UFAZ
FRENCH-AZERBAIJANI UNIVERSITY
Université
de Strasbourg

Networking Network Services

Project Work

OpenStack

Team members:

➢ **Rufat Isgandarov**
➢ **Yusif Hajizada**
➢ **Vugar Hasanov**

04.05.2025

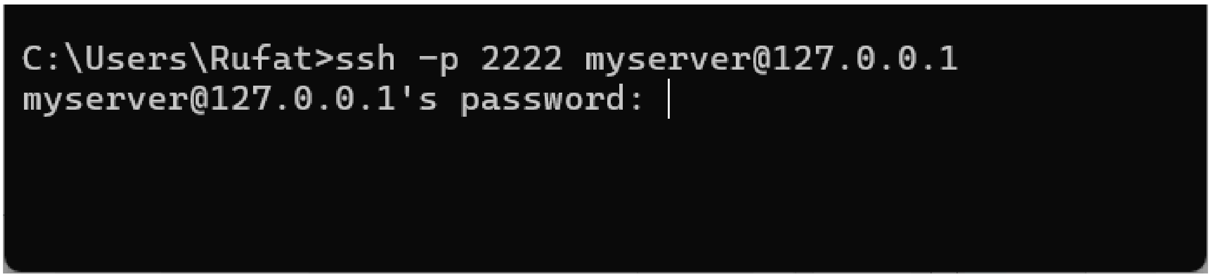
Introduction

In this project, we wanted to learn how to install and run OpenStack using Kolla-Ansible in a virtual environment Oracle Virtualbox. OpenStack is a widely used open-source cloud computing platform that enables the creation and management of both public and private clouds. The objective of this work was to gain practical experience with the installation and configuration of OpenStack services, as well as to understand how its core components interact to provide Infrastructure-as-a-Service (IaaS). So we decided to give it a try. We used a VirtualBox VM, installed Ubuntu, and followed the instructions to set everything up inside the VM. We used commands, config files, YAMLs. We followed structured steps including system preparation, network configuration, dependency installation, and the execution of Kolla-Ansible commands to build the OpenStack environment. We learned how different OpenStack services work together.Through this process, we became familiar with key services like Keystone, Nova, Neutron, Glance, and Horizon.

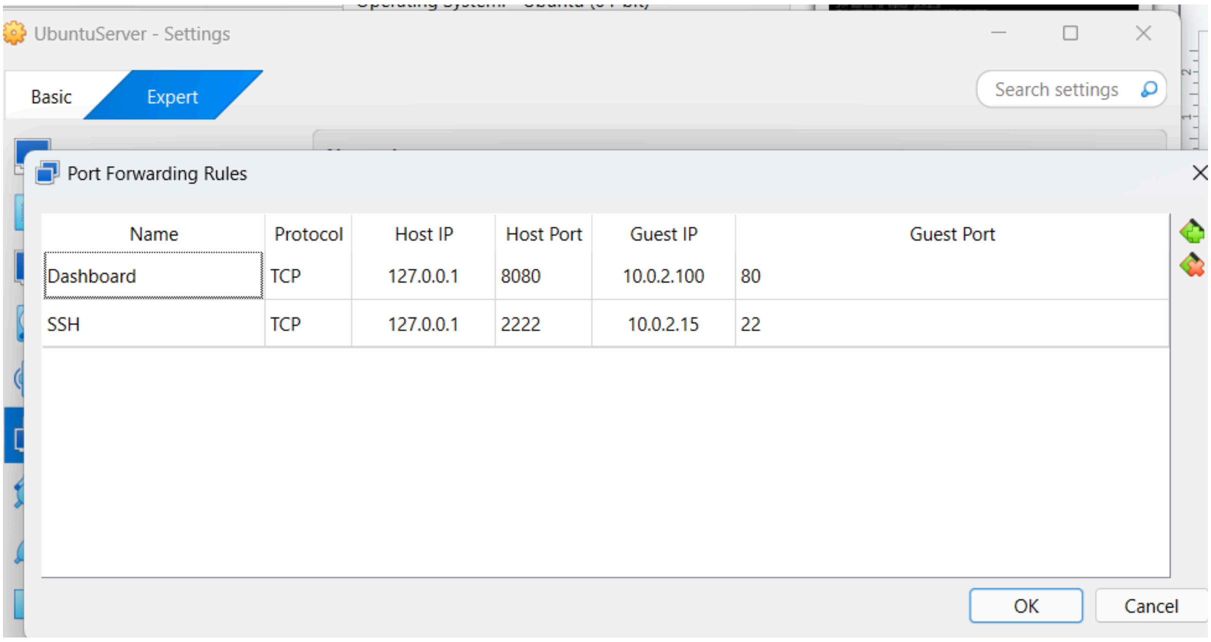
Main

Install dependencies:

First of all, we specified Port Forwarding to connect VM with ssh from Guest machine terminal. We opened up VirtualBox's settings for our Ubuntu VM and went to the "Network" tab and configured it related to specification in lab description.This setup allows us to connect to the VM using SSH by typing the command *ssh ubuntu@localhost -p 2222* from our main computer. This made it easier for us to use the terminal without needing to open the VirtualBox window every time.

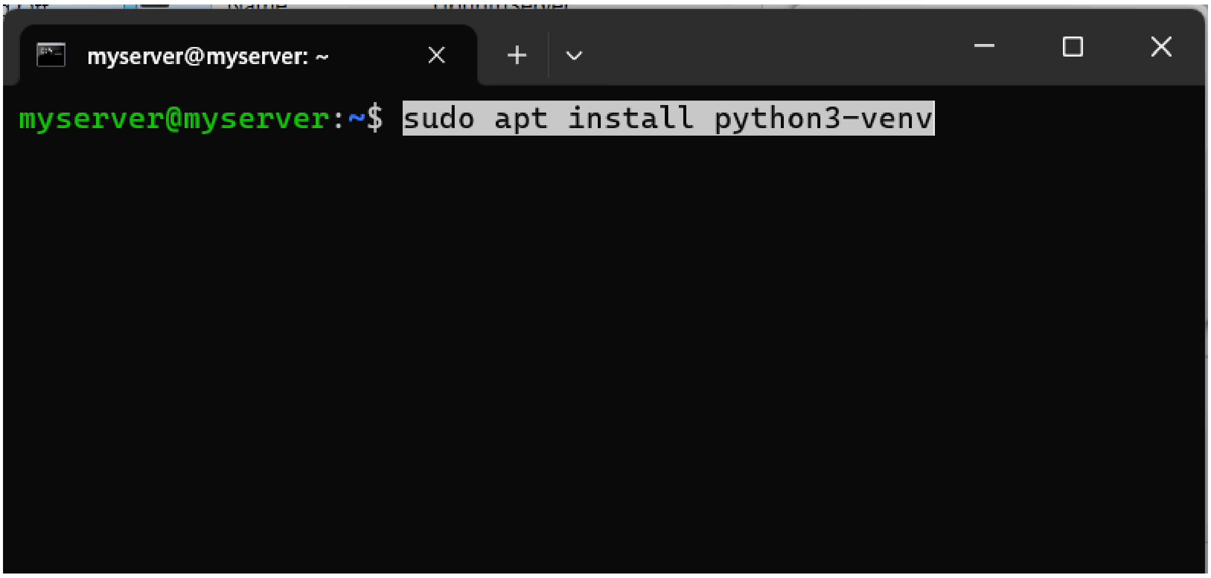


```
C:\Users\Rufat>ssh -p 2222 myserver@127.0.0.1
myserver@127.0.0.1's password:
```



Name	Protocol	Host IP	Host Port	Guest IP	Guest Port
Dashboard	TCP	127.0.0.1	8080	10.0.2.100	80
SSH	TCP	127.0.0.1	2222	10.0.2.15	22

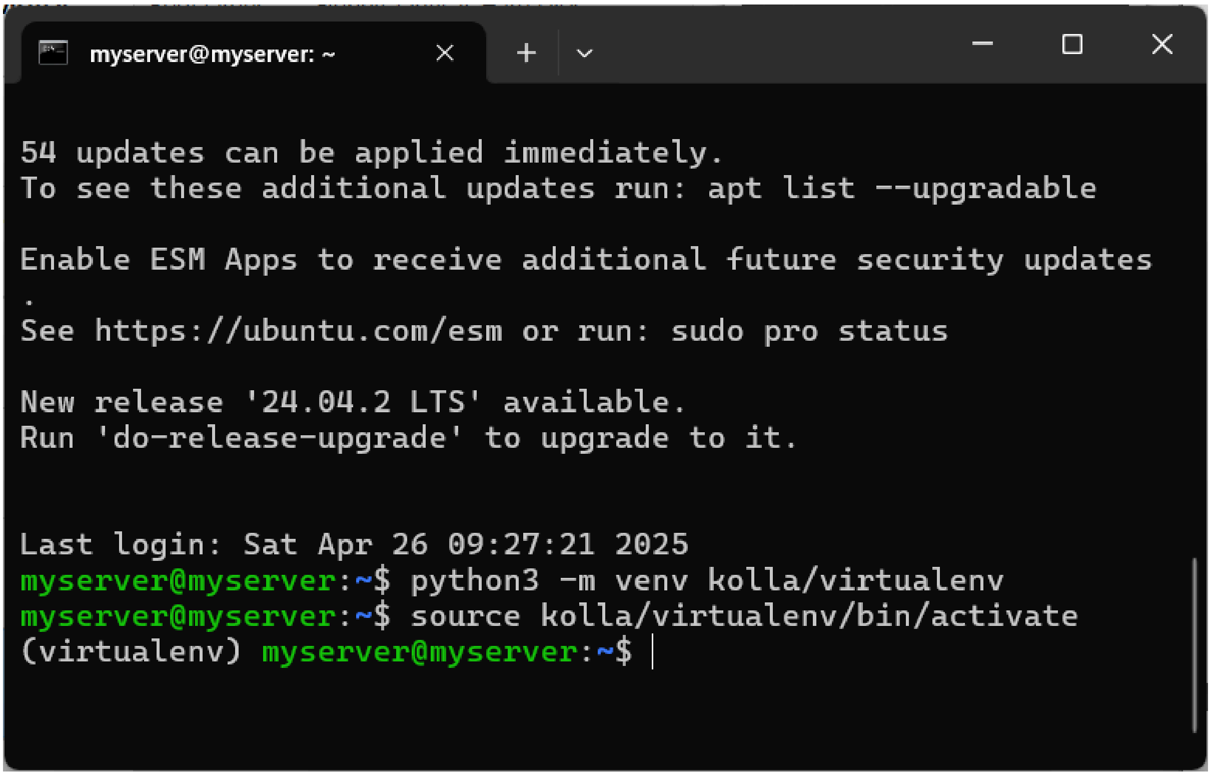
Now we Installing the virtual environment dependencies.



```
myserver@myserver:~$ sudo apt install python3-venv
```

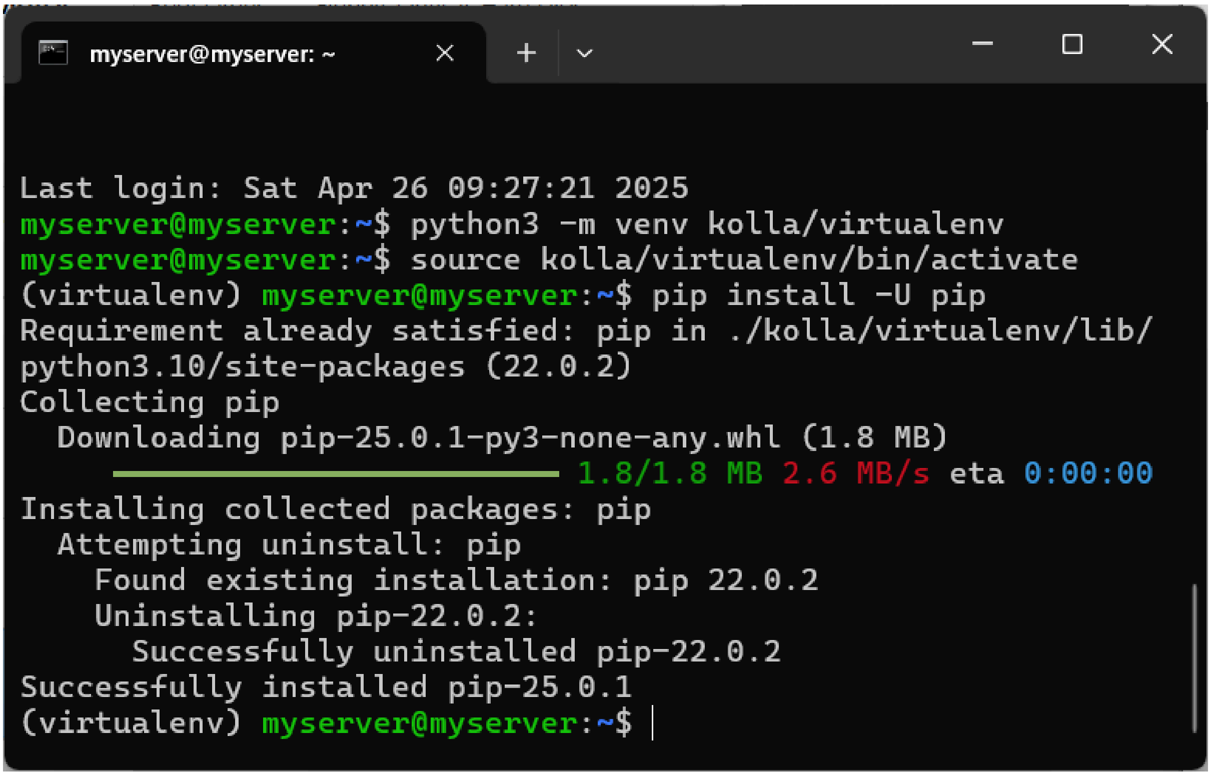
Create a virtual environment and activate it:

Once we were in the VM, we created a new Python virtual environment using this command: python3 *-m venv kolla/virtualenv* This command created a folder where Python packages can be installed separately from the system-wide packages. Then we did source *kolla/virtualenv/bin/activate* so that python and pip now point into that venv.



```
54 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates
.
See https://ubuntu.com/esm or run: sudo pro status

New release '24.04.2 LTS' available.
Run 'do-release-upgrade' to upgrade to it.


Last login: Sat Apr 26 09:27:21 2025
myserver@myserver:~$ python3 -m venv kolla/virtualenv
myserver@myserver:~$ source kolla/virtualenv/bin/activate
(virtualenv) myserver@myserver:~$ 
```

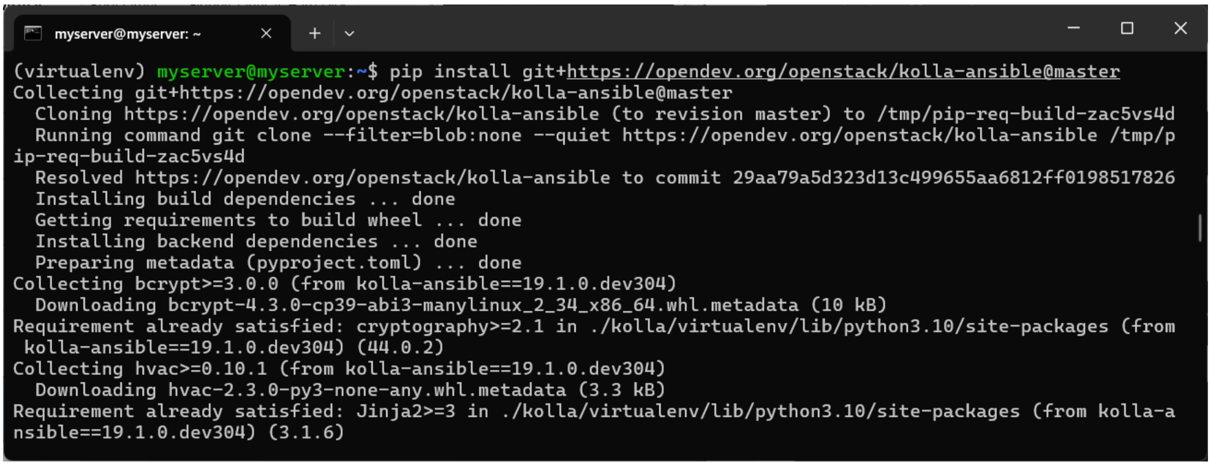
We noticed that the default pip version was outdated, so we decided to upgrade it to avoid issues during installation. We ran: *pip install -U pip* This updated pip to the latest version inside the virtual environment.



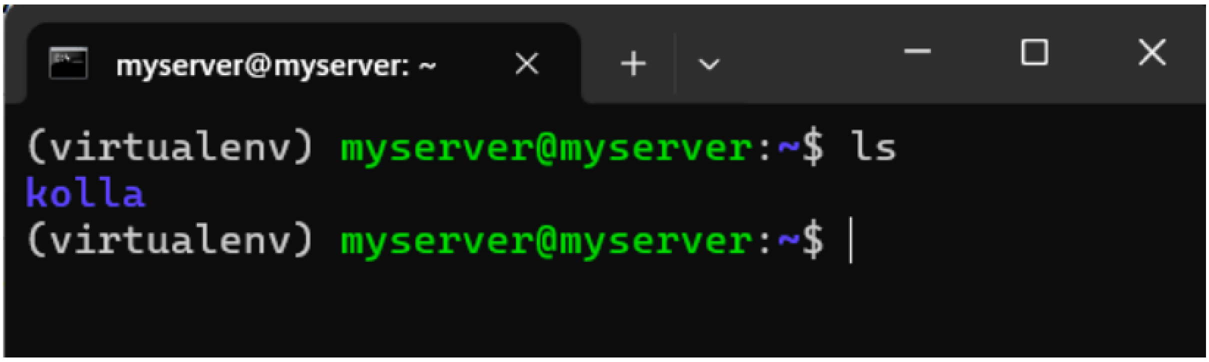
```
Last login: Sat Apr 26 09:27:21 2025
myserver@myserver:~$ python3 -m venv kolla/virtualenv
myserver@myserver:~$ source kolla/virtualenv/bin/activate
(virtualenv) myserver@myserver:~$ pip install -U pip
Requirement already satisfied: pip in ./kolla/virtualenv/lib/
python3.10/site-packages (22.0.2)
Collecting pip
  Downloading pip-25.0.1-py3-none-any.whl (1.8 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 1.8/1.8 MB 2.6 MB/s eta 0:00:00
Installing collected packages: pip
  Attempting uninstall: pip
    Found existing installation: pip 22.0.2
    Uninstalling pip-22.0.2:
      Successfully uninstalled pip-22.0.2
Successfully installed pip-25.0.1
(virtualenv) myserver@myserver:~$ 
```

Install Kolla-ansible. Install kolla-ansible and its dependencies using pip. With pip ready, we installed Kolla-Ansible directly from the official OpenDev repository. The command we used was:

pip install git+https://opendev.org/openstack/kolla-ansible@master
This downloaded and installed the latest Kolla-Ansible package and all its dependencies.

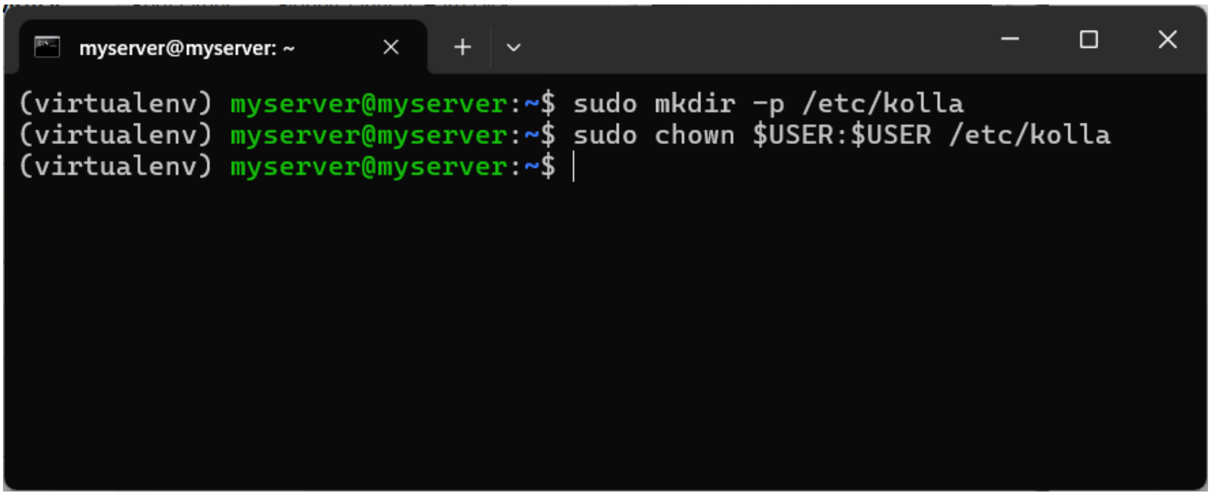


```
(virtualenv) myserver@myserver:~$ pip install git+https://opendev.org/openstack/kolla-ansible@master
Collecting git+https://opendev.org/openstack/kolla-ansible@master
  Cloning https://opendev.org/openstack/kolla-ansible (to revision master) to /tmp/pip-req-build-zac5vs4d
  Running command git clone --filter=blob:none --quiet https://opendev.org/openstack/kolla-ansible /tmp/p
ip-req-build-zac5vs4d
  Resolved https://opendev.org/openstack/kolla-ansible to commit 29aa79a5d323d13c499655aa6812ff0198517826
  Installing build dependencies ... done
  Getting requirements to build wheel ... done
  Installing backend dependencies ... done
  Preparing metadata (pyproject.toml) ... done
Collecting bcrypt>=3.0.0 (from kolla-ansible==19.1.0.dev304)
  Downloading bcrypt-4.3.0-cp39-abi3-manylinux_2_34_x86_64.whl.metadata (10 kB)
Requirement already satisfied: cryptography>=2.1 in ./kolla/virtualenv/lib/python3.10/site-packages (from
 kolla-ansible==19.1.0.dev304) (44.0.2)
Collecting hvac>=0.10.1 (from kolla-ansible==19.1.0.dev304)
  Downloading hvac-2.3.0-py3-none-any.whl.metadata (3.3 kB)
Requirement already satisfied: Jinja2>=3 in ./kolla/virtualenv/lib/python3.10/site-packages (from kolla-a
nsible==19.1.0.dev304) (3.1.6)
```



```
(virtualenv) myserver@myserver:~$ ls
kolla
(virtualenv) myserver@myserver:~$
```

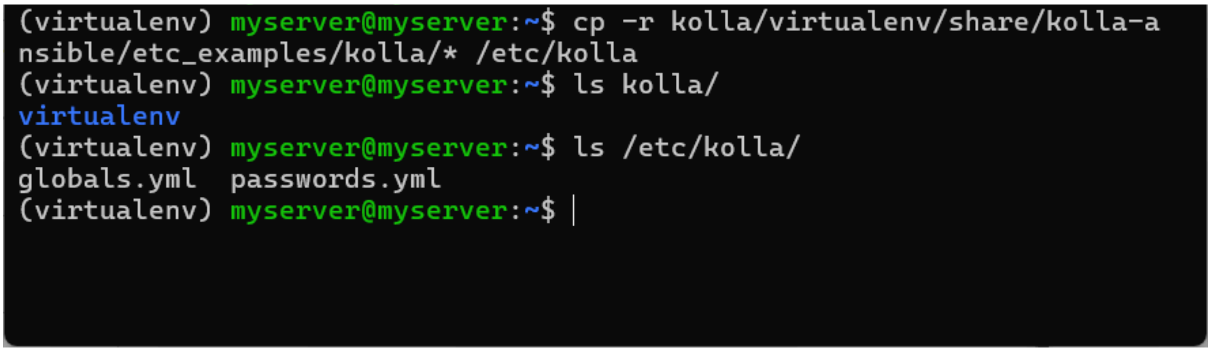
Next, we created the folder */etc/kolla* which will store all our configuration files. We did this using: This gave our user the right permissions to edit files inside */etc/kolla*.



```
(virtualenv) myserver@myserver:~$ sudo mkdir -p /etc/kolla
(virtualenv) myserver@myserver:~$ sudo chown $USER:$USER /etc/kolla
(virtualenv) myserver@myserver:~$
```

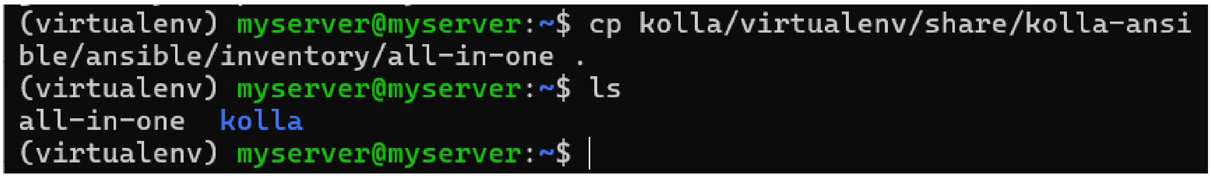
We copied the default example configuration files from the Kolla-Ansible installation directory to our newly created */etc/kolla* folder.
Command used:

cp -r kolla/virtualenv/share/kolla-ansible/etc_examples/kolla/ /etc/kolla*
After this, we had globals.yml and passwords.yml ready to edit.



