Automatic allocation of network topologies

« (config-address-scopes.html) » (config-az.html) (https://bugs.launchpad.net/neutron/+filebug?field.title=Automatic%20allocation%20of%20network%20topologies%20in%20Neutron&field.comment=%0A%0A%0AThis bug tracker is for errors with the documentation, use the following as a template and remove or add fields as you see fit. Convert [ ] into [x] to check boxes:%0A%0A- [ ] This doc is inaccurate in this way: _____%0A- [ ] This is a doc addition request.%0A- [ ] I have a fix to the document that I can paste below including example: input and output. %0A%0AIf you have a troubleshooting or support issue, use the following resources:%0A%0A - Ask OpenStack: http://ask.openstack.org%0A - The mailing list: http://lists.openstack.org%0A - IRC: 'openstack' channel on Freenode%0A%0A----------%0ARelease:%2012.0.1.dev11%20on%202018-03-07%2021:05%0ASHA:%2043df2709acbdce86686a40b75fd34e96880427d0%0ASource:%20https://git.openstack.org/cgit/openstack/neutron/tree/doc/source/admin/config-auto-allocation.rst%0AURL: https://docs.openstack.org/neutron/queens/admin/config-auto-allocation.html&field.tags=doc)

UPDATED: 2018-03-07 21:05

The auto-allocation feature introduced in Mitaka simplifies the procedure of setting up an external connectivity for end-users, and is also known as **Get Me A Network**.

Previously, a user had to configure a range of networking resources to boot a server and get access to the Internet. For example, the following steps are required:

- Create a network
- Create a subnet
- Create a router
- Uplink the router on an external network
- Downlink the router on the previously created subnet

These steps need to be performed on each logical segment that a VM needs to be connected to, and may require networking knowledge the user might not have.

This feature is designed to automate the basic networking provisioning for projects. The steps to provision a basic network are run during instance boot, making the networking setup hands-free.

To make this possible, provide a default external network and default subnetpools (one for IPv4, or one for IPv6, or one of each) so that the Networking service can choose what to do in lieu of input. Once these are in place, users can boot their VMs without specifying any networking details. The Compute service will then use this feature automatically to wire user VMs.

Enabling the deployment for auto-allocation¶

To use this feature, the neutron service must have the following extensions enabled:

- **auto-allocated-topology**
- **subnet_allocation**
- **external-net**
- **router**

Before the end-user can use the auto-allocation feature, the operator must create the resources that will be used for the auto-allocated network topology creation. To perform this task, proceed with the following steps:

1. Set up a default external network

   Setting up an external network is described in OpenStack Networking Guide (./archives/adv-features.html). Assuming the external network to be used for the auto-allocation feature is named **public**, make it the **default** external network with the following command:

   ```
   $ openstack network set public --default
   ```

   > **Note**
   >
   > The flag **--default** (and **--no-default** flag) is only effective with external networks and has no effects on regular (or internal) networks.

2. Create default subnetpools

   The auto-allocation feature requires at least one default subnetpool. One for IPv4, or one for IPv6, or one of each.

```
$ openstack subnet pool create --share --default \
  --pool-prefix 192.0.2.0/24 --default-prefix-length 26 \
  shared-default

  +-------------------+--------------------------------------+
  | Field             | Value                                |
  +-------------------+--------------------------------------+
  | address_scope_id  | None                                 |
  | created_at        | 2017-01-12T15:10:34Z                 |
  | default_prefixlen | 26                                   |
  | default_quota     | None                                 |
  | description       |                                      |
  | headers           |                                      |
  | id                | b41b7b9c-de57-4c19-b1c5-731985bceb7f |
  | ip_version        | 4                                    |
  | is_default        | True                                 |
  | max_prefixlen     | 32                                   |
  | min_prefixlen     | 8                                    |
  | name              | shared-default                       |
  | prefixes          | 192.0.2.0/24                         |
  | project_id        | 86acdbd1d72745fd8e8320edd7543400     |
  | revision_number   | 1                                    |
  | shared            | True                                 |
  | tags              | []                                   |
  | updated_at        | 2017-01-12T15:10:34Z                 |
  +-------------------+--------------------------------------+

$ openstack subnet pool create --share --default \
  --pool-prefix 2001:db8:8000::/48 --default-prefix-length 64 \
  default-v6

+-------------------+--------------------------------------+
| Field             | Value                                |
+-------------------+--------------------------------------+
| address_scope_id  | None                                 |
| created_at        | 2017-01-12T15:14:35Z                 |
| default_prefixlen | 64                                   |
| default_quota     | None                                 |
| description       |                                      |
| headers           |                                      |
| id                | 6f387016-17f0-4564-96ad-e34775b6ea14 |
| ip_version        | 6                                    |
| is_default        | True                                 |
| max_prefixlen     | 128                                  |
| min_prefixlen     | 64                                   |
| name              | default-v6                           |
| prefixes          | 2001:db8:8000::/48                   |
| project_id        | 86acdbd1d72745fd8e8320edd7543400     |
| revision_number   | 1                                    |
| shared            | True                                 |
| tags              | []                                   |
| updated_at        | 2017-01-12T15:14:35Z                 |
+-------------------+--------------------------------------+
```

Get Me A Network¶

In a deployment where the operator has set up the resources as described above, they can get their auto-allocated network topology as follows:

```
$ openstack network auto allocated topology create --or-show
+------------+--------------------------------------+
| Field      | Value                                |
+------------+--------------------------------------+
| id         | a380c780-d6cd-4510-a4c0-1a6ec9b85a29 |
| name       | None                                 |
| project_id | cfd1889ac7d64ad891d4f20aef9f8d7c     |
+------------+--------------------------------------+
```

Note

When the **`--or-show`** option is used the command returns the topology information if it already exists.

