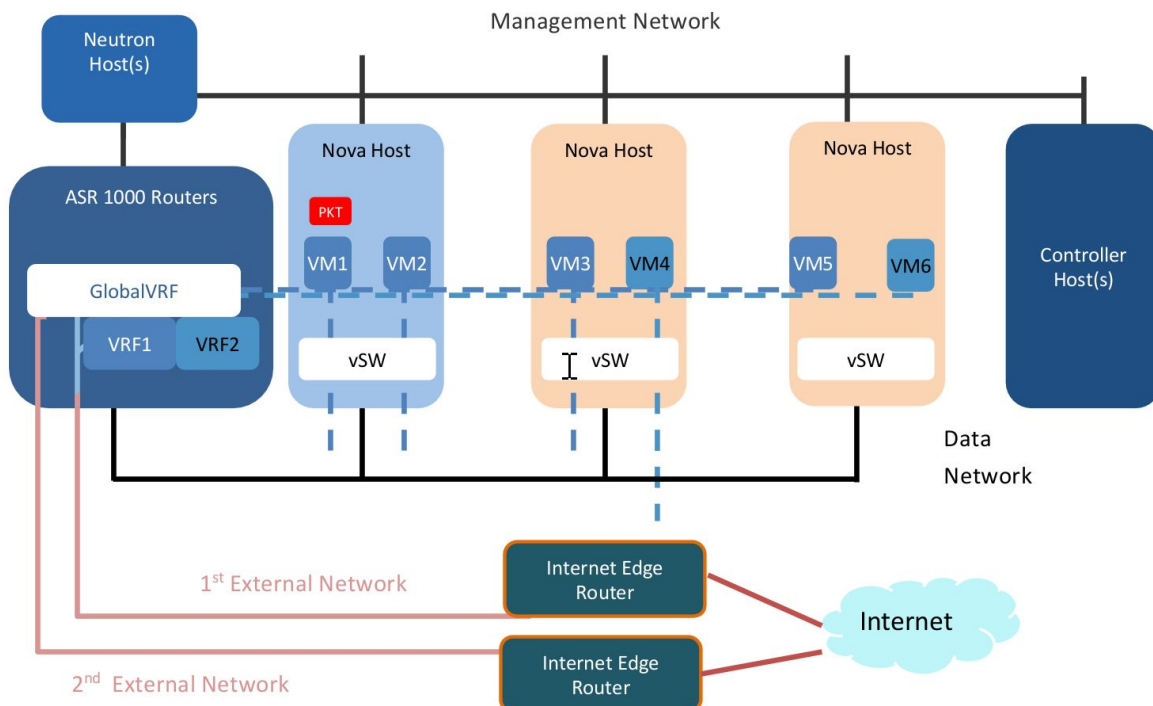


Figure 1 The diagram of the Network structure within OpenStack

As you can see from the Fig 1, ASR1K actually provides a gateway between inner Internet (Neutron) and external Internet. The plugins are something that will work between ASR1K and Neutron Host(s). Our goal is to enhance the plugins that will enable some of the features which are not available for the Neutron right now. So we won't work on anything outside of that.

What we'll work on

The features we will work on are listed as below.

1.Out of Band Config:

The Openstack ASR1K plugin implements openstack tenant router instances created by openstack neutron API calls. The tenant routers are implemented in the physical ASR1K's as virtual router forwarding (VRF) instances attached to tenant networks via VLAN subinterfaces.

Openstack cloud-administrators would like the capability to have the ASR1K plugin to create/delete some free-form configuration on their behalf during the neutron tenant router lifecycle events. This capability should rely on configuration scoping in the ASR1K. Example config scopes:

- Global config scope
- sub-interface config scope: specifically the interfaces created/deleted by the ASR1K plugin to attach to openstack tenant networks.
- VRFs mapping to tenant router instances.

2.Netflow:

Ability to turn on Netflow for subinterfaces being created by ASR1K plugin.

3.Firewall support:

We'll improve the way how the router authenticates a certain package source and admit its access to the inner network.

What we won't work on

What we're not going to do is changing the configuration of Neutron or ASR1K that have been set up by Cisco. We won't change things that work fine as there have already been existing customers for ASR1K.

3. Solution Concept

Global Architectural Structure Of the Project:

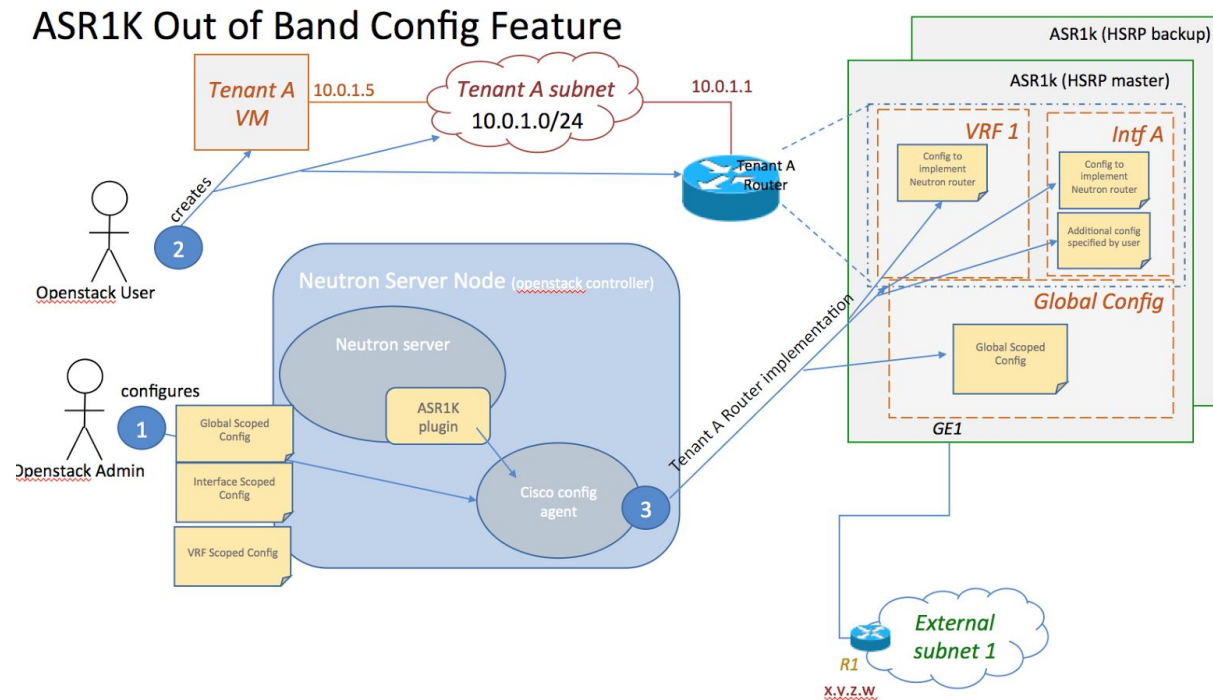


Figure 2 ASR1K out of band config feature

Figure 2 shows the Out of Band Config Feature we will aim to implement. The tentative steps that we will take include:

1. Admin configures the ASR1K plugin/config agent with additional config (template) to apply to the ASR1K along with the config to implement any tenant routers. (config is either via config agent config files OR via neutron API extension).
2. User creates a tenant A router
3. ASR1K plugin & config agent implements the tenant router via VRF1, Intf A, and global config in the ASR; ASR1K plugin & config agent apply additional config defined by user

The High-Level Description

- Dedicated channel for out-of-band configuration: Provide more availability for the out-of-band configuration.
- VLAN Groups: set up a group to determine who are privileged to configure.
- Configuration Library: Holds custom configuration sets

Design Implications and Discussion:

This section discusses the implications and reasons of the design decisions made during the global architecture design.

- Dedicated channel: If the admin want to configure the ASR1K during the peak network traffic period, it may meet problem because of the high collision and latency, a dedicated channel may perfectly solve this problem.
- Central deployment: In order to meet potential NetFlow requirement, we deploy the device central, which makes it easy to monitor the IP packets.

4. Acceptance Criteria

Minimum acceptance criteria is to add or edit some existing features in the current Openstack Neutron Cisco ASR1K plugin to provide additional functionality or support. These goals may include:

- Firewall support
- Out of band configuration
- Netflow configuration

Stretch Goals:

Stretch goals within these three fields have yet to be determined as our mentors are still working through different interests.

5. Release Planning

5.1 Basics with setup (due 02/23)

In this sprint we will get familiar with OpenStack. The points we will be working on are:

- Setup VM
- Devstack
- Try out Horizon & CLI
- Explore common workflows
- Create Router
- Create Network, subnet
- Attach interface to Router
- PyCharm
 - Install
 - Open neutron as project
 - Connect to neutron DB and inspect tables and their content
- Agile Best Practices

5.2 Some common workflows and get familiar with some tools (due 03/16)

The points we will be working on are:

- Freshen up on Ethernet, VLAN, routing, HTTP protocol, REST, tcpdump, Linux network namespaces, openvswitch
- Understand Plugin, Agent, Driver concepts
- Create VM
- Security Groups
- Attach Floating IP
- Traffic from Inside to Outside and Outside to inside
- Understand Basic ideas behind NAT
- Adding breakpoints in the neutron server and Agent and follow the code for one of the workflows above.
- Using Pycharm
 - Run neutron server in Pycharm
 - Set breakpoints and track processing of REST api call to create router
 - Remote debugging using Pycharm
- Understand the packet path from a VM to
 - Another VM on the same subnet
 - A VM on a different subnet
 - Destination outside the OpenStack cloud
- Understand the neutron extension/plugin/agent layers
- Understand Data models and look at mysql tables

5.3 Bringing up an ASR1k (due 03/30)

In this sprint we will implement what we did in sprint 1 on an ASR1k. The points we will be working on are:

- devstack environment to bring up an ASR1K
- Logging in to the ASR and looking at basic config
- Use IOS XE CLI
- Repeat common workflows, check impact on ASR config
- Understand the ASR1k Plugin and Config Agent
- Differences from the Community Implementation
- Follow the code for a simple workflow

5.4 Design for feature enhancement (due 04/13)

The points we will be working on are:

- Out of Band configuration
- Firewall
- Netflow
- Understand and define the changes for enhancement
- Design outline document covering the different pieces

5.5 Implementation (due 04/27)

In this sprint we will tie up any loose ends and do some unit testing if we have time.