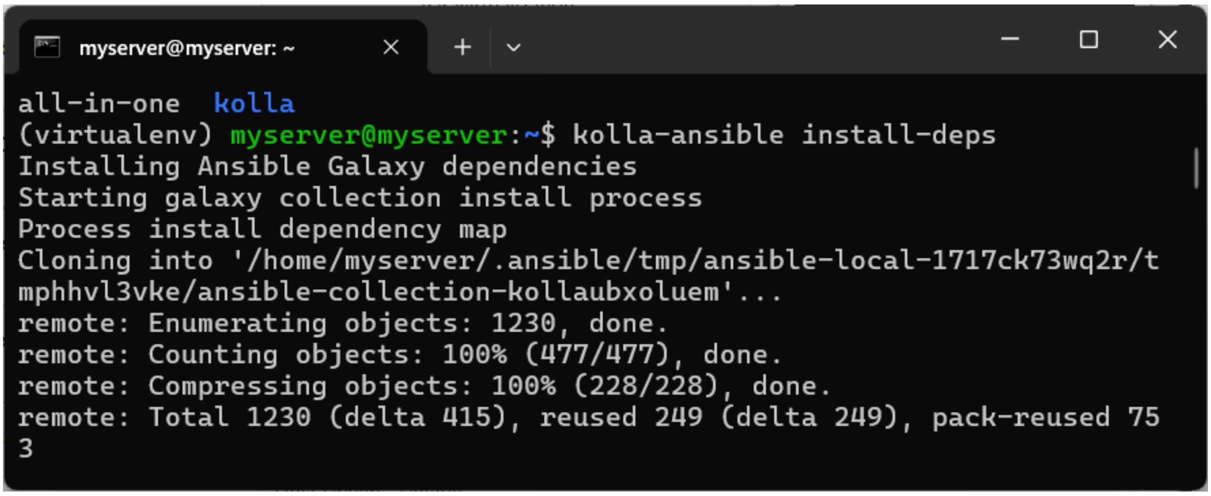
```
(virtualenv) myserver@myserver:~$ cp -r kolla/virtualenv/share/kolla-a
nsible/etc_examples/kolla/* /etc/kolla
(virtualenv) myserver@myserver:~$ ls kolla/
virtualenv
(virtualenv) myserver@myserver:~$ ls /etc/kolla/
globals.yml  passwords.yml
(virtualenv) myserver@myserver:~$ |
```

To make the deployment process easier, we copied the all-in-one inventory file to our home directory:
cp kolla/virtualenv/share/kolla-ansible/ansible/inventory/all-in-one .
This file is used by Ansible to know which hosts to install services on. Since we are using a single machine for everything, this file fits our needs perfectly.



```
(virtualenv) myserver@myserver:~$ cp kolla/virtualenv/share/kolla-ansi
ble/ansible/inventory/all-in-one .
(virtualenv) myserver@myserver:~$ ls
all-in-one  kolla
(virtualenv) myserver@myserver:~$ |
```

Install Ansible Galaxy requirements. Kolla-Ansible uses some Ansible roles and plugins that are hosted on Ansible Galaxy. We installed them using: *kolla-ansible install-deps*
This downloaded all required roles.



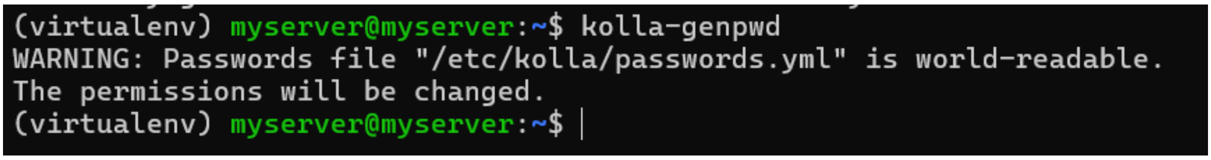
```
all-in-one  kolla
(virtualenv) myserver@myserver:~$ kolla-ansible install-deps
Installing Ansible Galaxy dependencies
Starting galaxy collection install process
Process install dependency map
Cloning into '/home/myserver/.ansible/tmp/ansible-local-1717ck73wq2r/t
mphhvl3vke/ansible-collection-kollaubxoluem'...
remote: Enumerating objects: 1230, done.
remote: Counting objects: 100% (477/477), done.
remote: Compressing objects: 100% (228/228), done.
remote: Total 1230 (delta 415), reused 249 (delta 249), pack-reused 75
3
```

Prepare initial configuration:
We ran *kolla-genpwd* to generate strong random passwords automatically and save them into */etc/kolla/passwords.yml*. This avoids

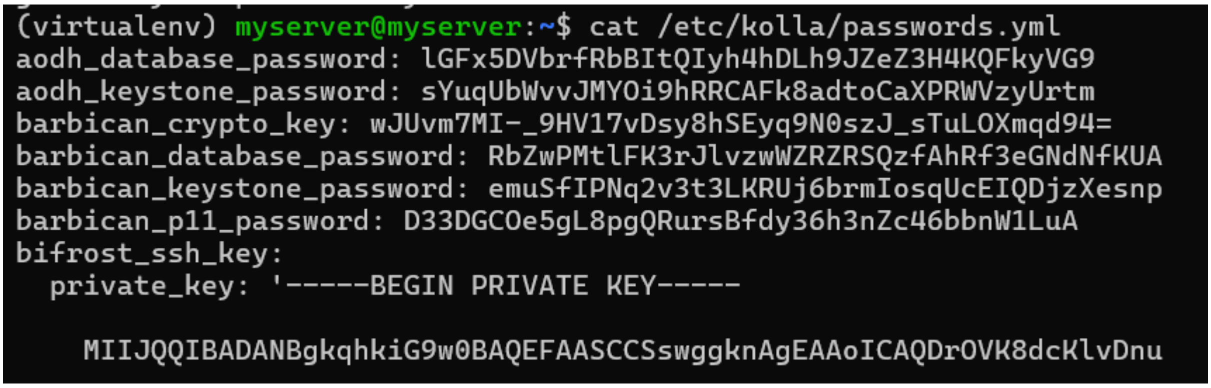
us having to set all the passwords manually. We checked the contents using cat to make sure everything was generated correctly.

Kolla passwords:

Passwords used in our deployment are stored in */etc/kolla/passwords.yml file.*



```
(virtualenv) myserver@myserver:~$ kolla-genpwd
WARNING: Passwords file "/etc/kolla/passwords.yml" is world-readable.
The permissions will be changed.
(virtualenv) myserver@myserver:~$ 
```



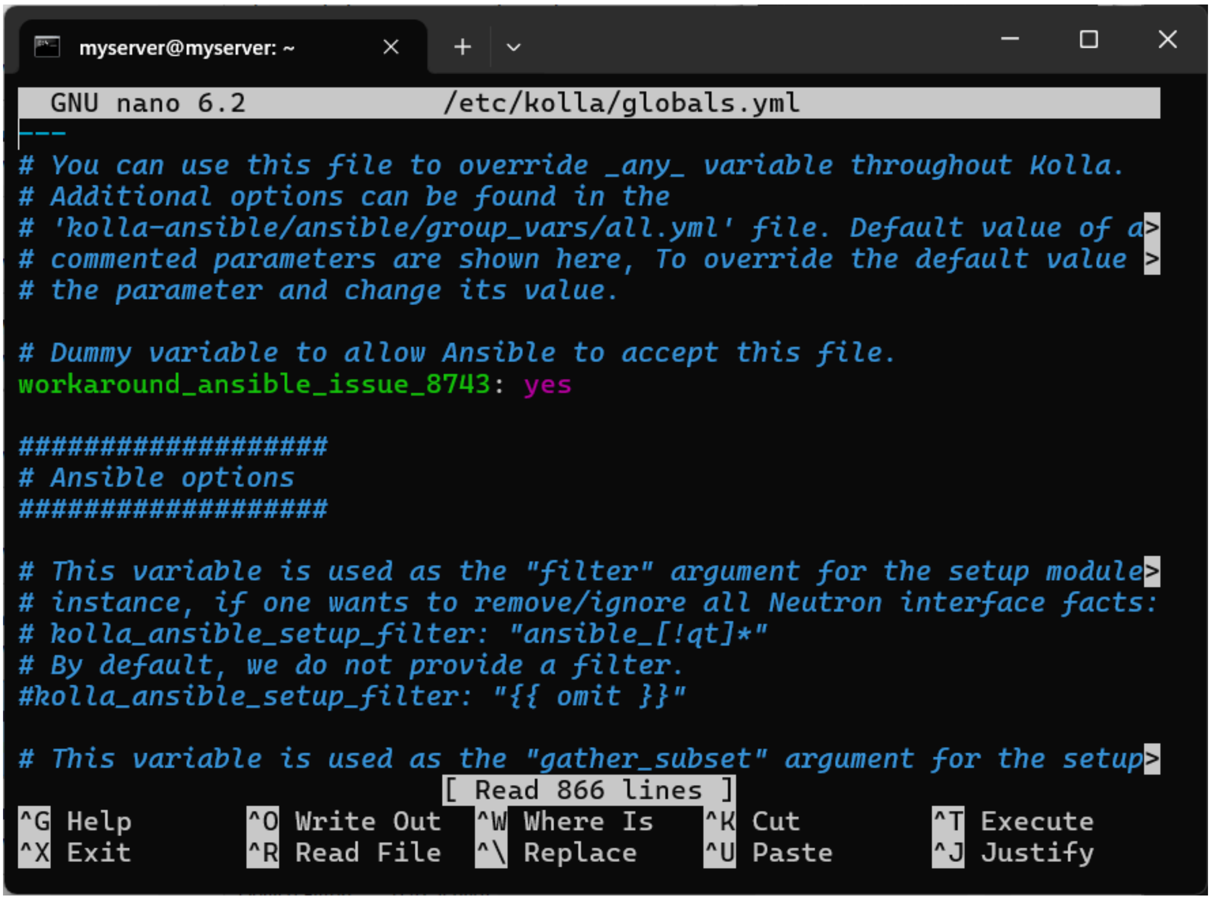
```
(virtualenv) myserver@myserver:~$ cat /etc/kolla/passwords.yml
aodh_database_password: lGFx5DVbrfRbBItQIyh4hDLh9JZeZ3H4KQFkyVG9
aodh_keystone_password: sYuqUbWvvJMYOi9hRRCAFk8adtoCaXPRWVzyUrtm
barbican_crypto_key: wJUvm7MI-_9HV17vDsy8hSEyq9N0szJ_sTuLOXmqd94=
barbican_database_password: RbZwPMtlFK3rJlvzwWZRZRSQzfAhRf3eGNdNfKUA
barbican_keystone_password: emuSfIPNq2v3t3LKRUj6brmIosqUcEIQDjzXesnp
barbican_p11_password: D33DGCOe5gL8pgQRursBfdy36h3nZc46bbnW1LuA
bifrost_ssh_key:
  private_key: '-----BEGIN PRIVATE KEY-----

    MIIJQQIBADANBgkqhkiG9w0BAQEFAASCCSswggknAgEAAoICAQDrOVK8dcKlvDnu
```

Kolla globals.yml globals.yml is the main configuration file for Kolla Ansible.

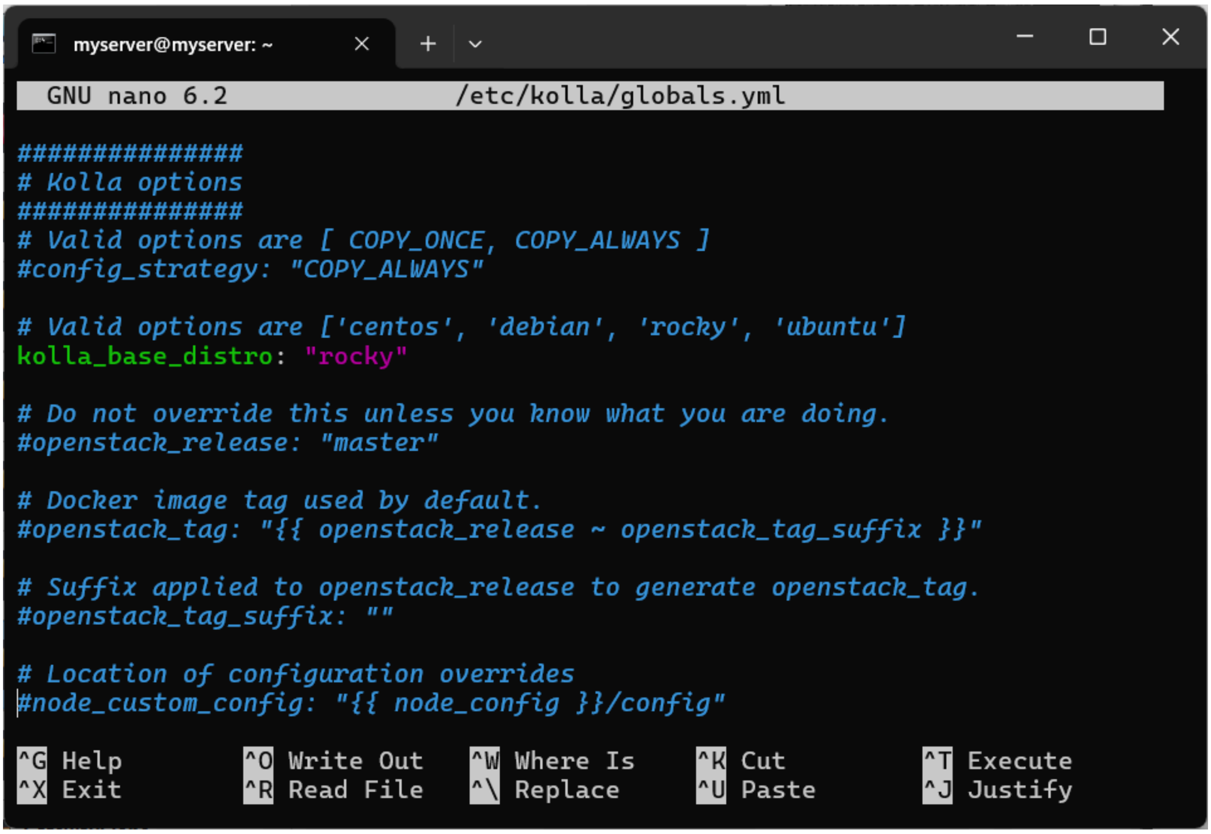
We opened the globals.yml file and set the following important settings:

- ➢ kolla_base_distro: "rocky"
- ➢ network_interface: "eth0"
- ➢ neutron_external_interface: "eth1"
- ➢ kolla_internal_vip_address: "10.0.0.50"

These settings define the Linux distribution we use, the interface for internal communication, and the interface for external access. We also set a virtual IP that services will use inside the network.

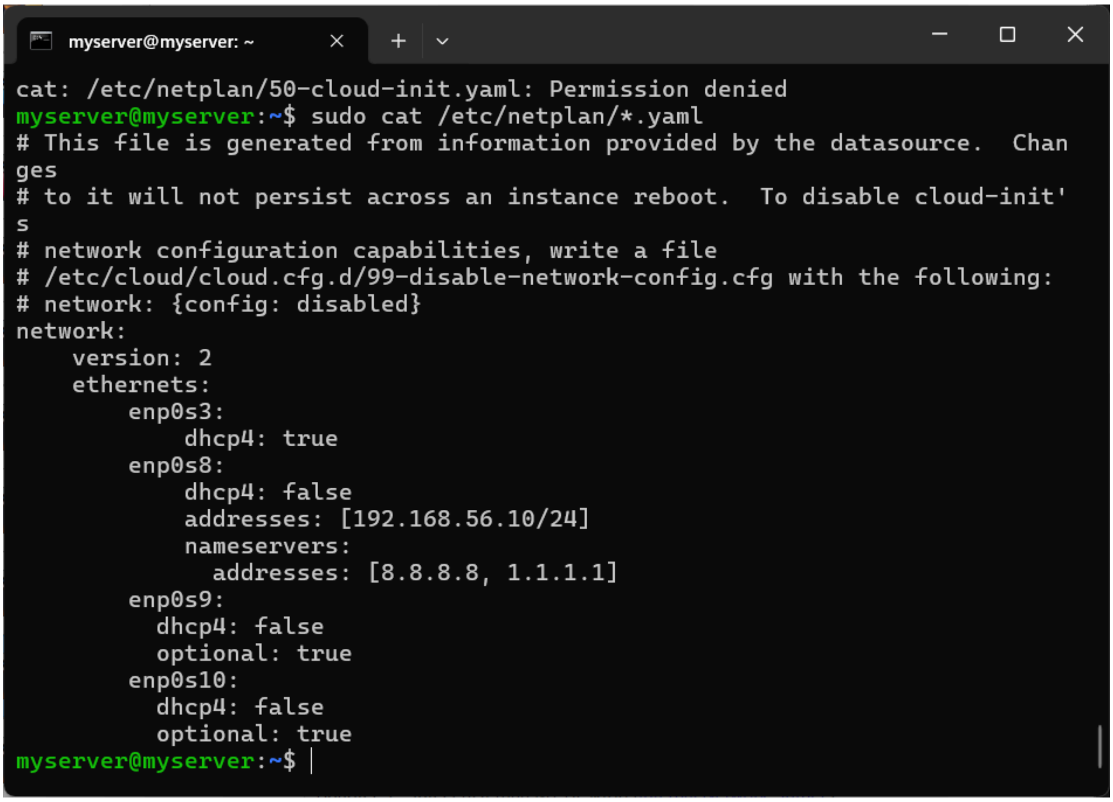


```
  GNU nano 6.2                /etc/kolla/globals.yml
---
# You can use this file to override _any_ variable throughout Kolla.
# Additional options can be found in the
# 'kolla-ansible/ansible/group_vars/all.yml' file. Default value of a>
# commented parameters are shown here, To override the default value >
# the parameter and change its value.

# Dummy variable to allow Ansible to accept this file.
workaround_ansible_issue_8743: yes

###################
# Ansible options
###################

# This variable is used as the "filter" argument for the setup module>
# instance, if one wants to remove/ignore all Neutron interface facts:
# kolla_ansible_setup_filter: "ansible_[!qt]*"
# By default, we do not provide a filter.
#kolla_ansible_setup_filter: "{{ omit }}"

# This variable is used as the "gather_subset" argument for the setup>
                           [ Read 866 lines ]
^G Help          ^O Write Out    ^W Where Is     ^K Cut          ^T Execute
^X Exit          ^R Read File    ^\ Replace      ^U Paste        ^J Justify
```

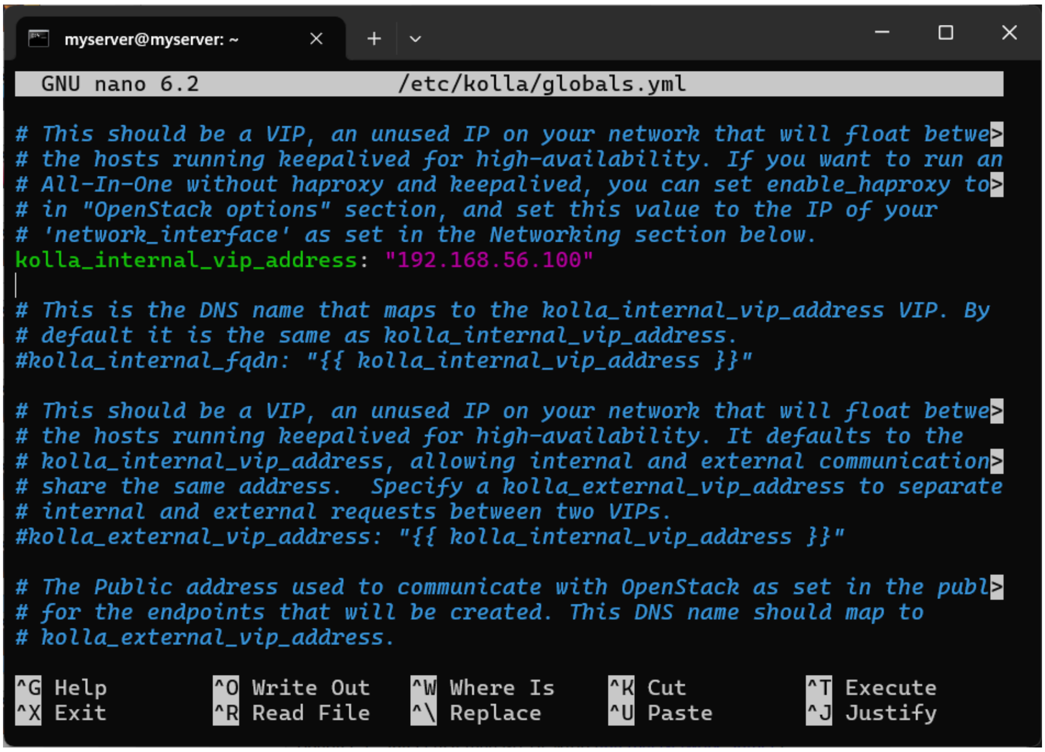


```
  GNU nano 6.2                /etc/kolla/globals.yml

###############
# Kolla options
###############
# Valid options are [ COPY_ONCE, COPY_ALWAYS ]
#config_strategy: "COPY_ALWAYS"

# Valid options are ['centos', 'debian', 'rocky', 'ubuntu']
kolla_base_distro: "rocky"

# Do not override this unless you know what you are doing.
#openstack_release: "master"

# Docker image tag used by default.
#openstack_tag: "{{ openstack_release ~ openstack_tag_suffix }}"

# Suffix applied to openstack_release to generate openstack_tag.
#openstack_tag_suffix: ""

# Location of configuration overrides
#node_custom_config: "{{ node_config }}/config"

^G Help          ^O Write Out    ^W Where Is     ^K Cut          ^T Execute
^X Exit          ^R Read File    ^\ Replace      ^U Paste        ^J Justify
```