Operators (and users with admin role) can get the auto-allocated topology for a project by specifying the project ID:

```
$ openstack network auto allocated topology create --project \
  cfd1889ac7d64ad891d4f20aef9f8d7c --or-show
+------------+--------------------------------------+
| Field      | Value                                |
+------------+--------------------------------------+
| id         | a380c780-d6cd-4510-a4c0-1a6ec9b85a29 |
| name       | None                                 |
| project_id | cfd1889ac7d64ad891d4f20aef9f8d7c     |
+------------+--------------------------------------+
```

The ID returned by this command is a network which can be used for booting a VM.

```
$ openstack server create --flavor m1.small --image \
  cirros-0.3.5-x86_64-uec --nic \
  net-id=8b835bfb-cae2-4acc-b53f-c16bb5f9a7d0 vm1
```

The auto-allocated topology for a user never changes. In practice, when a user boots a server omitting the **--nic** option, and there is more than one network available, the Compute service will invoke the API behind **auto allocated topology create**, fetch the network UUID, and pass it on during the boot process.

Validating the requirements for auto-allocation¶

To validate that the required resources are correctly set up for auto-allocation, without actually provisioning anything, use the **--check-resources** option:

```
$ openstack network auto allocated topology create --check-resources
Deployment error: No default router:external network.

$ openstack network set public --default

$ openstack network auto allocated topology create --check-resources
Deployment error: No default subnetpools defined.

$ openstack subnet pool set shared-default --default

$ openstack network auto allocated topology create --check-resources
+---------+-------+
| Field   | Value |
+---------+-------+
| dry-run | pass  |
+---------+-------+
```

The validation option behaves identically for all users. However, it is considered primarily an admin or service utility since it is the operator who must set up the requirements.

Project resources created by auto-allocation¶

The auto-allocation feature creates one network topology in every project where it is used. The auto-allocated network topology for a project contains the following resources:

Resource	Name
network	**auto_allocated_network**
subnet (IPv4)	**auto_allocated_subnet_v4**
subnet (IPv6)	**auto_allocated_subnet_v6**
router	**auto_allocated_router**

Compatibility notes¶

Nova uses the **auto allocated topology** feature with API micro version 2.37 or later. This is because, unlike the neutron feature which was implemented in the Mitaka release, the integration for nova was completed during the Newton release cycle. Note that the CLI option **--nic** can be omitted regardless of the microversion used as long as there is no more than one network available to the project, in which case nova fails with a 400 error because it does not know which network to use. Furthermore, nova does not start using the feature, regardless of whether or not a user requests micro version 2.37 or later, unless all of the **nova-compute** services are running Newton-level code.

« (config-address-scopes.html) » (config-az.html)

UPDATED: 2018-03-07 21:05

FOUND AN ERROR? REPORT A BUG (HTTPS://BUGS.LAUNCHPAD.NET/NEUTRON/+FILEBUG?FIELD.TITLE=AUTOMATIC%20ALLOCATION%20OF%20NETWORK%20TOPOLOGIES%20IN%20NEUTRON&FIELD.COMMENT=%0A%0A%0ATHIS BUG TRACKER IS FOR ERRORS WITH THE DOCUMENTATION, USE THE FOLLOWING AS A TEMPLATE AND REMOVE OR ADD FIELDS AS YOU SEE FIT. CONVERT [ ] INTO [X] TO CHECK BOXES:%0A%0A- [ ] THIS DOC IS INACCURATE IN THIS WAY: _____%0A- [ ] THIS IS A DOC ADDITION REQUEST.%0A- [ ] I HAVE A FIX TO THE DOCUMENT THAT I CAN PASTE BELOW INCLUDING EXAMPLE: INPUT AND OUTPUT. %0A%0AIF YOU HAVE A TROUBLESHOOTING OR SUPPORT ISSUE, USE THE FOLLOWING RESOURCES:%0A%0A - ASK OPENSTACK: HTTP://ASK.OPENSTACK.ORG%0A - THE MAILING LIST: HTTP://LISTS.OPENSTACK.ORG%0A - IRC: 'OPENSTACK' CHANNEL ON FREENODE%0A%0A----------%0ARELEASE:%2012.0.1.DEV11%20ON%202018-03-07%2021:05%0ASHA:%2043DF2709ACBDCE86686A40B75FD34E96880427D0%0ASOURCE:%20HTTPS://GIT.OPENSTACK.ORG/CGIT/OPENSTACK/NEUTRON/TREE/DOC/SOURCE/ADMIN/CONFIG-AUTO-ALLOCATION.RST%0AURL: HTTPS://DOCS.OPENSTACK.ORG/NEUTRON/QUEENS/ADMIN/CONFIG-AUTO-ALLOCATION.HTML&FIELD.TAGS=DOC)

QUESTIONS? (HTTP://ASK.OPENSTACK.ORG)

Page Contents