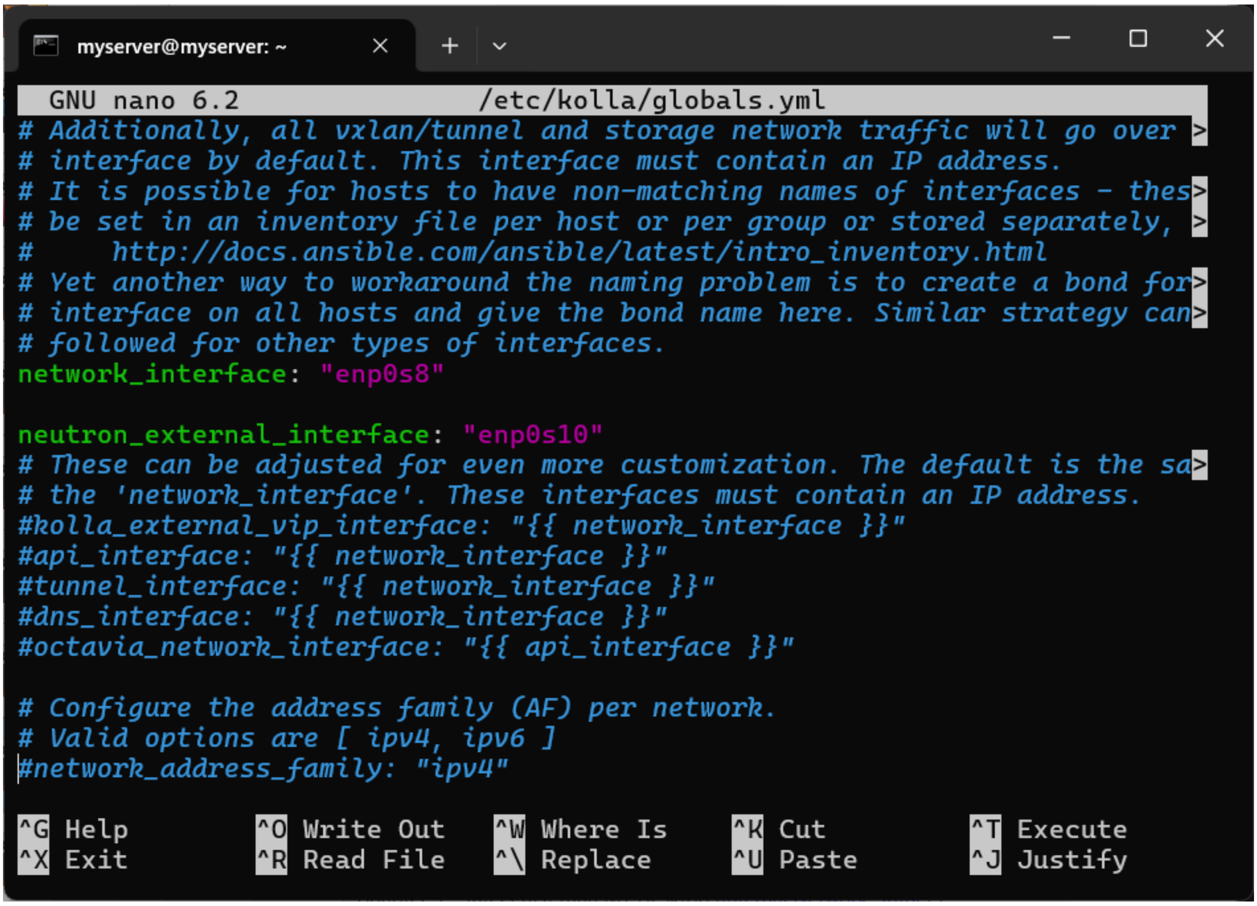
Netplan configuration:
To make sure our VM had a fixed IP, we edited *\/etc/netplan/01-netcfg.yaml* and added static IP configuration for eth0. We included the IP address, netmask, gateway, and DNS server. After saving the file, we applied the settings using sudo netplan apply. This step ensured our server always keeps the same IP address after reboot.



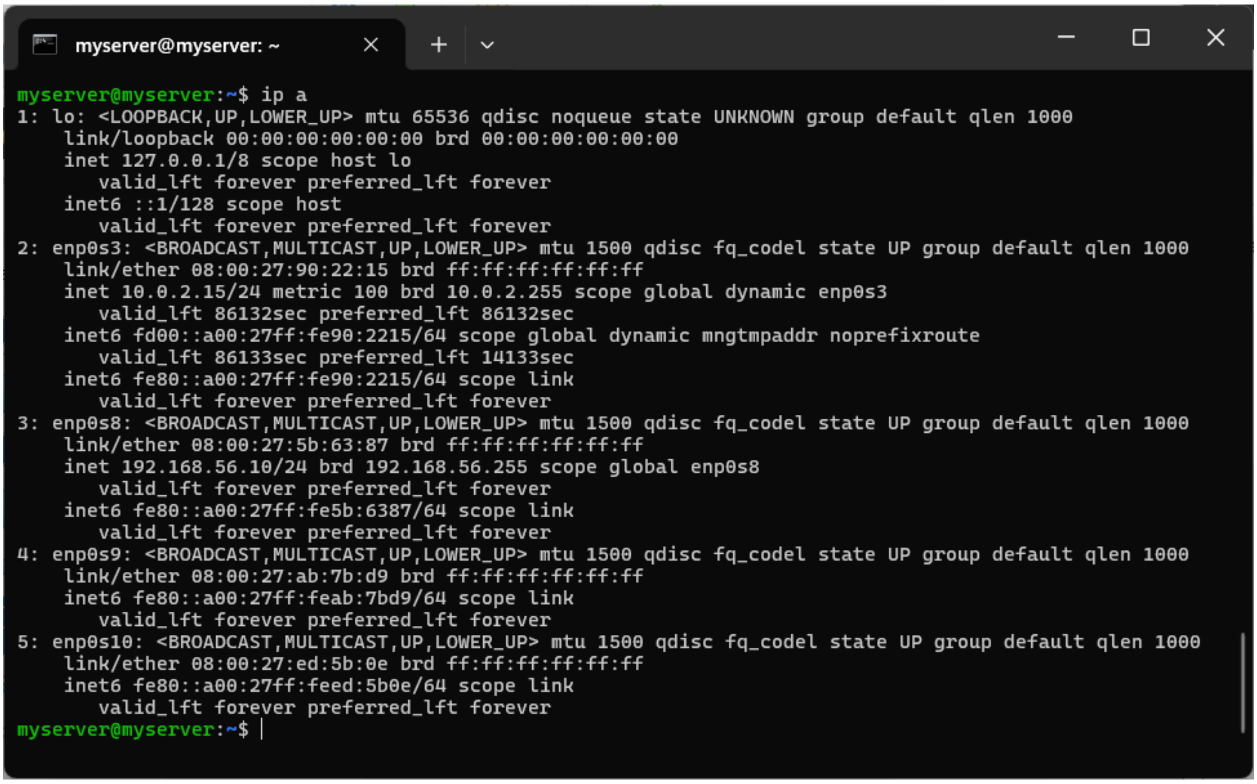
```
cat: /etc/netplan/50-cloud-init.yaml: Permission denied
myserver@myserver:~$ sudo cat /etc/netplan/*.yaml
# This file is generated from information provided by the datasource.  Chan
ges
# to it will not persist across an instance reboot.  To disable cloud-init'
s
# network configuration capabilities, write a file
# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg with the following:
# network: {config: disabled}
network:
    version: 2
    ethernets:
        enp0s3:
            dhcp4: true
        enp0s8:
            dhcp4: false
            addresses: [192.168.56.10/24]
            nameservers:
              addresses: [8.8.8.8, 1.1.1.1]
        enp0s9:
          dhcp4: false
          optional: true
        enp0s10:
          dhcp4: false
          optional: true
myserver@myserver:~$ 
```



```
  GNU nano 6.2                 /etc/kolla/globals.yml

# This should be a VIP, an unused IP on your network that will float betwe>
# the hosts running keepalived for high-availability. If you want to run an
# All-In-One without haproxy and keepalived, you can set enable_haproxy to>
# in "OpenStack options" section, and set this value to the IP of your
# 'network_interface' as set in the Networking section below.
kolla_internal_vip_address: "192.168.56.100"

# This is the DNS name that maps to the kolla_internal_vip_address VIP. By
# default it is the same as kolla_internal_vip_address.
#kolla_internal_fqdn: "{{ kolla_internal_vip_address }}"

# This should be a VIP, an unused IP on your network that will float betwe>
# the hosts running keepalived for high-availability. It defaults to the
# kolla_internal_vip_address, allowing internal and external communication>
# share the same address.  Specify a kolla_external_vip_address to separate
# internal and external requests between two VIPs.
#kolla_external_vip_address: "{{ kolla_internal_vip_address }}"

# The Public address used to communicate with OpenStack as set in the publ>
# for the endpoints that will be created. This DNS name should map to
# kolla_external_vip_address.

^G Help         ^O Write Out    ^W Where Is     ^K Cut          ^T Execute
^X Exit         ^R Read File    ^\ Replace      ^U Paste        ^J Justify
```



```
  GNU nano 6.2                  /etc/kolla/globals.yml
# Additionally, all vxlan/tunnel and storage network traffic will go over >
# interface by default. This interface must contain an IP address.
# It is possible for hosts to have non-matching names of interfaces - thes>
# be set in an inventory file per host or per group or stored separately, >
#     http://docs.ansible.com/ansible/latest/intro_inventory.html
# Yet another way to workaround the naming problem is to create a bond for>
# interface on all hosts and give the bond name here. Similar strategy can>
# followed for other types of interfaces.
network_interface: "enp0s8"

neutron_external_interface: "enp0s10"
# These can be adjusted for even more customization. The default is the sa>
# the 'network_interface'. These interfaces must contain an IP address.
#kolla_external_vip_interface: "{{ network_interface }}"
#api_interface: "{{ network_interface }}"
#tunnel_interface: "{{ network_interface }}"
#dns_interface: "{{ network_interface }}"
#octavia_network_interface: "{{ api_interface }}"

# Configure the address family (AF) per network.
# Valid options are [ ipv4, ipv6 ]
#network_address_family: "ipv4"

^G Help        ^O Write Out   ^W Where Is    ^K Cut         ^T Execute
^X Exit        ^R Read File   ^\ Replace     ^U Paste       ^J Justify
```



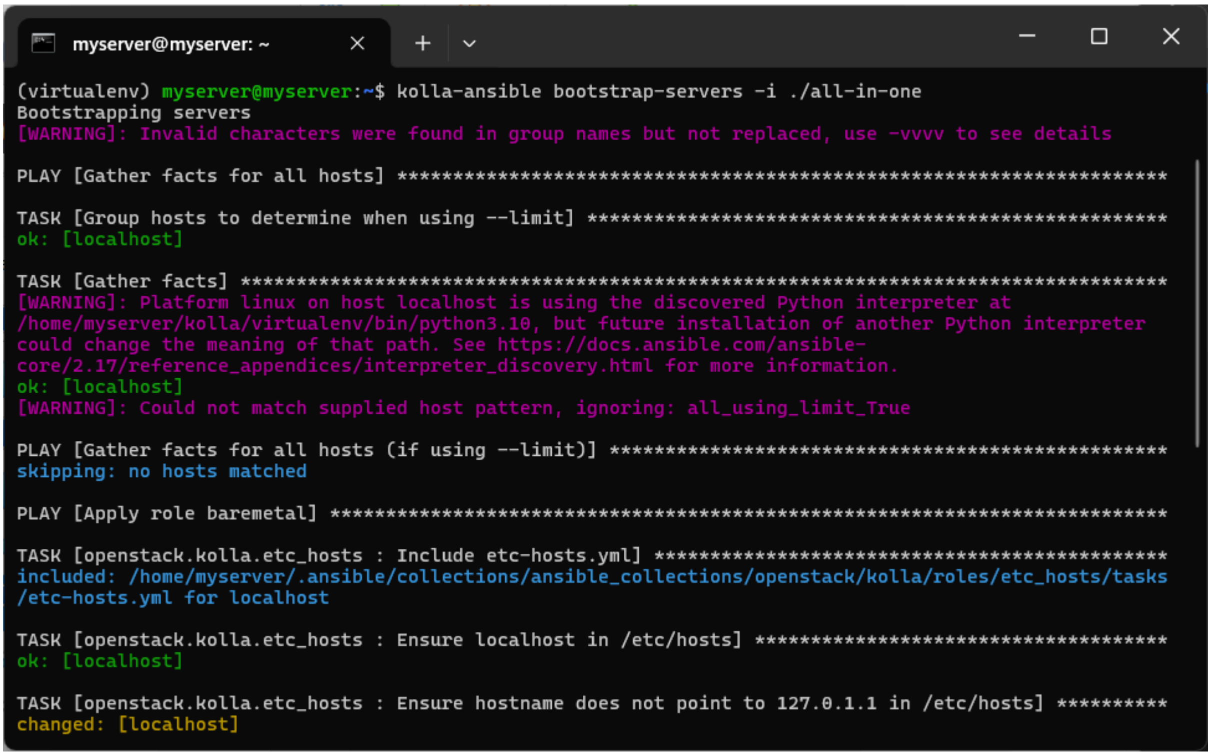
```
myserver@myserver:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:90:22:15 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 metric 100 brd 10.0.2.255 scope global dynamic enp0s3
       valid_lft 86132sec preferred_lft 86132sec
    inet6 fd00::a00:27ff:fe90:2215/64 scope global dynamic mngtmpaddr noprefixroute
       valid_lft 86133sec preferred_lft 14133sec
    inet6 fe80::a00:27ff:fe90:2215/64 scope link
       valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:5b:63:87 brd ff:ff:ff:ff:ff:ff
    inet 192.168.56.10/24 brd 192.168.56.255 scope global enp0s8
       valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe5b:6387/64 scope link
       valid_lft forever preferred_lft forever
4: enp0s9: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:ab:7b:d9 brd ff:ff:ff:ff:ff:ff
    inet6 fe80::a00:27ff:feab:7bd9/64 scope link
       valid_lft forever preferred_lft forever
5: enp0s10: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:ed:5b:0e brd ff:ff:ff:ff:ff:ff
    inet6 fe80::a00:27ff:feed:5b0e/64 scope link
       valid_lft forever preferred_lft forever
myserver@myserver:~$
```

Deployment

After configuration is set, we can proceed to the deployment phase. First we need to setup basic host-level dependencies, like docker. We prepared the server for deployment using this command:

kolla-ansible -i ./all-in-one bootstrap-servers

This step installed Docker and all other necessary packages.



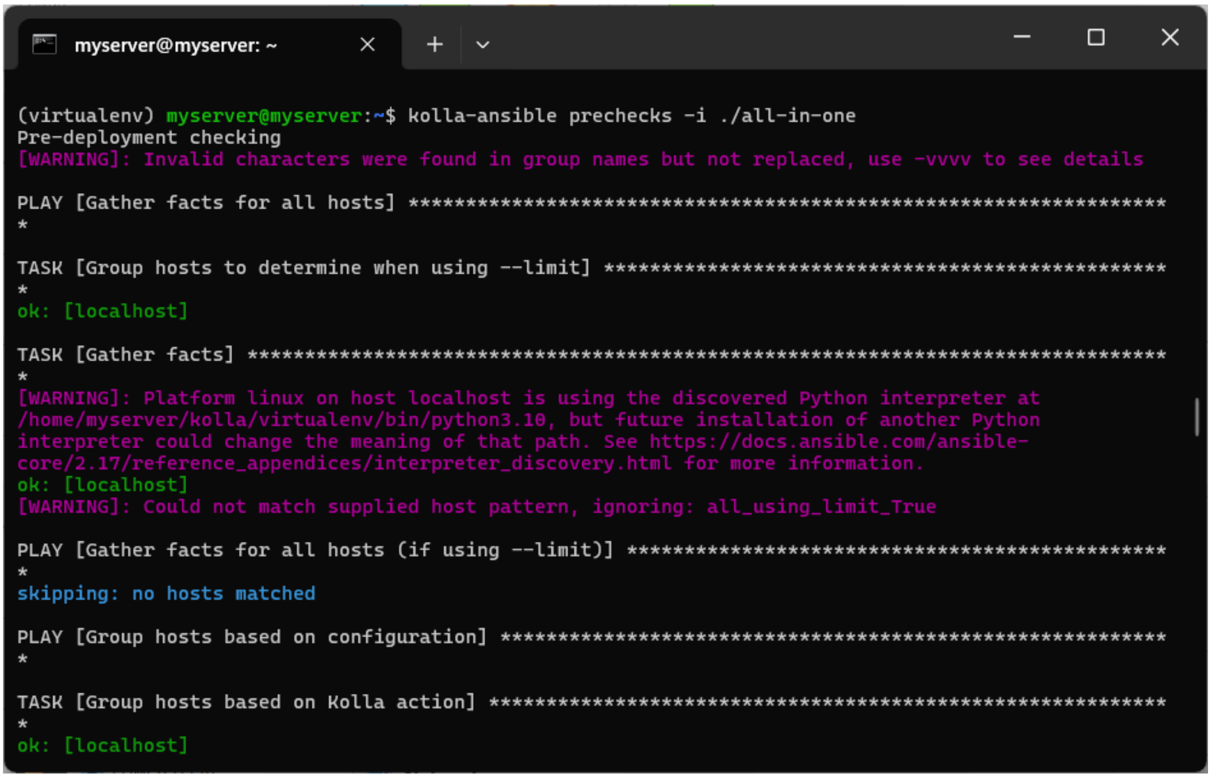
```
(virtualenv) myserver@myserver:~$ kolla-ansible bootstrap-servers -i ./all-in-one
Bootstrapping servers
[WARNING]: Invalid characters were found in group names but not replaced, use -vvvv to see details

PLAY [Gather facts for all hosts] *********************************************************************

TASK [Group hosts to determine when using --limit] ***************************************************
ok: [localhost]

TASK [Gather facts] ***********************************************************************************
[WARNING]: Platform linux on host localhost is using the discovered Python interpreter at
/home/myserver/kolla/virtualenv/bin/python3.10, but future installation of another Python interpreter
could change the meaning of that path. See https://docs.ansible.com/ansible-
core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [localhost]
[WARNING]: Could not match supplied host pattern, ignoring: all_using_limit_True

PLAY [Gather facts for all hosts (if using --limit)] **************************************************
skipping: no hosts matched

PLAY [Apply role baremetal] ***************************************************************************

TASK [openstack.kolla.etc_hosts : Include etc-hosts.yml] **********************************************
included: /home/myserver/.ansible/collections/ansible_collections/openstack/kolla/roles/etc_hosts/tasks
/etc-hosts.yml for localhost

TASK [openstack.kolla.etc_hosts : Ensure localhost in /etc/hosts] *************************************
ok: [localhost]

TASK [openstack.kolla.etc_hosts : Ensure hostname does not point to 127.0.1.1 in /etc/hosts] **********
changed: [localhost]
```

We are doing pre-deployment checks for hosts:

Before starting the deployment, we wanted to make sure everything was ready. We used this command:

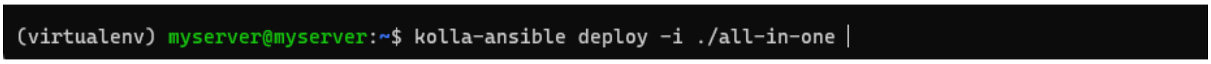
kolla-ansible -i ./all-in-one prechecks

This command checked network settings, disk space, user permissions, and other important things.

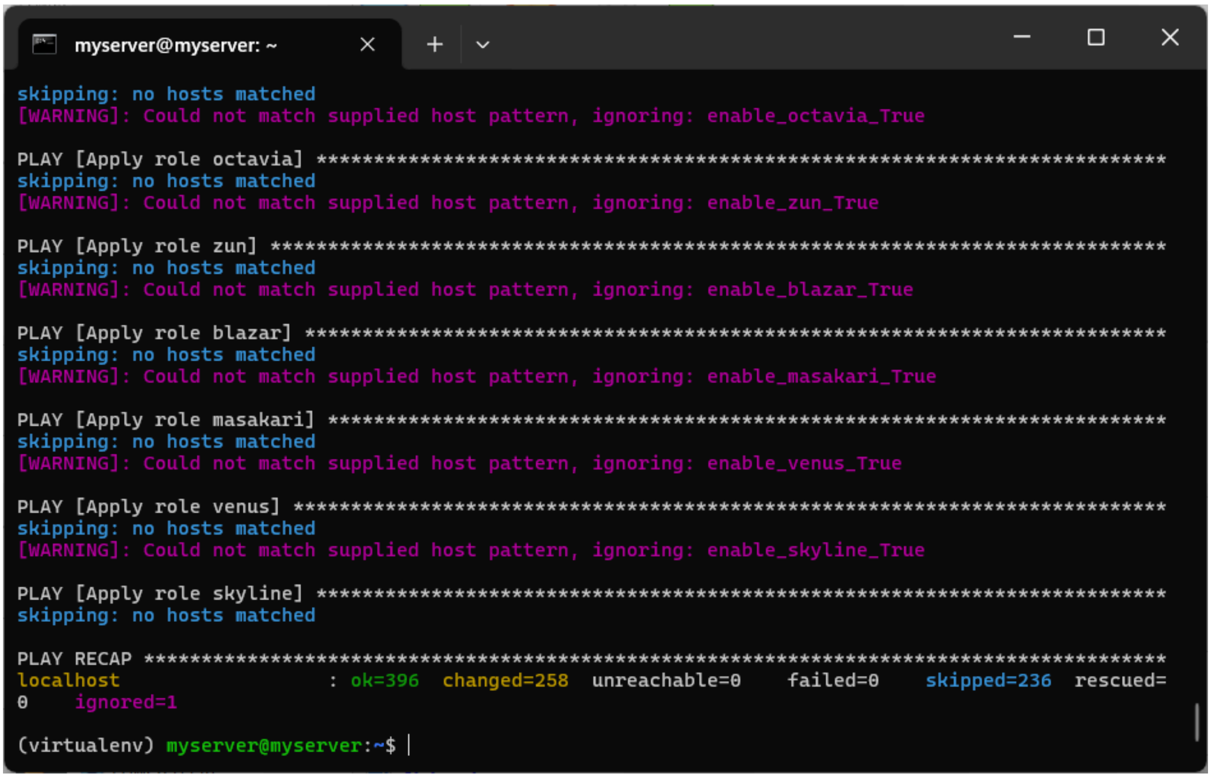


```
(virtualenv) myserver@myserver:~$ kolla-ansible prechecks -i ./all-in-one
Pre-deployment checking
[WARNING]: Invalid characters were found in group names but not replaced, use -vvvv to see details

PLAY [Gather facts for all hosts] *****************************************************************
*

TASK [Group hosts to determine when using --limit] ************************************************
*
ok: [localhost]

TASK [Gather facts] *******************************************************************************
*
[WARNING]: Platform linux on host localhost is using the discovered Python interpreter at
/home/myserver/kolla/virtualenv/bin/python3.10, but future installation of another Python
interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-
core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [localhost]
[WARNING]: Could not match supplied host pattern, ignoring: all_using_limit_True

PLAY [Gather facts for all hosts (if using --limit)] **********************************************
*
skipping: no hosts matched

PLAY [Group hosts based on configuration] *********************************************************
*

TASK [Group hosts based on Kolla action] **********************************************************
*
ok: [localhost]
```

Now we were ready to install OpenStack services. We started the deployment with: *kolla-ansible -i ./all-in-one deploy*
This took some time. It installed all the services such as Keystone, Glance, Nova, Neutron, and more



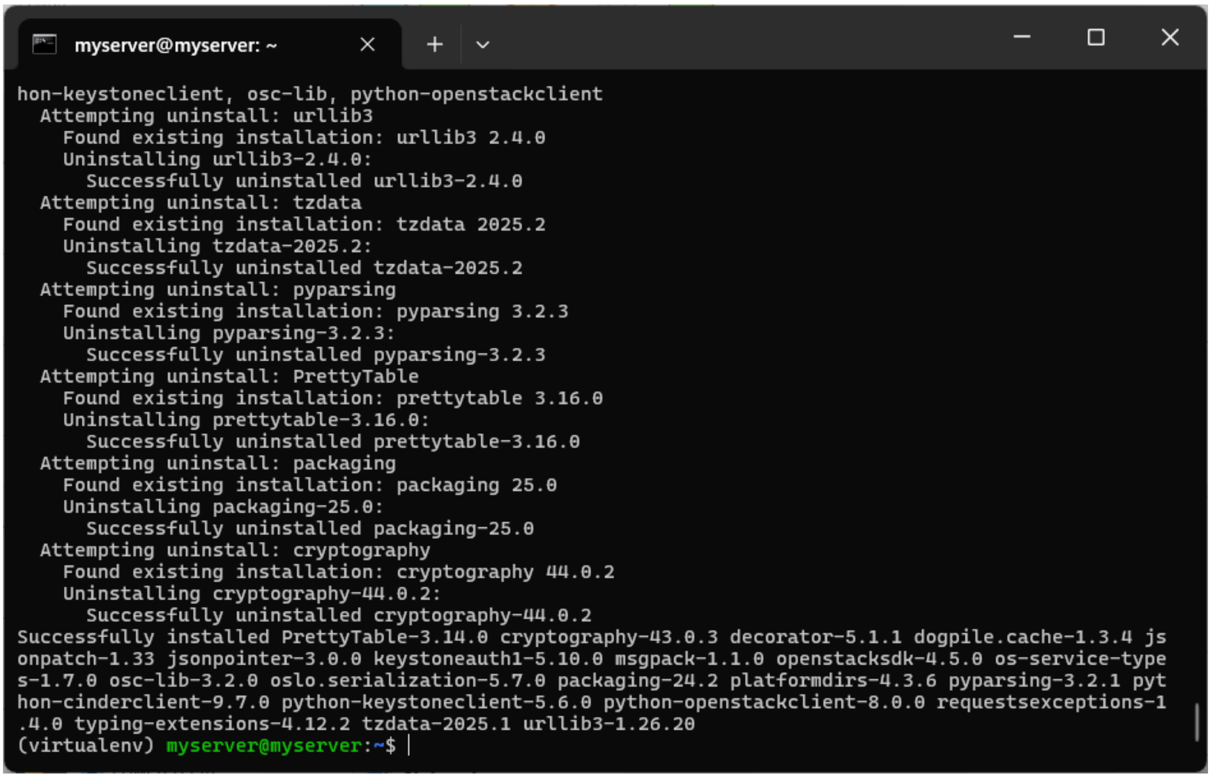
```
(virtualenv) myserver@myserver:~$ kolla-ansible deploy -i ./all-in-one |
```



```
skipping: no hosts matched
[WARNING]: Could not match supplied host pattern, ignoring: enable_octavia_True

PLAY [Apply role octavia] *************************************************************************
skipping: no hosts matched
[WARNING]: Could not match supplied host pattern, ignoring: enable_zun_True

PLAY [Apply role zun] *****************************************************************************
skipping: no hosts matched
[WARNING]: Could not match supplied host pattern, ignoring: enable_blazar_True

PLAY [Apply role blazar] **************************************************************************
skipping: no hosts matched
[WARNING]: Could not match supplied host pattern, ignoring: enable_masakari_True

PLAY [Apply role masakari] ************************************************************************
skipping: no hosts matched
[WARNING]: Could not match supplied host pattern, ignoring: enable_venus_True

PLAY [Apply role venus] ***************************************************************************
skipping: no hosts matched
[WARNING]: Could not match supplied host pattern, ignoring: enable_skyline_True

PLAY [Apply role skyline] *************************************************************************
skipping: no hosts matched

PLAY RECAP ****************************************************************************************
localhost                  : ok=396  changed=258  unreachable=0    failed=0    skipped=236  rescued=
0    ignored=1

(virtualenv) myserver@myserver:~$ |
```

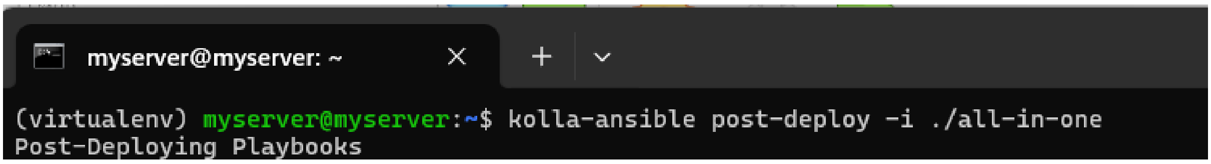
Deployment successful.

Now, To interact with OpenStack from the command line, we installed the OpenStack client (OpenStack CLI):

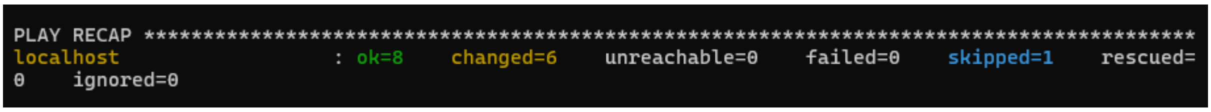


```
hon-keystoneclient, osc-lib, python-openstackclient
  Attempting uninstall: urllib3
    Found existing installation: urllib3 2.4.0
    Uninstalling urllib3-2.4.0:
      Successfully uninstalled urllib3-2.4.0
  Attempting uninstall: tzdata
    Found existing installation: tzdata 2025.2
    Uninstalling tzdata-2025.2:
      Successfully uninstalled tzdata-2025.2
  Attempting uninstall: pyparsing
    Found existing installation: pyparsing 3.2.3
    Uninstalling pyparsing-3.2.3:
      Successfully uninstalled pyparsing-3.2.3
  Attempting uninstall: PrettyTable
    Found existing installation: prettytable 3.16.0
    Uninstalling prettytable-3.16.0:
      Successfully uninstalled prettytable-3.16.0
  Attempting uninstall: packaging
    Found existing installation: packaging 25.0
    Uninstalling packaging-25.0:
      Successfully uninstalled packaging-25.0
  Attempting uninstall: cryptography
    Found existing installation: cryptography 44.0.2
    Uninstalling cryptography-44.0.2:
      Successfully uninstalled cryptography-44.0.2
Successfully installed PrettyTable-3.14.0 cryptography-43.0.3 decorator-5.1.1 dogpile.cache-1.3.4 js
onpatch-1.33 jsonpointer-3.0.0 keystoneauth1-5.10.0 msgpack-1.1.0 openstacksdk-4.5.0 os-service-type
s-1.7.0 osc-lib-3.2.0 oslo.serialization-5.7.0 packaging-24.2 platformdirs-4.3.6 pyparsing-3.2.1 pyt
hon-cinderclient-9.7.0 python-keystoneclient-5.6.0 python-openstackclient-8.0.0 requestsexceptions-1
.4.0 typing-extensions-4.12.2 tzdata-2025.1 urllib3-1.26.20
(virtualenv) myserver@myserver:~$ |
```

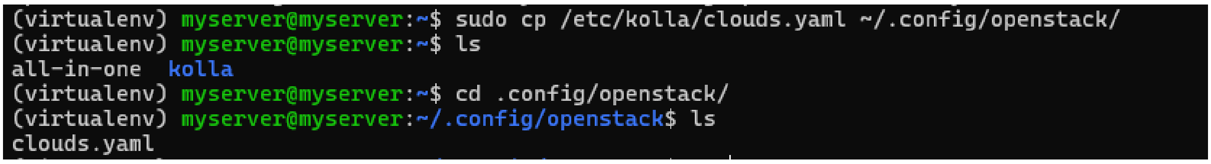
OpenStack requires a **clouds.yaml file** where credentials for the admin user are set. We ran kolla-ansible post-deploy to finish the setup. This created the admin-openrc.sh file with all the OpenStack environment variables. We also got the clouds.yaml file for using the OpenStack CLI. We copied the config to the right place with:
mkdir -p ~/.config/openstack && cp /etc/kolla/clouds.yaml ~/.config/openstack/



```
(virtualenv) myserver@myserver:~$ kolla-ansible post-deploy -i ./all-in-one
Post-Deploying Playbooks
```

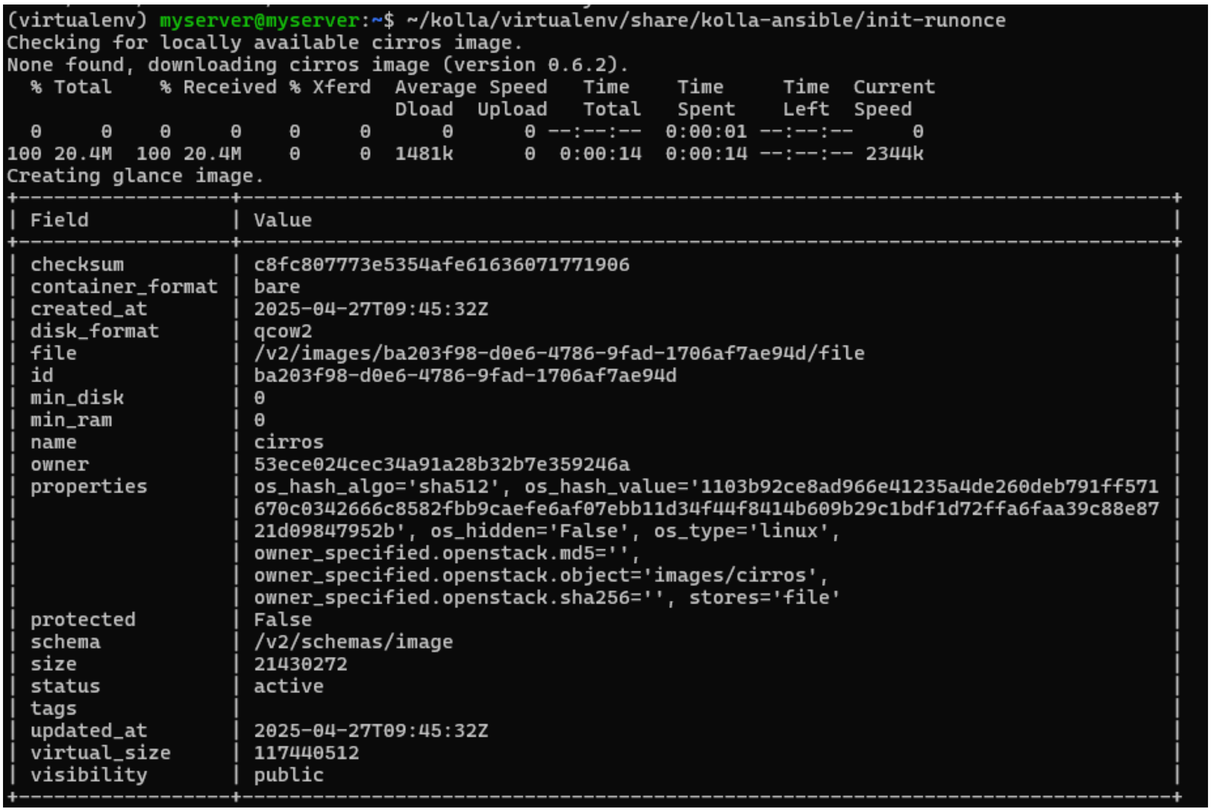


```
PLAY RECAP *****************************************************************************************
localhost                  : ok=8    changed=6    unreachable=0    failed=0    skipped=1    rescued=
0    ignored=0
```

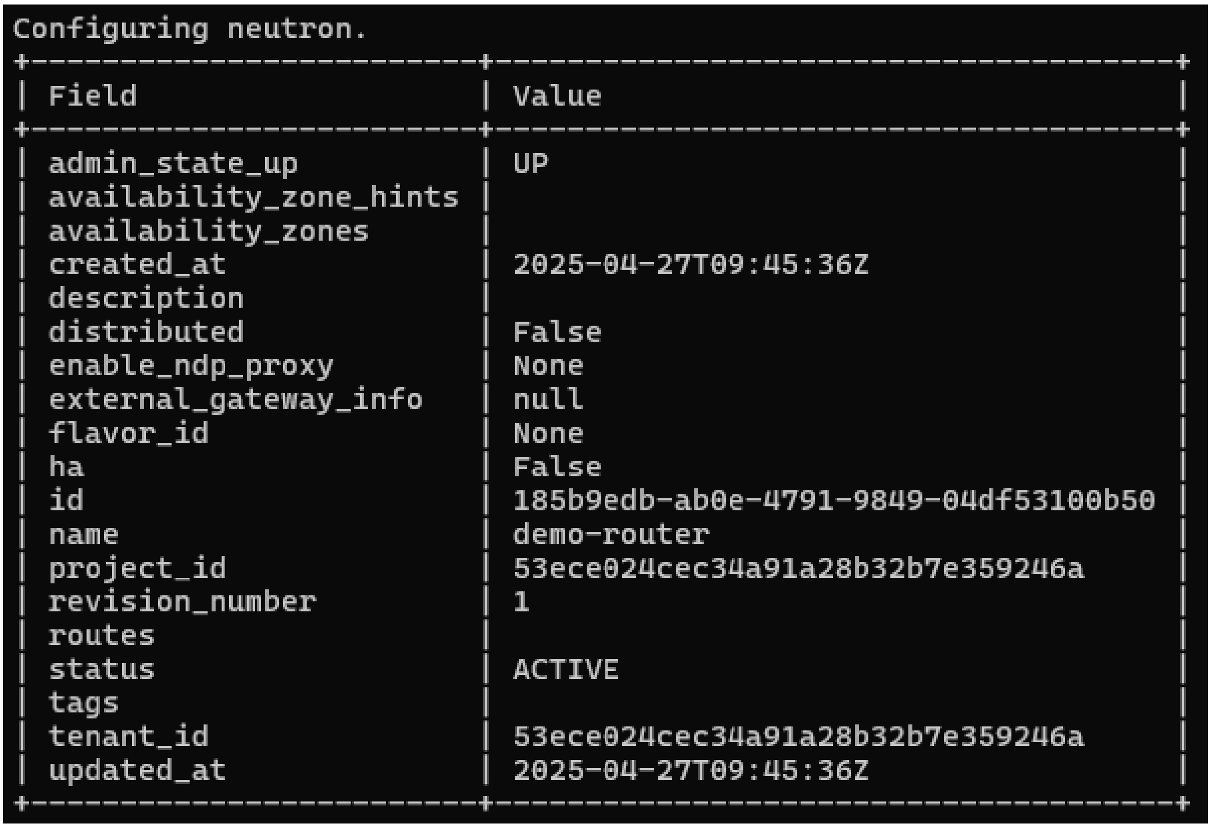


```
(virtualenv) myserver@myserver:~$ sudo cp /etc/kolla/clouds.yaml ~/.config/openstack/
(virtualenv) myserver@myserver:~$ ls
all-in-one  kolla
(virtualenv) myserver@myserver:~$ cd .config/openstack/
(virtualenv) myserver@myserver:~/.config/openstack$ ls
clouds.yaml
```

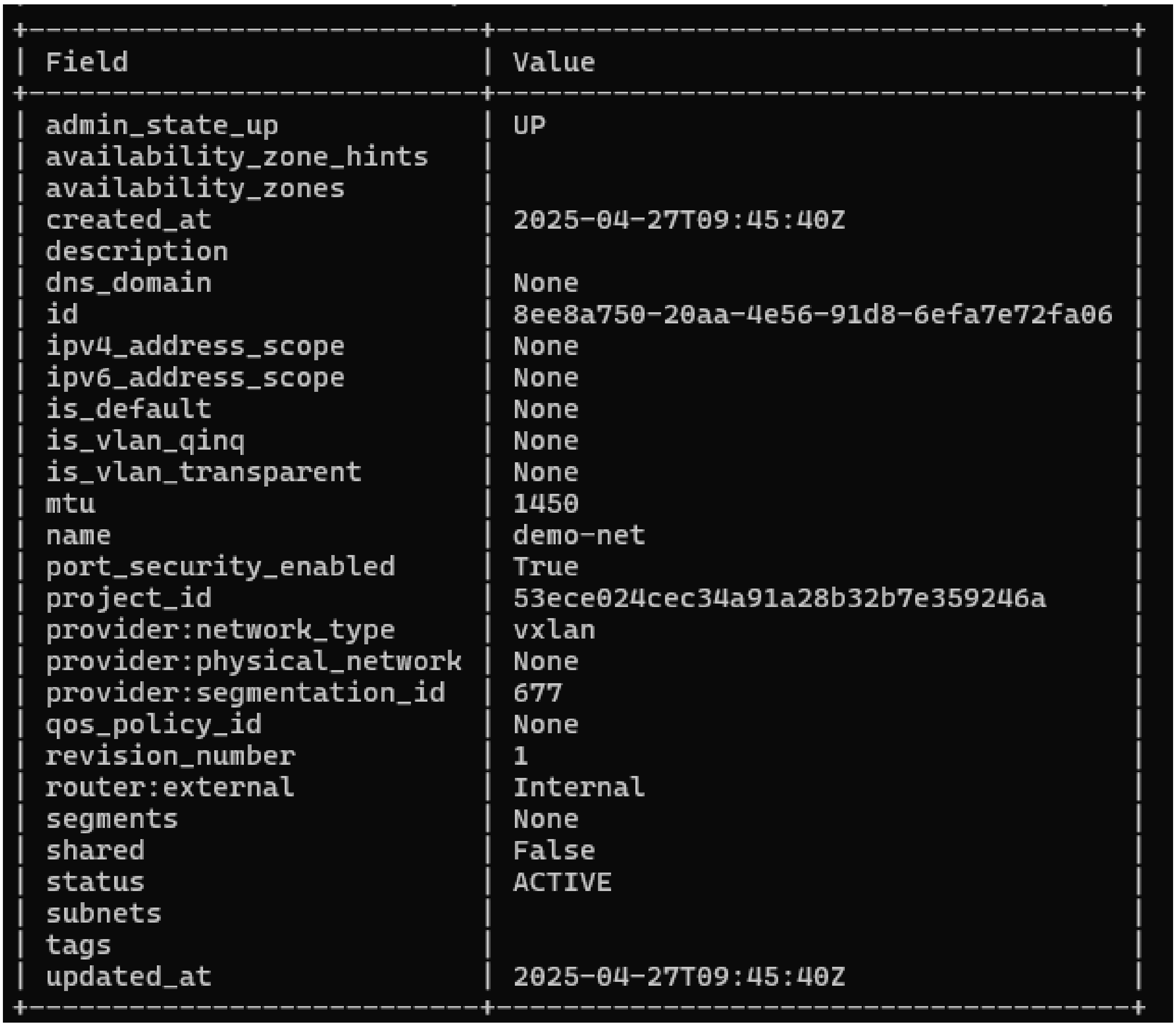
We ran this command to create a demo project, network, subnet, router, image, and a basic flavor:*kolla/virtualenv/share/kolla-ansible/init-runonce*
This helped us quickly get a working setup to test instances
Running **init script** to create demo resources:



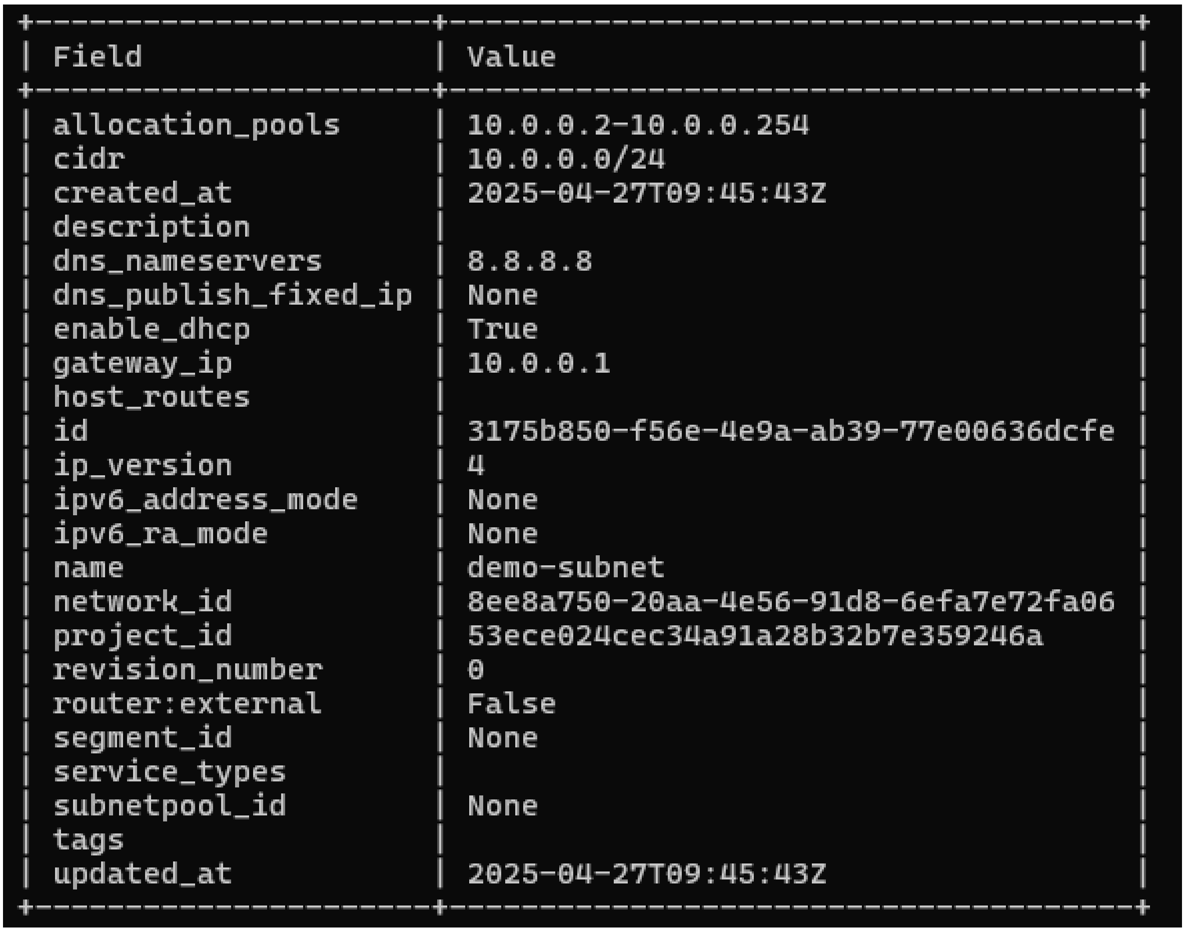
```
(virtualenv) myserver@myserver:~$ ~/kolla/virtualenv/share/kolla-ansible/init-runonce
Checking for locally available cirros image.
None found, downloading cirros image (version 0.6.2).
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
  0     0    0     0    0     0      0      0 --:--:--  0:00:01 --:--:--     0
100 20.4M  100 20.4M    0     0  1481k      0  0:00:14  0:00:14 --:--:-- 2344k
Creating glance image.
+------------------+-------------------------------------------------------------------------------+
| Field            | Value                                                                         |
+------------------+-------------------------------------------------------------------------------+
| checksum         | c8fc807773e5354afe61636071771906                                              |
| container_format | bare                                                                          |
| created_at       | 2025-04-27T09:45:32Z                                                          |
| disk_format      | qcow2                                                                         |
| file             | /v2/images/ba203f98-d0e6-4786-9fad-1706af7ae94d/file                          |
| id               | ba203f98-d0e6-4786-9fad-1706af7ae94d                                          |
| min_disk         | 0                                                                             |
| min_ram          | 0                                                                             |
| name             | cirros                                                                        |
| owner            | 53ece024cec34a91a28b32b7e359246a                                              |
| properties       | os_hash_algo='sha512', os_hash_value='1103b92ce8ad966e41235a4de260deb791ff571 |
|                  | 670c0342666c8582fbb9caefe6af07ebb11d34f44f8414b609b29c1bdf1d72ffa6faa39c88e87 |
|                  | 21d09847952b', os_hidden='False', os_type='linux',                            |
|                  | owner_specified.openstack.md5='',                                             |
|                  | owner_specified.openstack.object='images/cirros',                             |
|                  | owner_specified.openstack.sha256='', stores='file'                            |
| protected        | False                                                                         |
| schema           | /v2/schemas/image                                                             |
| size             | 21430272                                                                      |
| status           | active                                                                        |
| tags             |                                                                               |
| updated_at       | 2025-04-27T09:45:32Z                                                          |
| virtual_size     | 117440512                                                                     |
| visibility       | public                                                                        |
+------------------+-------------------------------------------------------------------------------+
```



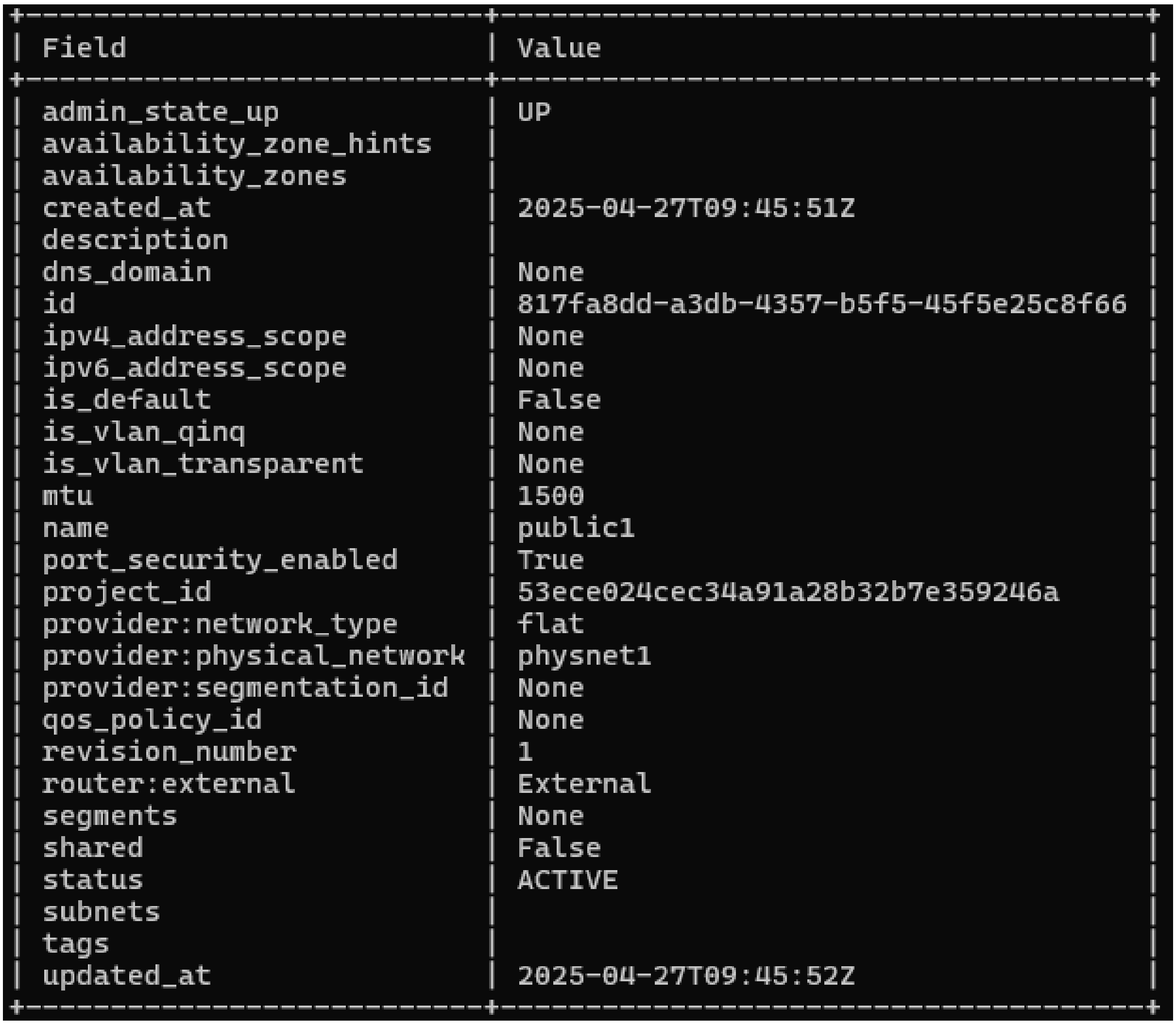
```
Configuring neutron.
+-------------------------+--------------------------------------+
| Field                   | Value                                |
+-------------------------+--------------------------------------+
| admin_state_up          | UP                                   |
| availability_zone_hints |                                      |
| availability_zones      |                                      |
| created_at              | 2025-04-27T09:45:36Z                 |
| description             |                                      |
| distributed             | False                                |
| enable_ndp_proxy        | None                                 |
| external_gateway_info   | null                                 |
| flavor_id               | None                                 |
| ha                      | False                                |
| id                      | 185b9edb-ab0e-4791-9849-04df53100b50 |
| name                    | demo-router                          |
| project_id              | 53ece024cec34a91a28b32b7e359246a     |
| revision_number         | 1                                    |
| routes                  |                                      |
| status                  | ACTIVE                               |
| tags                    |                                      |
| tenant_id               | 53ece024cec34a91a28b32b7e359246a     |
| updated_at              | 2025-04-27T09:45:36Z                 |
+-------------------------+--------------------------------------+
```



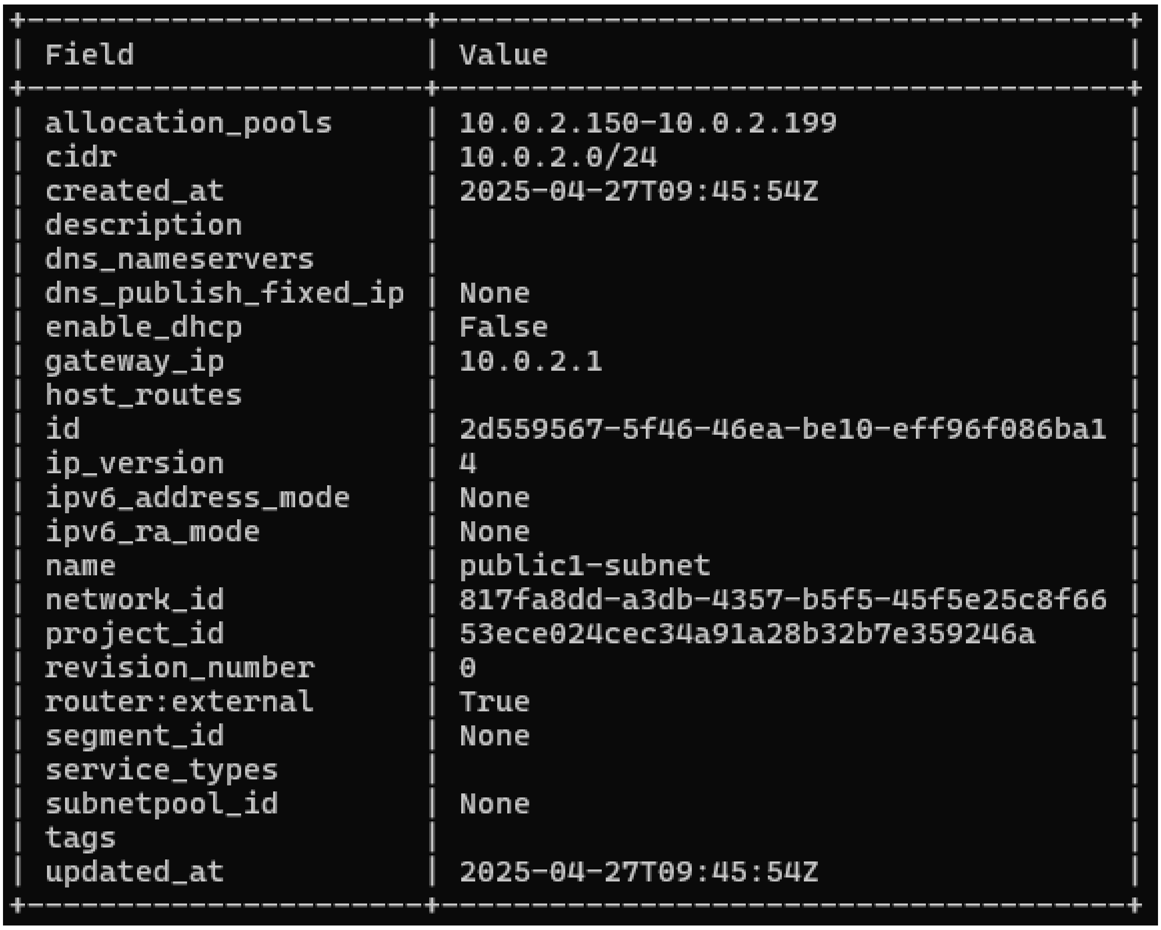
```
+---------------------------+--------------------------------------+
| Field                     | Value                                |
+---------------------------+--------------------------------------+
| admin_state_up            | UP                                   |
| availability_zone_hints   |                                      |
| availability_zones        |                                      |
| created_at                | 2025-04-27T09:45:40Z                 |
| description               |                                      |
| dns_domain                | None                                 |
| id                        | 8ee8a750-20aa-4e56-91d8-6efa7e72fa06 |
| ipv4_address_scope        | None                                 |
| ipv6_address_scope        | None                                 |
| is_default                | None                                 |
| is_vlan_qinq              | None                                 |
| is_vlan_transparent       | None                                 |
| mtu                       | 1450                                 |
| name                      | demo-net                             |
| port_security_enabled     | True                                 |
| project_id                | 53ece024cec34a91a28b32b7e359246a     |
| provider:network_type     | vxlan                                |
| provider:physical_network | None                                 |
| provider:segmentation_id  | 677                                  |
| qos_policy_id             | None                                 |
| revision_number           | 1                                    |
| router:external           | Internal                             |
| segments                  | None                                 |
| shared                    | False                                |
| status                    | ACTIVE                               |
| subnets                   |                                      |
| tags                      |                                      |
| updated_at                | 2025-04-27T09:45:40Z                 |
+---------------------------+--------------------------------------+
```



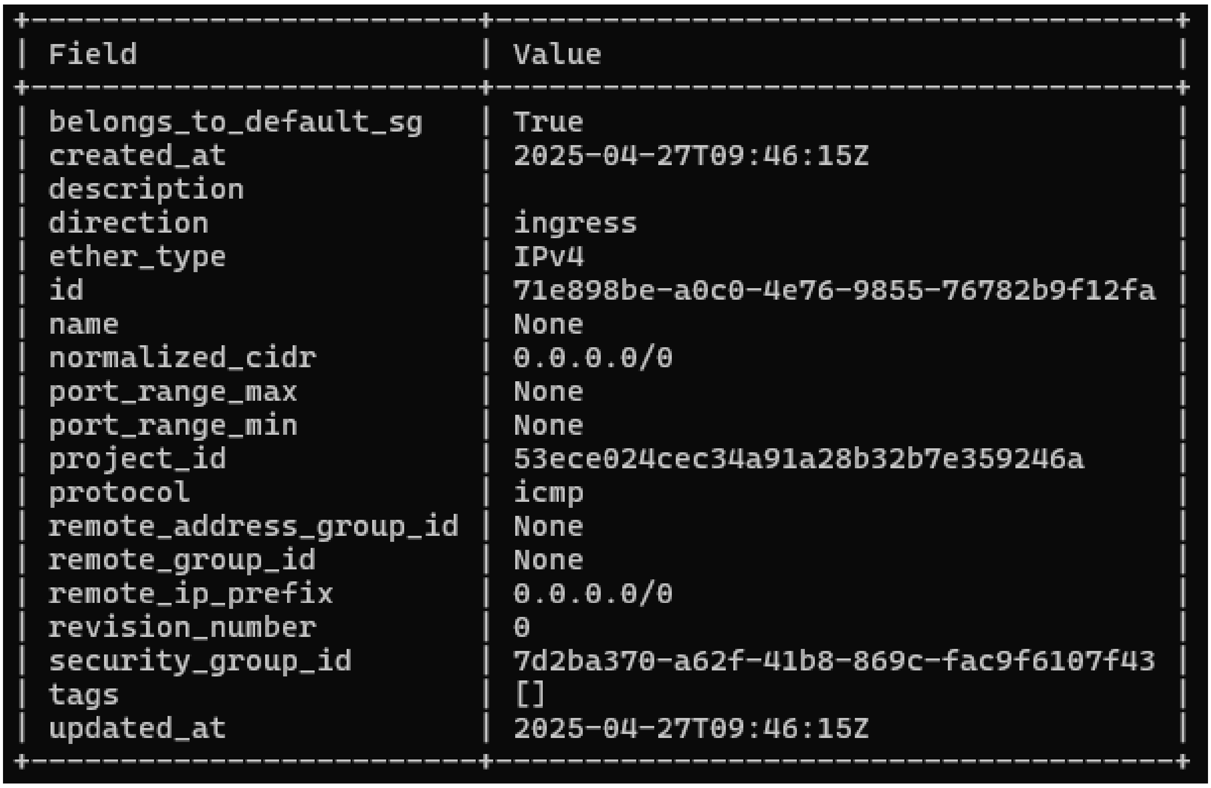
```
+----------------------+--------------------------------------+
| Field                | Value                                |
+----------------------+--------------------------------------+
| allocation_pools     | 10.0.0.2-10.0.0.254                  |
| cidr                 | 10.0.0.0/24                          |
| created_at           | 2025-04-27T09:45:43Z                 |
| description          |                                      |
| dns_nameservers      | 8.8.8.8                              |
| dns_publish_fixed_ip | None                                 |
| enable_dhcp          | True                                 |
| gateway_ip           | 10.0.0.1                             |
| host_routes          |                                      |
| id                   | 3175b850-f56e-4e9a-ab39-77e00636dcfe |
| ip_version           | 4                                    |
| ipv6_address_mode    | None                                 |
| ipv6_ra_mode         | None                                 |
| name                 | demo-subnet                          |
| network_id           | 8ee8a750-20aa-4e56-91d8-6efa7e72fa06 |
| project_id           | 53ece024cec34a91a28b32b7e359246a     |
| revision_number      | 0                                    |
| router:external      | False                                |
| segment_id           | None                                 |
| service_types        |                                      |
| subnetpool_id        | None                                 |
| tags                 |                                      |
| updated_at           | 2025-04-27T09:45:43Z                 |
+----------------------+--------------------------------------+
```



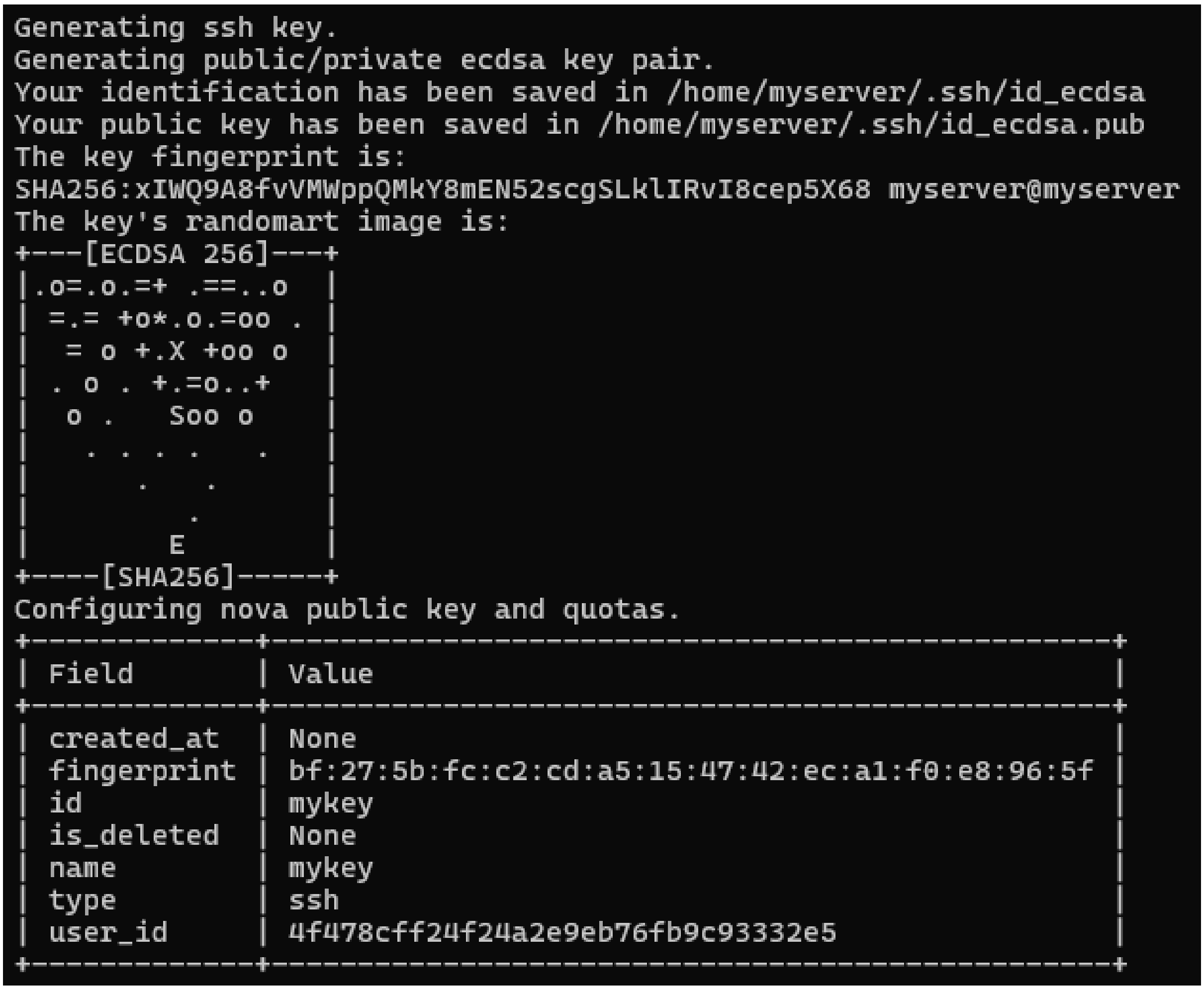
```
+---------------------------+--------------------------------------+
| Field                     | Value                                |
+---------------------------+--------------------------------------+
| admin_state_up            | UP                                   |
| availability_zone_hints   |                                      |
| availability_zones        |                                      |
| created_at                | 2025-04-27T09:45:51Z                 |
| description               |                                      |
| dns_domain                | None                                 |
| id                        | 817fa8dd-a3db-4357-b5f5-45f5e25c8f66 |
| ipv4_address_scope        | None                                 |
| ipv6_address_scope        | None                                 |
| is_default                | False                                |
| is_vlan_qinq              | None                                 |
| is_vlan_transparent       | None                                 |
| mtu                       | 1500                                 |
| name                      | public1                              |
| port_security_enabled     | True                                 |
| project_id                | 53ece024cec34a91a28b32b7e359246a     |
| provider:network_type     | flat                                 |
| provider:physical_network | physnet1                             |
| provider:segmentation_id  | None                                 |
| qos_policy_id             | None                                 |
| revision_number           | 1                                    |
| router:external           | External                             |
| segments                  | None                                 |
| shared                    | False                                |
| status                    | ACTIVE                               |
| subnets                   |                                      |
| tags                      |                                      |
| updated_at                | 2025-04-27T09:45:52Z                 |
+---------------------------+--------------------------------------+
```



```
+----------------------+--------------------------------------+
| Field                | Value                                |
+----------------------+--------------------------------------+
| allocation_pools     | 10.0.2.150-10.0.2.199                |
| cidr                 | 10.0.2.0/24                          |
| created_at           | 2025-04-27T09:45:54Z                 |
| description          |                                      |
| dns_nameservers      |                                      |
| dns_publish_fixed_ip | None                                 |
| enable_dhcp          | False                                |
| gateway_ip           | 10.0.2.1                             |
| host_routes          |                                      |
| id                   | 2d559567-5f46-46ea-be10-eff96f086ba1 |
| ip_version           | 4                                    |
| ipv6_address_mode    | None                                 |
| ipv6_ra_mode         | None                                 |
| name                 | public1-subnet                       |
| network_id           | 817fa8dd-a3db-4357-b5f5-45f5e25c8f66 |
| project_id           | 53ece024cec34a91a28b32b7e359246a     |
| revision_number      | 0                                    |
| router:external      | True                                 |
| segment_id           | None                                 |
| service_types        |                                      |
| subnetpool_id        | None                                 |
| tags                 |                                      |
| updated_at           | 2025-04-27T09:45:54Z                 |
+----------------------+--------------------------------------+
```



```
+-------------------------+--------------------------------------+
| Field                   | Value                                |
+-------------------------+--------------------------------------+
| belongs_to_default_sg   | True                                 |
| created_at              | 2025-04-27T09:46:15Z                 |
| description             |                                      |
| direction               | ingress                              |
| ether_type              | IPv4                                 |
| id                      | 71e898be-a0c0-4e76-9855-76782b9f12fa |
| name                    | None                                 |
| normalized_cidr         | 0.0.0.0/0                            |
| port_range_max          | None                                 |
| port_range_min          | None                                 |
| project_id              | 53ece024cec34a91a28b32b7e359246a     |
| protocol                | icmp                                 |
| remote_address_group_id | None                                 |
| remote_group_id         | None                                 |
| remote_ip_prefix        | 0.0.0.0/0                            |
| revision_number         | 0                                    |
| security_group_id       | 7d2ba370-a62f-41b8-869c-fac9f6107f43 |
| tags                    | []                                   |
| updated_at              | 2025-04-27T09:46:15Z                 |
+-------------------------+--------------------------------------+
```

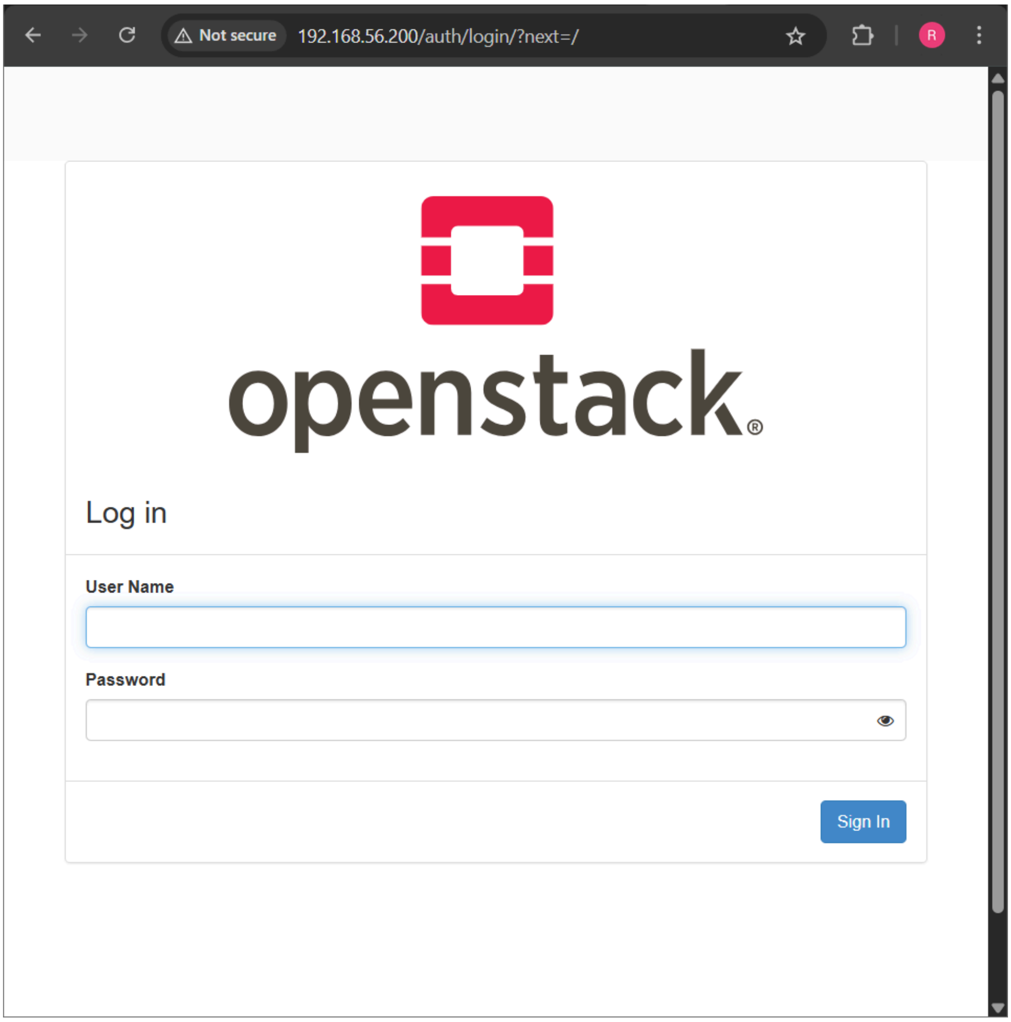


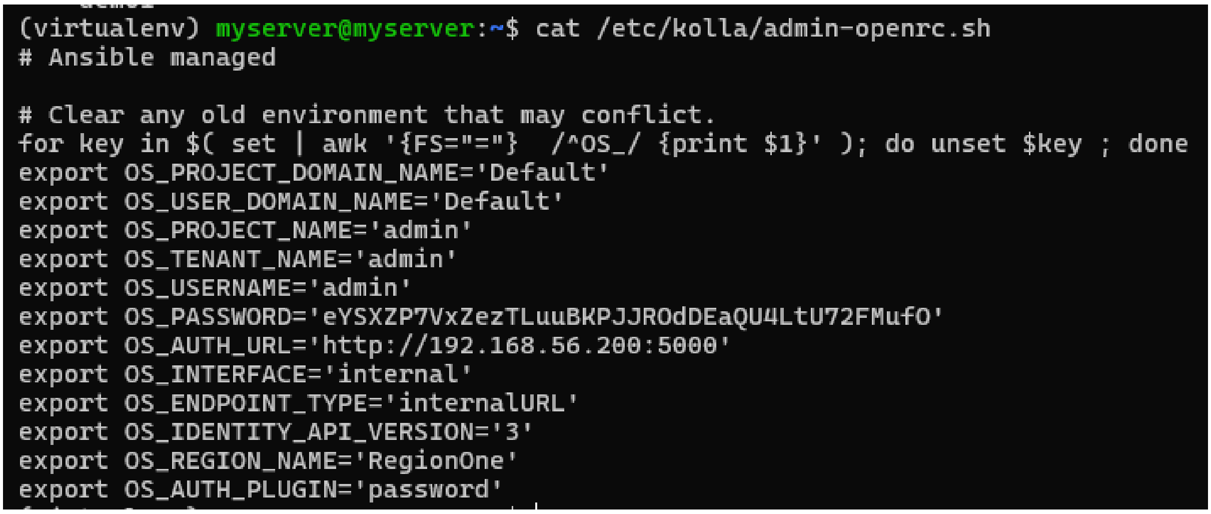
```
Generating ssh key.
Generating public/private ecdsa key pair.
Your identification has been saved in /home/myserver/.ssh/id_ecdsa
Your public key has been saved in /home/myserver/.ssh/id_ecdsa.pub
The key fingerprint is:
SHA256:xIWQ9A8fvVMWppQMkY8mEN52scgSLklIRvI8cep5X68 myserver@myserver
The key's randomart image is:
+---[ECDSA 256]---+
|.o=.o.=+ .==..o  |
| =.= +o*.o.=oo . |
|  = o +.X +oo o  |
| . o . +.=o..+   |
|  o .   Soo o    |
|   . . . .   .   |
|      .   .      |
|         .       |
|        E        |
+----[SHA256]-----+
Configuring nova public key and quotas.
+-------------+-------------------------------------------------+
| Field       | Value                                           |
+-------------+-------------------------------------------------+
| created_at  | None                                            |
| fingerprint | bf:27:5b:fc:c2:cd:a5:15:47:42:ec:a1:f0:e8:96:5f |
| id          | mykey                                           |
| is_deleted  | None                                            |
| name        | mykey                                           |
| type        | ssh                                             |
| user_id     | 4f478cff24f24a2e9eb76fb9c93332e5                |
+-------------+-------------------------------------------------+
```

Openstack administration

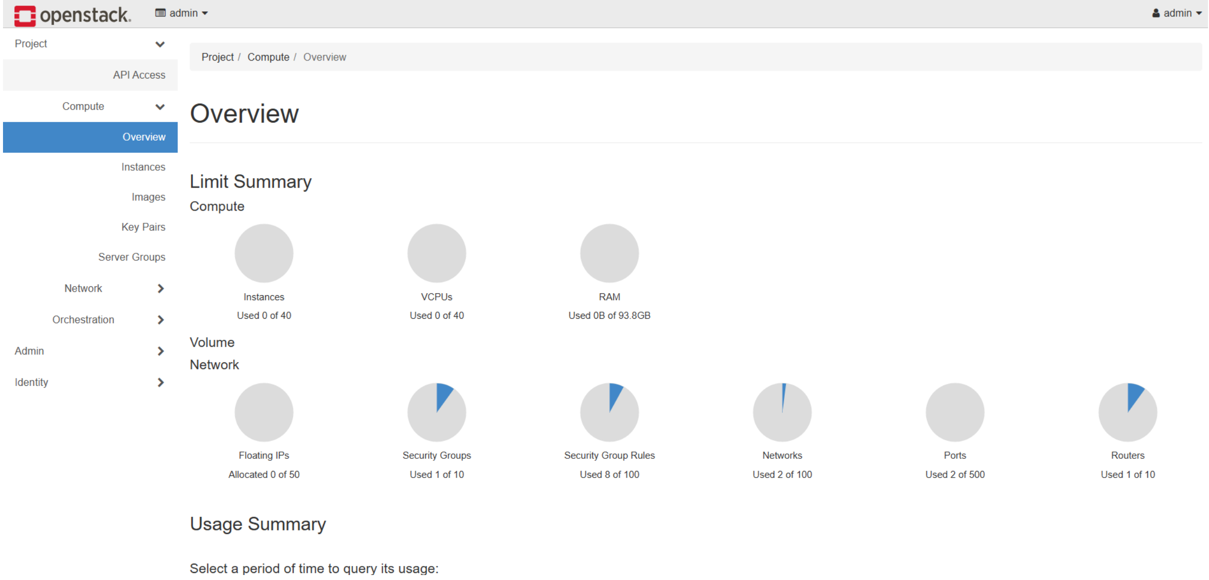
Access to Dashboard

We opened a browser and went to *http://192.168.56.200* which is the IP of our VM. The Horizon web interface loaded, and we logged in using the admin credentials from the *admin-openrc.sh*





```
(virtualenv) myserver@myserver:~$ cat /etc/kolla/admin-openrc.sh
# Ansible managed

# Clear any old environment that may conflict.
for key in $( set | awk '{FS="="}  /^OS_/ {print $1}' ); do unset $key ; done
export OS_PROJECT_DOMAIN_NAME='Default'
export OS_USER_DOMAIN_NAME='Default'
export OS_PROJECT_NAME='admin'
export OS_TENANT_NAME='admin'
export OS_USERNAME='admin'
export OS_PASSWORD='eYSXZP7VxZezTLuuBKPJJROdDEaQU4LtU72FMufO'
export OS_AUTH_URL='http://192.168.56.200:5000'
export OS_INTERFACE='internal'
export OS_ENDPOINT_TYPE='internalURL'
export OS_IDENTITY_API_VERSION='3'
export OS_REGION_NAME='RegionOne'
export OS_AUTH_PLUGIN='password'
```



Overview

Limit Summary

Compute

Instances
Used 0 of 40

VCPUs
Used 0 of 40

RAM
Used 0B of 93.8GB

Volume

Network

Floating IPs
Allocated 0 of 50

Security Groups
Used 1 of 10

Security Group Rules
Used 8 of 100

Networks
Used 2 of 100

Ports
Used 2 of 500

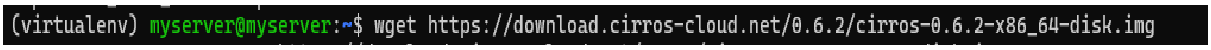
Routers
Used 1 of 10

Usage Summary

Select a period of time to query its usage:

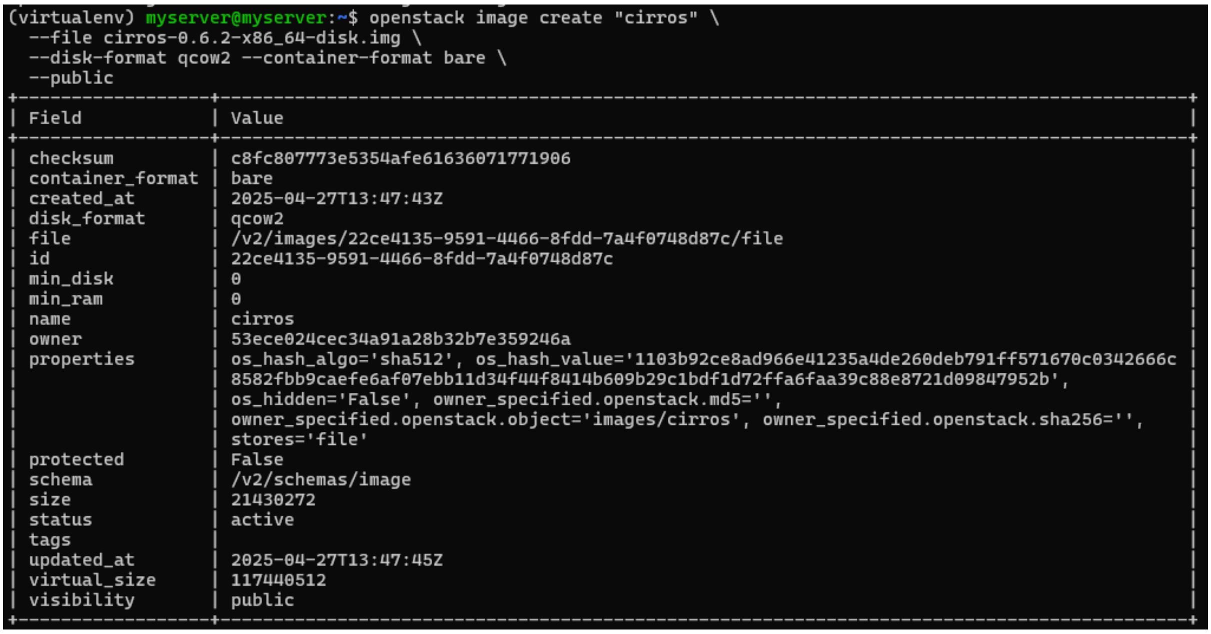
Download the Cirros Image

We downloaded a test Cirros image *called cirros-0.6.2-x86_64-disk.img* and uploaded it. This made the image available to launch new virtual machines.



```
(virtualenv) myserver@myserver:~$ wget https://download.cirros-cloud.net/0.6.2/cirros-0.6.2-x86_64-disk.img
```

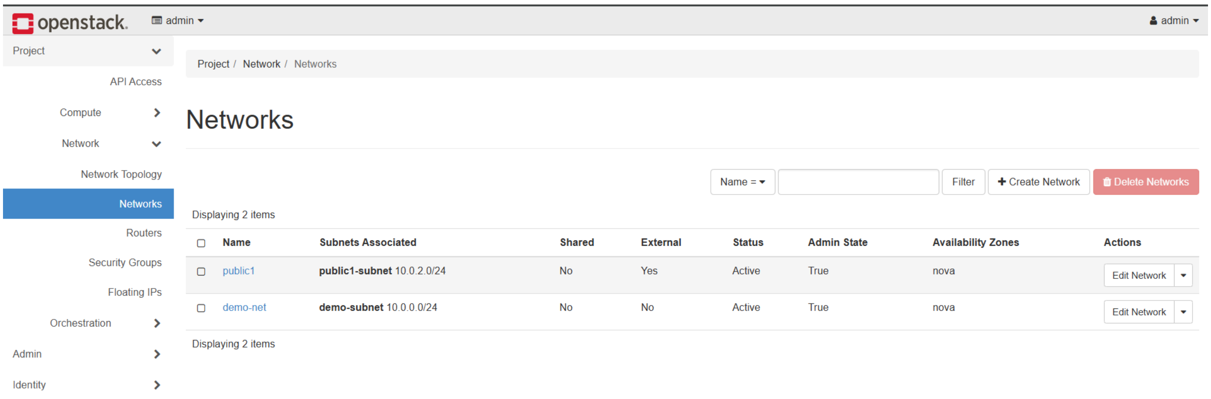
The image Cirros via openstack CLI, is created.



```
(virtualenv) myserver@myserver:~$ openstack image create "cirros" \
  --file cirros-0.6.2-x86_64-disk.img \
  --disk-format qcow2 --container-format bare \
  --public
+------------------+------------------------------------------------------------------------------------------+
| Field            | Value                                                                                    |
+------------------+------------------------------------------------------------------------------------------+
| checksum         | c8fc807773e5354afe61636071771906                                                         |
| container_format | bare                                                                                     |
| created_at       | 2025-04-27T13:47:43Z                                                                     |
| disk_format      | qcow2                                                                                    |
| file             | /v2/images/22ce4135-9591-4466-8fdd-7a4f0748d87c/file                                     |
| id               | 22ce4135-9591-4466-8fdd-7a4f0748d87c                                                     |
| min_disk         | 0                                                                                        |
| min_ram          | 0                                                                                        |
| name             | cirros                                                                                   |
| owner            | 53ece024cec34a91a28b32b7e359246a                                                         |
| properties       | os_hash_algo='sha512', os_hash_value='1103b92ce8ad966e41235a4de260deb791ff571670c0342666c |
|                  | 8582fbb9caefe6af07ebb11d34f44f8414b609b29c1bdf1d72ffa6faa39c88e8721d09847952b',          |
|                  | os_hidden='False', owner_specified.openstack.md5='',                                     |
|                  | owner_specified.openstack.object='images/cirros', owner_specified.openstack.sha256='',   |
|                  | stores='file'                                                                            |
| protected        | False                                                                                    |
| schema           | /v2/schemas/image                                                                        |
| size             | 21430272                                                                                 |
| status           | active                                                                                   |
| tags             |                                                                                          |
| updated_at       | 2025-04-27T13:47:45Z                                                                     |
| virtual_size     | 117440512                                                                                |
| visibility       | public                                                                                   |
+------------------+------------------------------------------------------------------------------------------+
```

We created two networks:

- ➢ An internal private network called **demo-net** with IP range 10.0.0.0/24
- ➢ An external network called **public1** with IP range 10.0.2.0/24 (the one we will connect our router to).



Project / Network / Networks

Networks

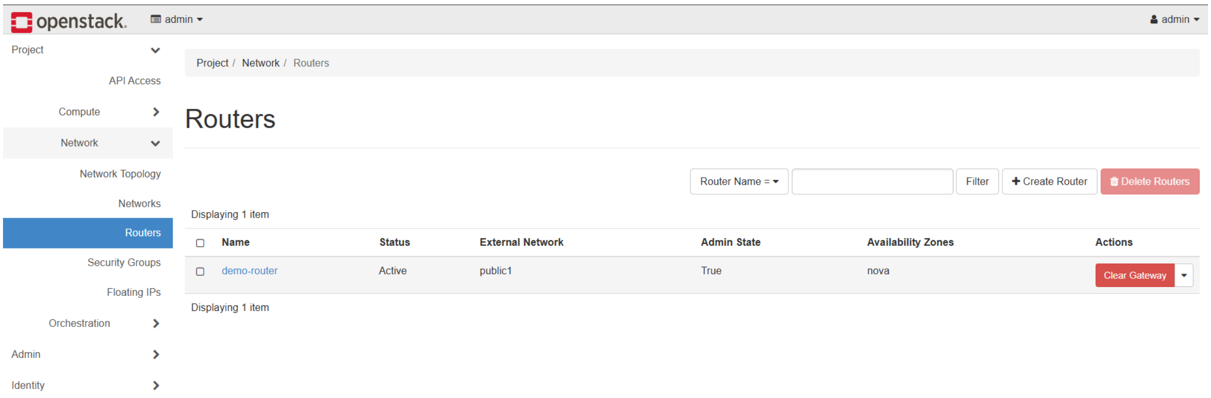
Displaying 2 items

□	Name	Subnets Associated	Shared	External	Status	Admin State	Availability Zones	Actions
□	public1	**public1-subnet** 10.0.2.0/24	No	Yes	Active	True	nova	Edit Network
□	demo-net	**demo-subnet** 10.0.0.0/24	No	No	Active	True	nova	Edit Network

Displaying 2 items

We created a router named router1, set its external gateway to public1, and added an interface connecting it to demo-net. This allowed traffic between the private network and external network. The router is connected to the External Network (`public1`, 10.0.2.0/24).

And it's also connected to the Internal Network (`demo-net`, 10.0.0.0/24).



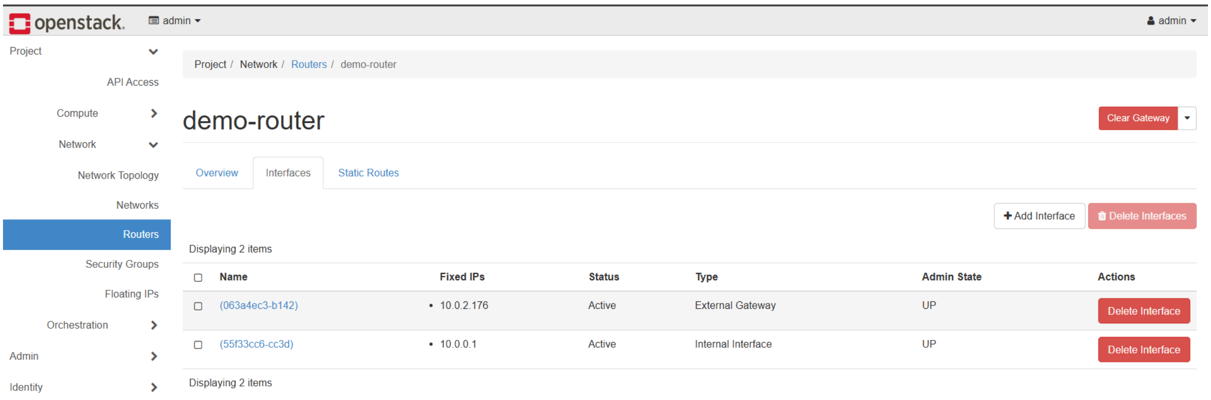
Project / Network / Routers

Routers

Displaying 1 item

□	Name	Status	External Network	Admin State	Availability Zones	Actions
□	demo-router	Active	public1	True	nova	Clear Gateway

Displaying 1 item



Project / Network / Routers / demo-router

demo-router

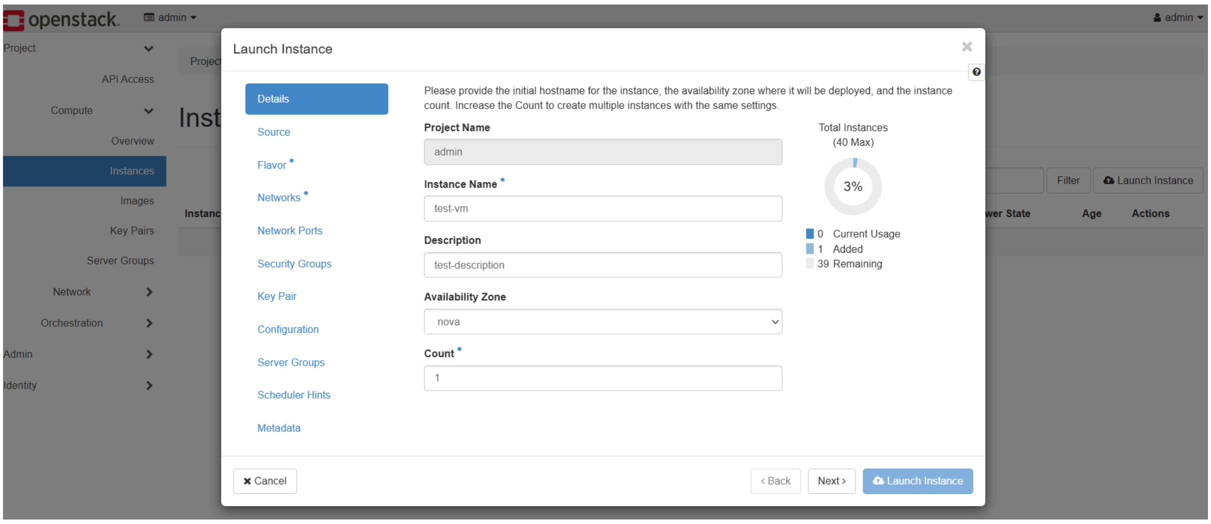
Displaying 2 items

□	Name	Fixed IPs	Status	Type	Admin State	Actions
□	(063a4ec3-b142)	• 10.0.2.176	Active	External Gateway	UP	Delete Interface
□	(55f33cc6-cc3d)	• 10.0.0.1	Active	Internal Interface	UP	Delete Interface

Displaying 2 items

Launching the instance and testing the connectivity the external networks:

Using the Cirros image and demo network, we created a small virtual machine



Launch Instance

Please provide the initial hostname for the instance, the availability zone where it will be deployed, and the instance count. Increase the Count to create multiple instances with the same settings.

Project Name: admin

Instance Name *: test-vm

Description: test-description

Availability Zone: nova

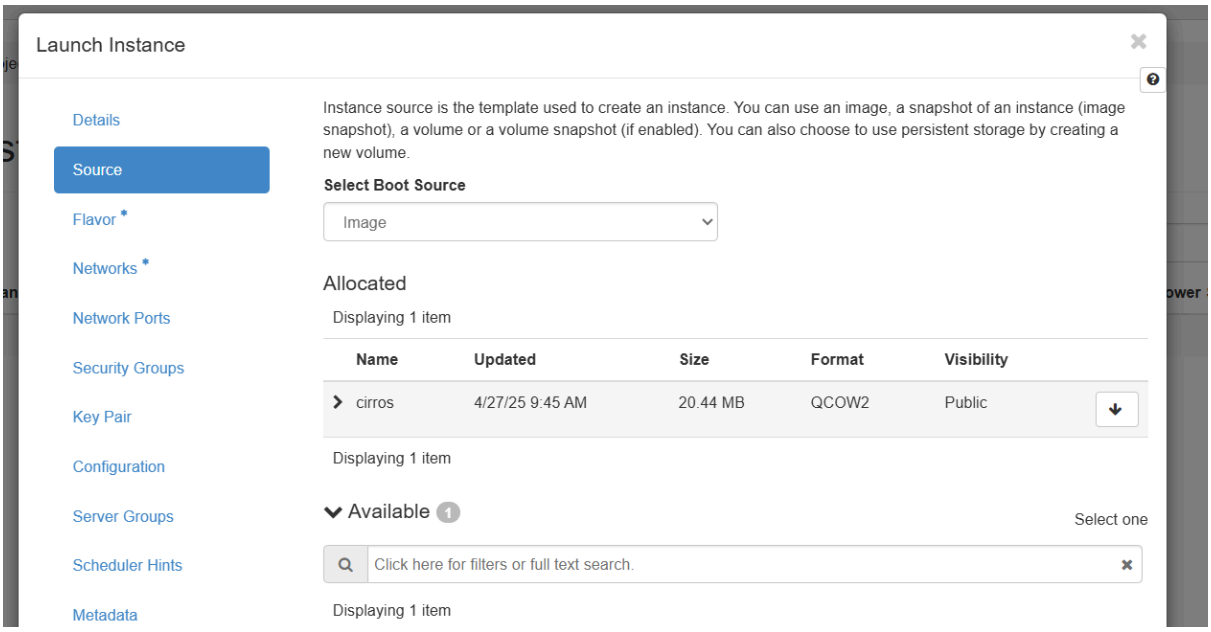
Count *: 1

Total Instances (40 Max)

3%

0 Current Usage
1 Added
39 Remaining

We selected the Cirros image that we uploaded earlier.



Launch Instance

Instance source is the template used to create an instance. You can use an image, a snapshot of an instance (image snapshot), a volume or a volume snapshot (if enabled). You can also choose to use persistent storage by creating a new volume.

Select Boot Source: Image

Allocated

Displaying 1 item

Name	Updated	Size	Format	Visibility
cirros	4/27/25 9:45 AM	20.44 MB	QCOW2	Public

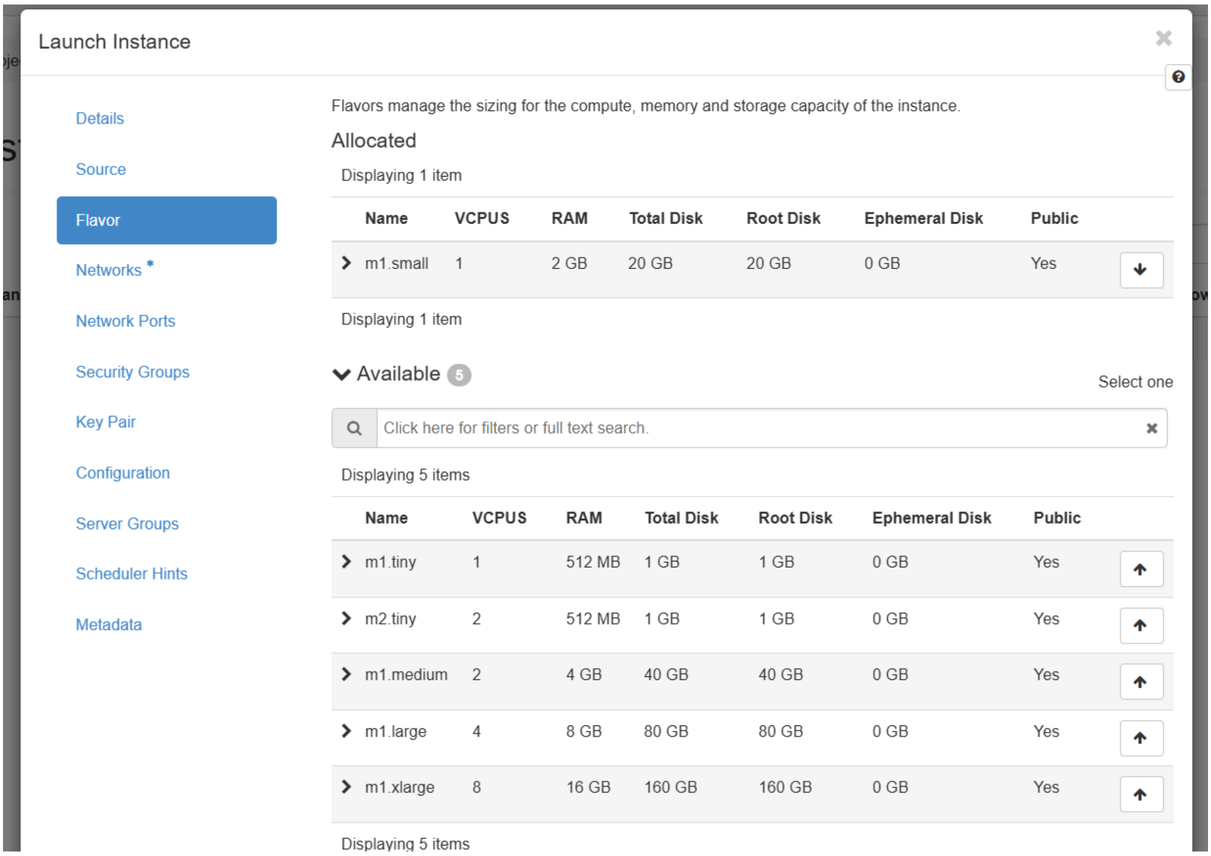
Displaying 1 item

Available 1 — Select one

Click here for filters or full text search.

Displaying 1 item

We selected Flavor: small one (`m1.small`)



Launch Instance

Flavors manage the sizing for the compute, memory and storage capacity of the instance.

Allocated

Displaying 1 item

Name	VCPUS	RAM	Total Disk	Root Disk	Ephemeral Disk	Public
m1.small	1	2 GB	20 GB	20 GB	0 GB	Yes

Displaying 1 item

Available 5 — Select one

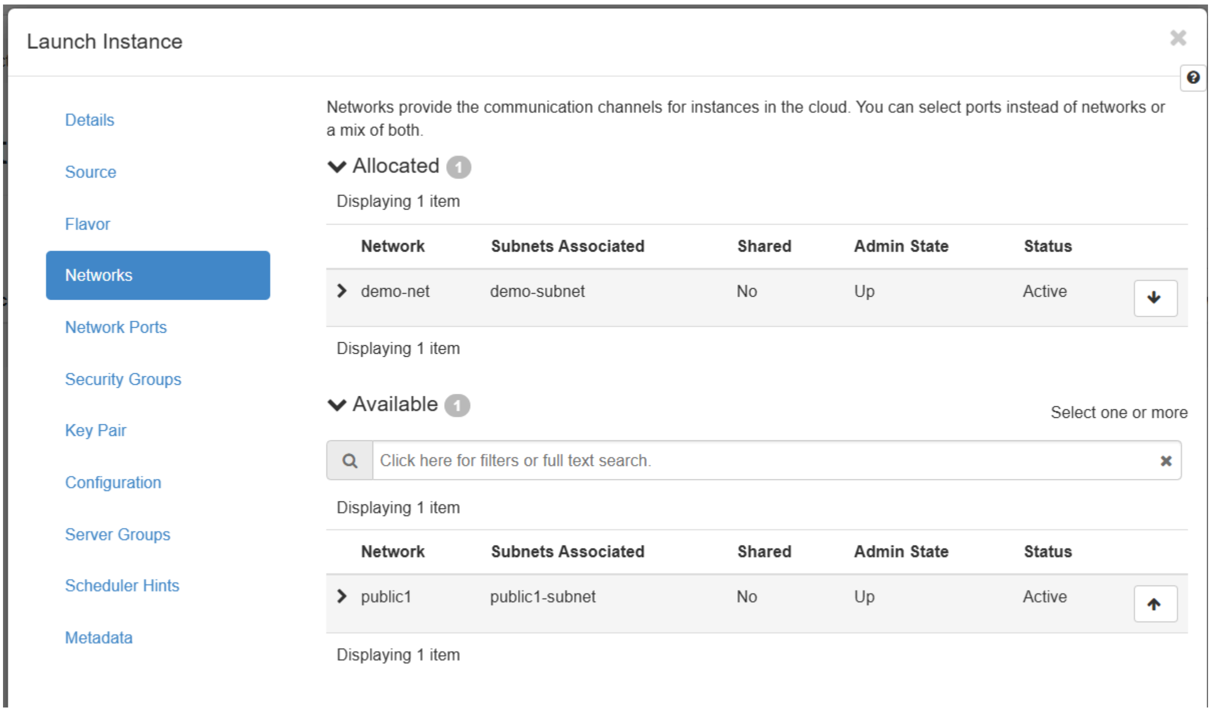
Click here for filters or full text search.

Displaying 5 items

Name	VCPUS	RAM	Total Disk	Root Disk	Ephemeral Disk	Public
m1.tiny	1	512 MB	1 GB	1 GB	0 GB	Yes
m2.tiny	2	512 MB	1 GB	1 GB	0 GB	Yes
m1.medium	2	4 GB	40 GB	40 GB	0 GB	Yes
m1.large	4	8 GB	80 GB	80 GB	0 GB	Yes
m1.xlarge	8	16 GB	160 GB	160 GB	0 GB	Yes

Displaying 5 items

Attach it to demo-net (the private network).



Launch Instance

Networks provide the communication channels for instances in the cloud. You can select ports instead of networks or a mix of both.

Allocated 1

Displaying 1 item

Network	Subnets Associated	Shared	Admin State	Status
demo-net	demo-subnet	No	Up	Active

Displaying 1 item

Available 1 — Select one or more

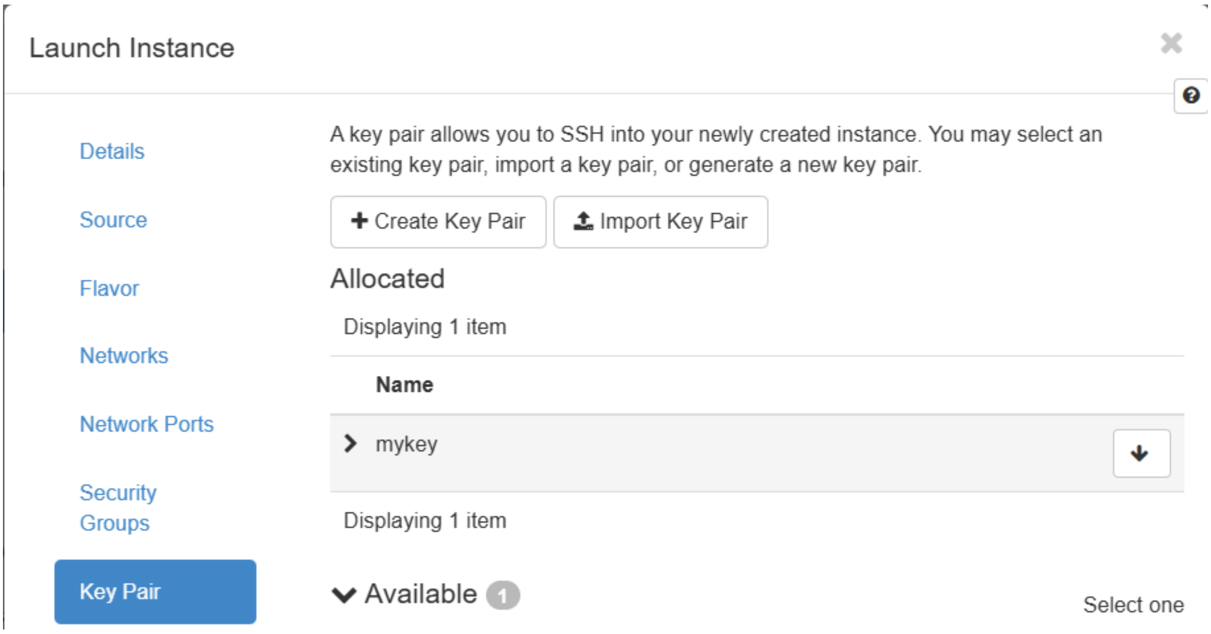
Click here for filters or full text search.

Displaying 1 item

Network	Subnets Associated	Shared	Admin State	Status
public1	public1-subnet	No	Up	Active

Displaying 1 item

Attaching key pair:



Launch Instance

A key pair allows you to SSH into your newly created instance. You may select an existing key pair, import a key pair, or generate a new key pair.

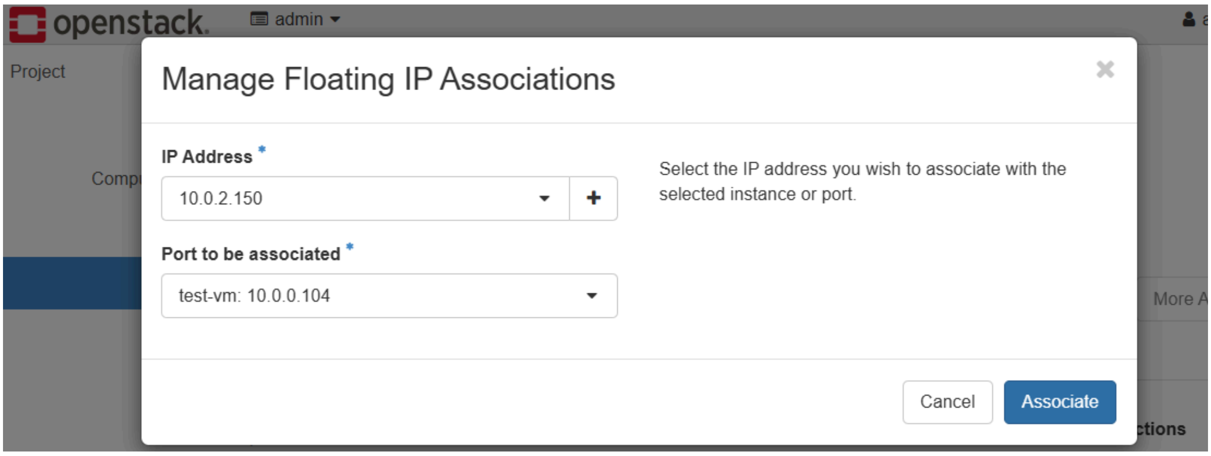
Allocated

Displaying 1 item

Name
mykey

Displaying 1 item

Associate the floating IP `10.0.2.150` to our instance (`test-vm`) in the OpenStack Dashboard:



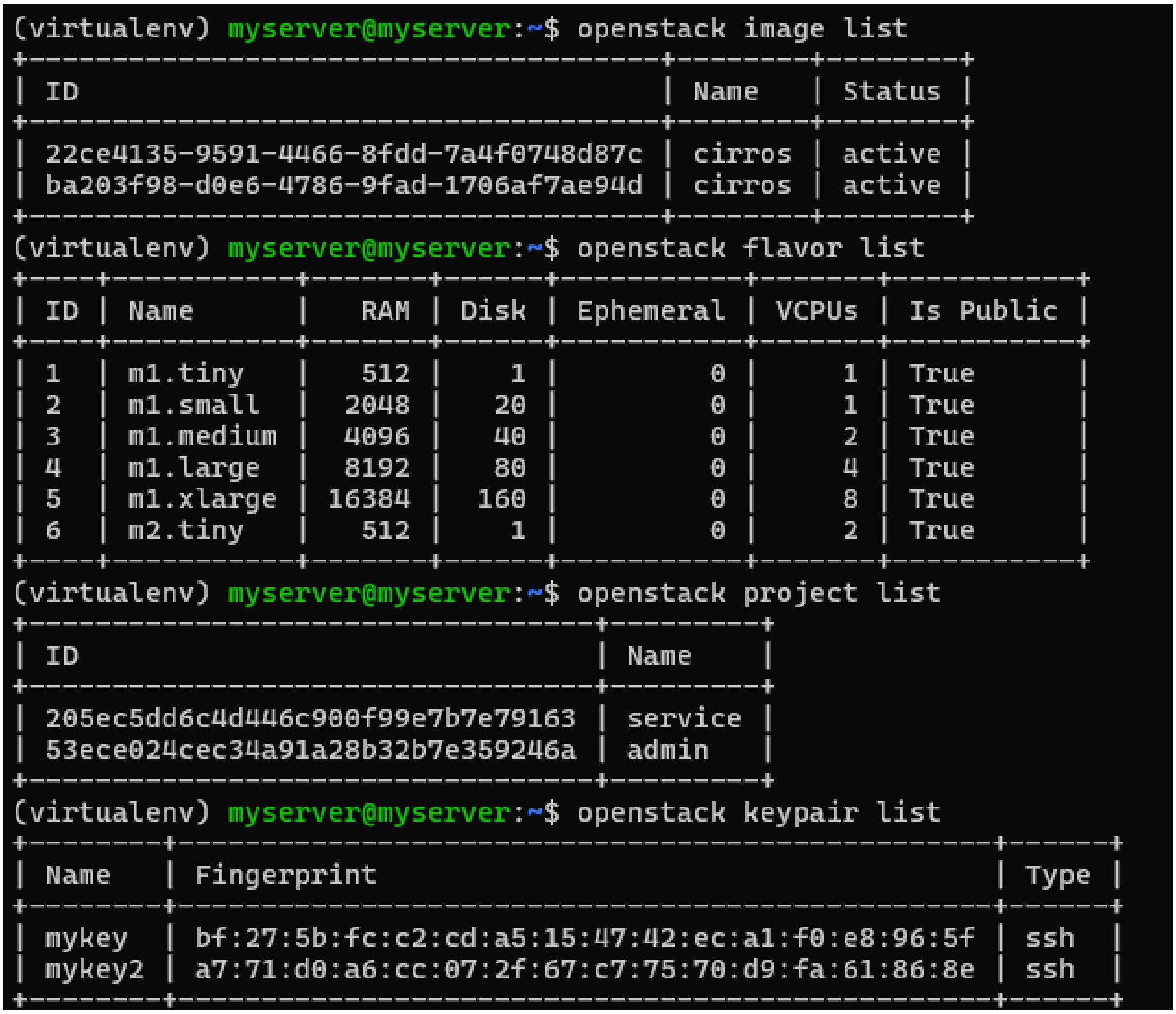
Manage Floating IP Associations

IP Address *: 10.0.2.150

Port to be associated *: test-vm: 10.0.0.104

Select the IP address you wish to associate with the selected instance or port.

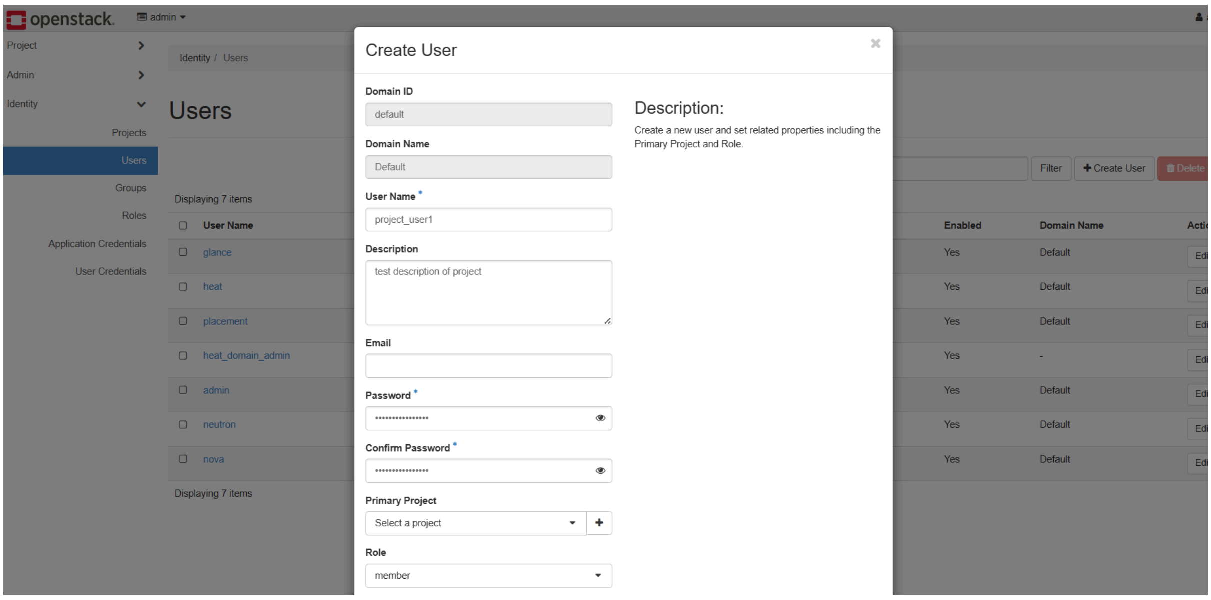
The information: image, flavor and etc.



```
(virtualenv) myserver@myserver:~$ openstack image list
+--------------------------------------+--------+--------+
| ID                                   | Name   | Status |
+--------------------------------------+--------+--------+
| 22ce4135-9591-4466-8fdd-7a4f0748d87c | cirros | active |
| ba203f98-d0e6-4786-9fad-1706af7ae94d | cirros | active |
+--------------------------------------+--------+--------+
(virtualenv) myserver@myserver:~$ openstack flavor list
+----+-----------+-------+------+-----------+-------+-----------+
| ID | Name      |   RAM | Disk | Ephemeral | VCPUs | Is Public |
+----+-----------+-------+------+-----------+-------+-----------+
| 1  | m1.tiny   |   512 |    1 |         0 |     1 | True      |
| 2  | m1.small  |  2048 |   20 |         0 |     1 | True      |
| 3  | m1.medium |  4096 |   40 |         0 |     2 | True      |
| 4  | m1.large  |  8192 |   80 |         0 |     4 | True      |
| 5  | m1.xlarge | 16384 |  160 |         0 |     8 | True      |
| 6  | m2.tiny   |   512 |    1 |         0 |     2 | True      |
+----+-----------+-------+------+-----------+-------+-----------+
(virtualenv) myserver@myserver:~$ openstack project list
+----------------------------------+---------+
| ID                               | Name    |
+----------------------------------+---------+
| 205ec5dd6c4d446c900f99e7b7e79163 | service |
| 53ece024cec34a91a28b32b7e359246a | admin   |
+----------------------------------+---------+
(virtualenv) myserver@myserver:~$ openstack keypair list
+--------+-------------------------------------------------+------+
| Name   | Fingerprint                                     | Type |
+--------+-------------------------------------------------+------+
| mykey  | bf:27:5b:fc:c2:cd:a5:15:47:42:ec:a1:f0:e8:96:5f | ssh  |
| mykey2 | a7:71:d0:a6:cc:07:2f:67:c7:75:70:d9:fa:61:86:8e | ssh  |
+--------+-------------------------------------------------+------+
```

Openstack administration: manage user project

Creating new user:



Users

Displaying 7 items

User Name	Enabled	Domain Name
glance	Yes	Default
heat	Yes	Default
placement	Yes	Default
heat_domain_admin	Yes	-
admin	Yes	Default
neutron	Yes	Default
nova	Yes	Default

Displaying 7 items

Create User

Domain ID: default

Domain Name: Default

User Name *: project_user1

Description: test description of project

Email:

Password *: ••••••••••••••••

Confirm Password *: ••••••••••••••••

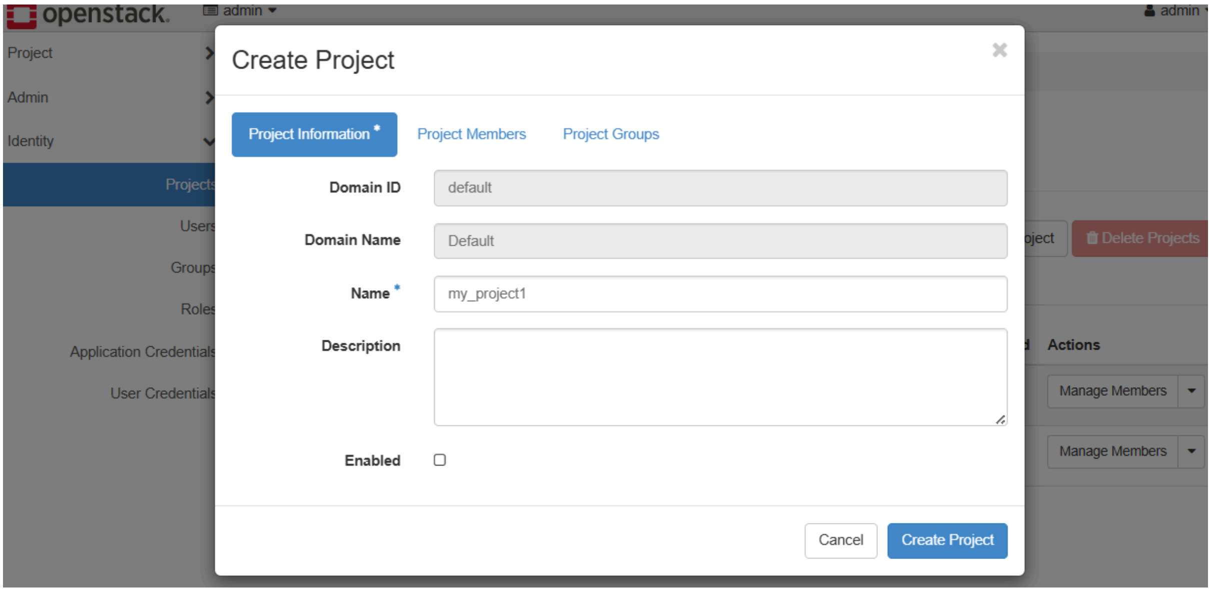
Primary Project: Select a project

Role: member

Description:

Create a new user and set related properties including the Primary Project and Role.

Creating new project:



Create Project

Project Information * | Project Members | Project Groups

Domain ID: default

Domain Name: Default

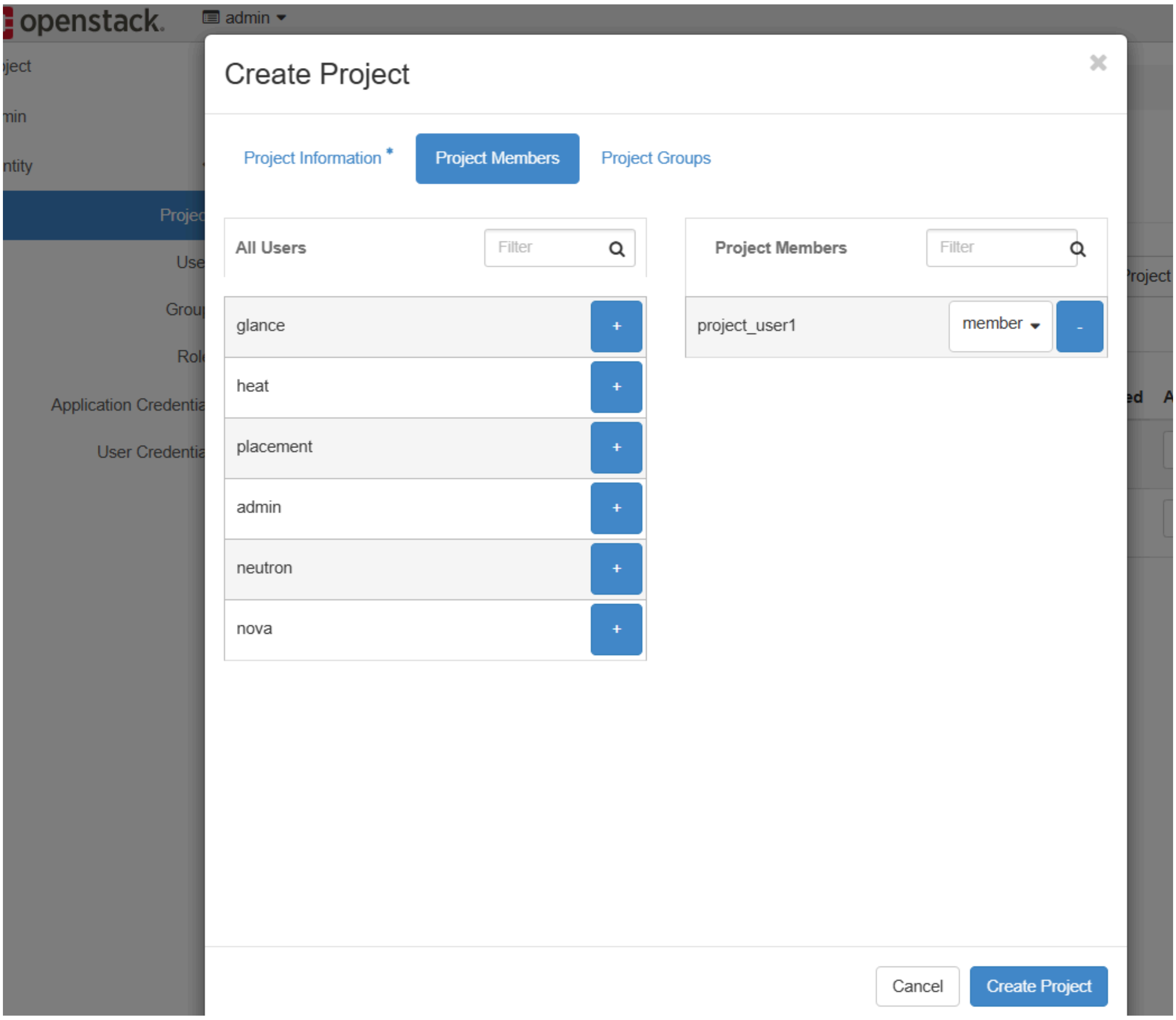
Name *: my_project1

Description:

Enabled ☐

Cancel | Create Project

Adding the user to the new project as Project Member.

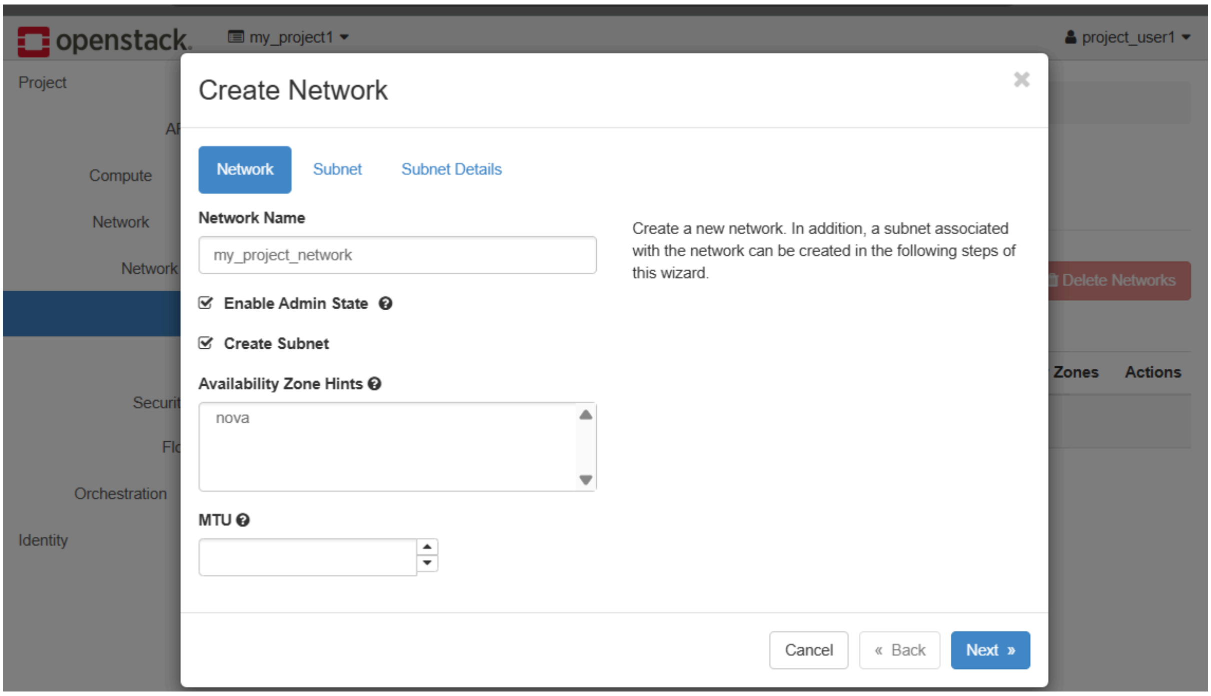


Create Project

All Users	
glance	+
heat	+
placement	+
admin	+
neutron	+
nova	+

Project Members		
project_user1	member	-

Creating network in new user:



Create Network

Network Name: my_project_network

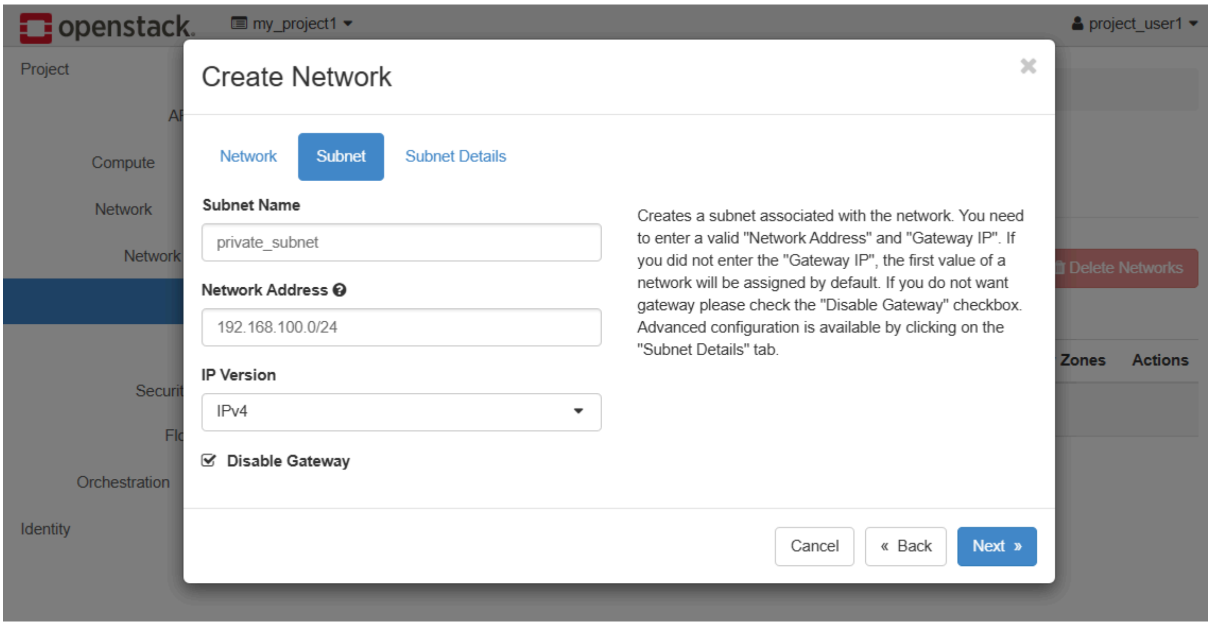
☑ Enable Admin State

☑ Create Subnet

Availability Zone Hints: nova

MTU:

Create a new network. In addition, a subnet associated with the network can be created in the following steps of this wizard.



Create Network

Subnet Name: private_subnet

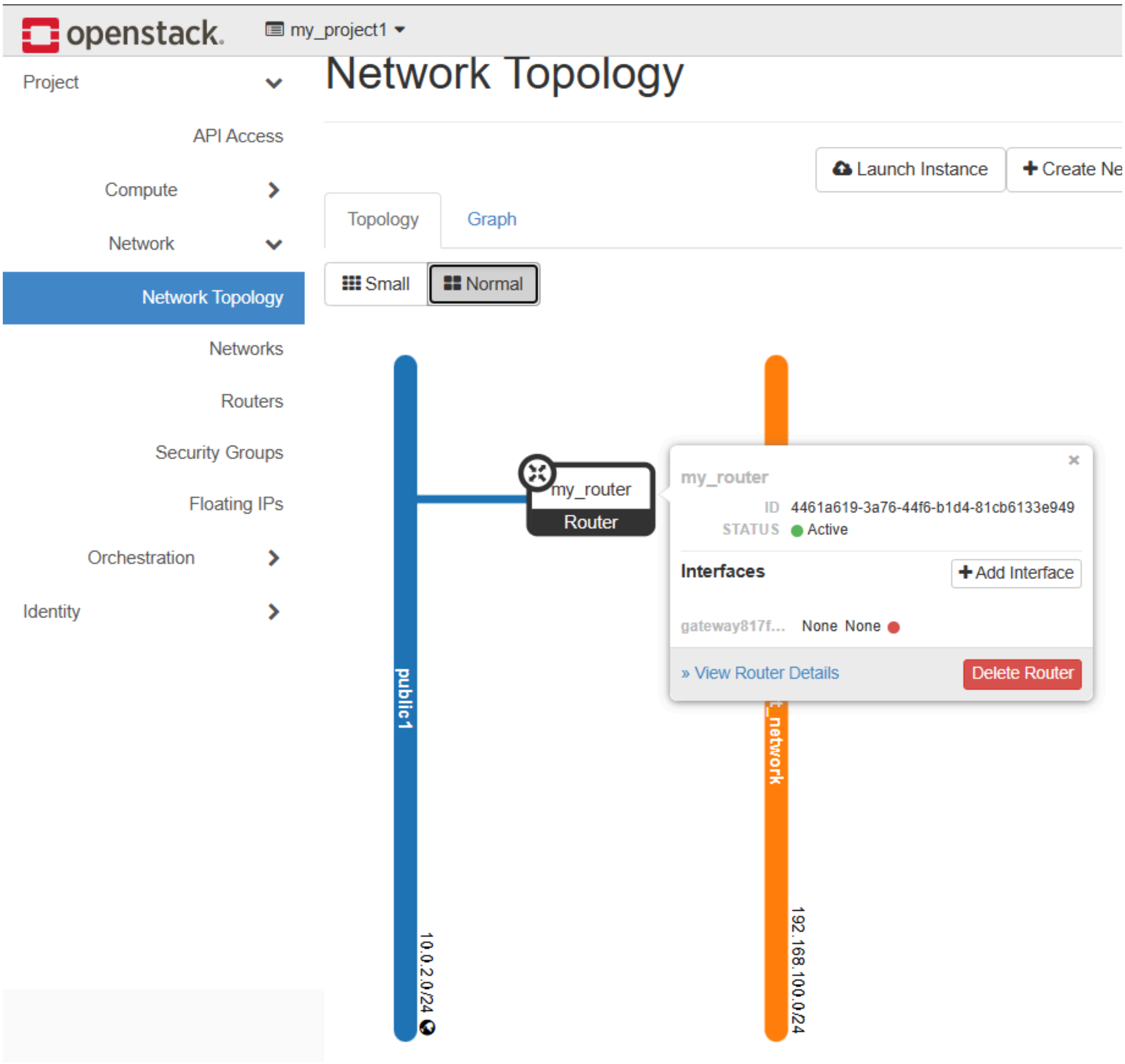
Network Address: 192.168.100.0/24

IP Version: IPv4

☑ Disable Gateway

Creates a subnet associated with the network. You need to enter a valid "Network Address" and "Gateway IP". If you did not enter the "Gateway IP", the first value of a network will be assigned by default. If you do not want gateway please check the "Disable Gateway" checkbox. Advanced configuration is available by clicking on the "Subnet Details" tab.

Displaying the Network topology and our router is connected to the « public network »



Network Topology

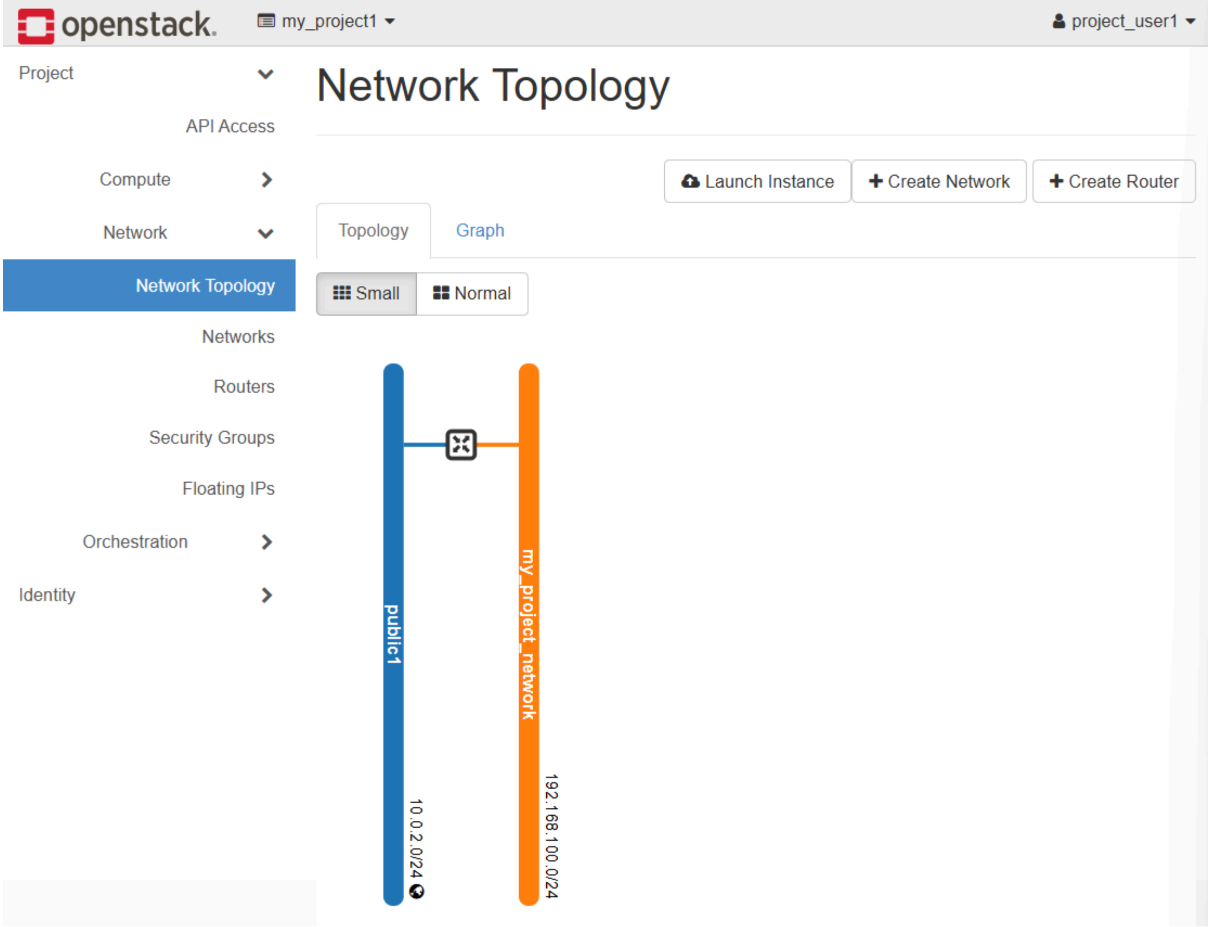
my_router

ID 4461a619-3a76-44f6-b1d4-81cb6133e949
STATUS ● Active

Interfaces

gateway817f... None None ●

Displaying toplogy, we see that our private network is attached to the router:



Conclusion

After completion of the whole project, we feel way more confident with OpenStack. We gained valuable hands-on experience with OpenStack deployment using Kolla-Ansible. We now understand how to use Kolla-Ansible to deploy OpenStack on one single node VM, how to configure networks, launch instances, and manage cloud resources through both command-line tools and the Horizon dashboard. We also realized that YAML files. This project helped solidify our understanding of the architecture and functionality of OpenStack components. It also highlighted the importance of system preparation, proper configuration management, and troubleshooting skills when working with complex cloud platforms. This project served as a strong introduction to cloud infrastructure management and provided a solid foundation for more advanced OpenStack use cases in the future. In the end, everything worked, and we saw our VM running inside the cloud that we built ourselves.