ZYC2 User Guide

Raphael Polig
Version: May 28, 2020

Connecting to ZYC2

In order to use the ZYC2 infrastructure you will need to connect to it via a VPN client. You should have received an OpenVPN configuration file together with your ZYC2 credentials upon registration.

INFO: The OpenVPN client must be version 2.4 or later

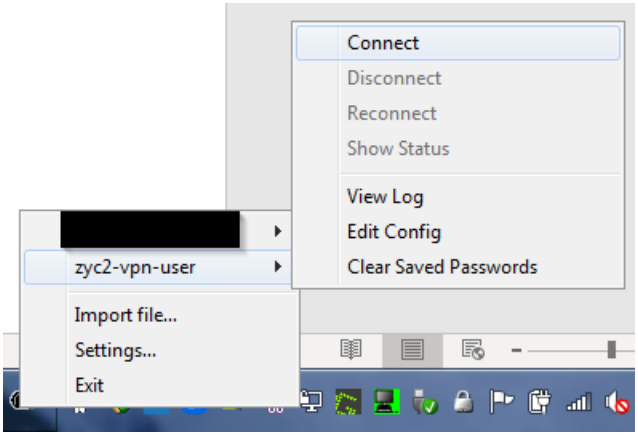
Linux

On most Linux distributions you can either import the OpenVPN configuration file to the Network Manager or use the file directly with the openvpn client.

Windows

On Windows you can download the OpenVPN client software (Windows Installer) from here: https://openvpn.net/community-downloads/

After installation you can launch the OpenVPN GUI and a small tray icon should appear. If you right click on it you can select “Import file…” and navigate to the ZYC2 OpenVPN configuration file. After successful import, the connection should appear in the menu as shown in the figure below.



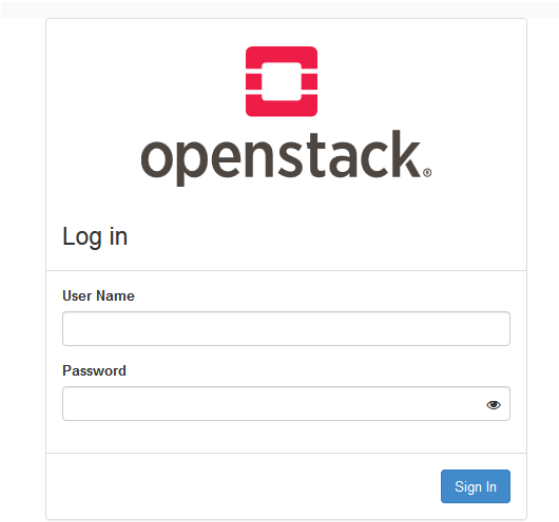
When you click connect you will be prompted a username and password. These are the credentials provided in the registration email.

Mac

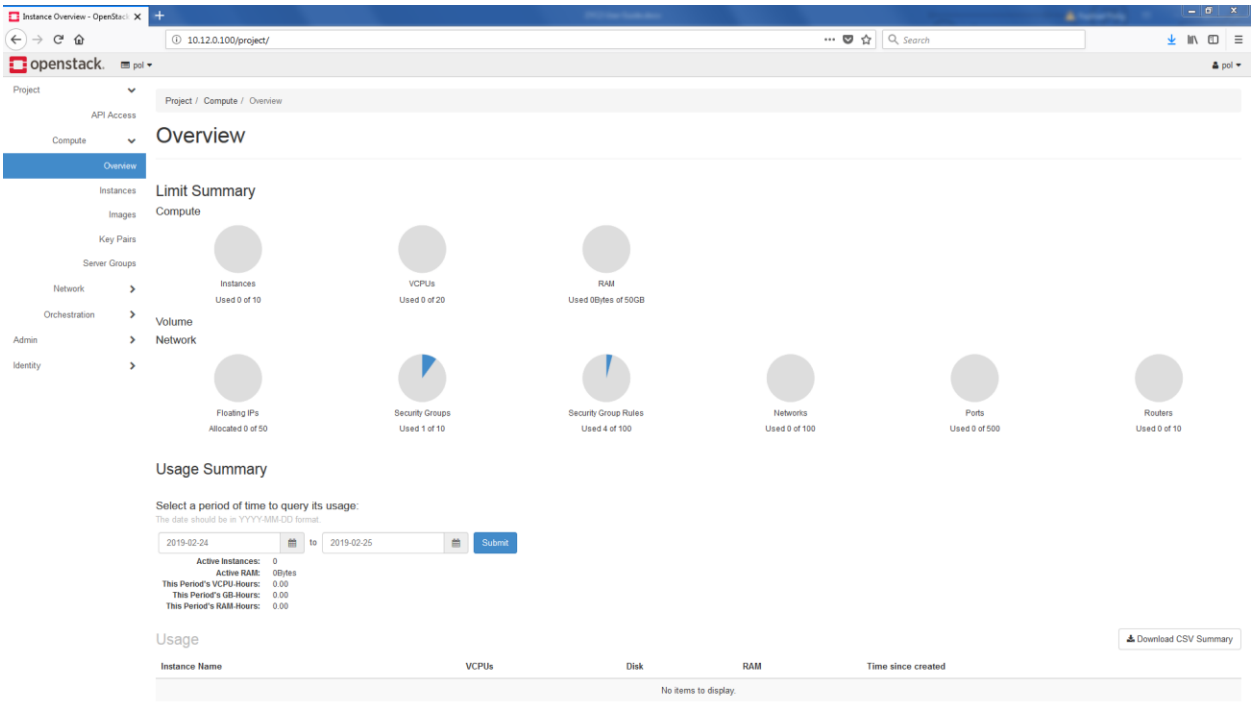
To be tested

Getting started with OpenStack

ZYC2 runs OpenStack to provide an Infrastructure-as-a-Service cloud. This means you can create your own virtual networks and machines. The main point of interaction is the OpenStack Dashboard which can be reached via http://10.12.0.201/

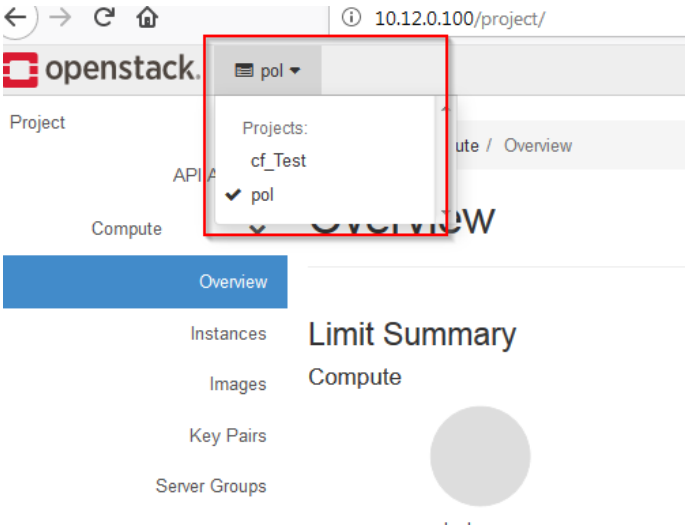


You can login using you ZYC2 credentials. After successful login you are presented an overview of your current project resource usage.



Projects

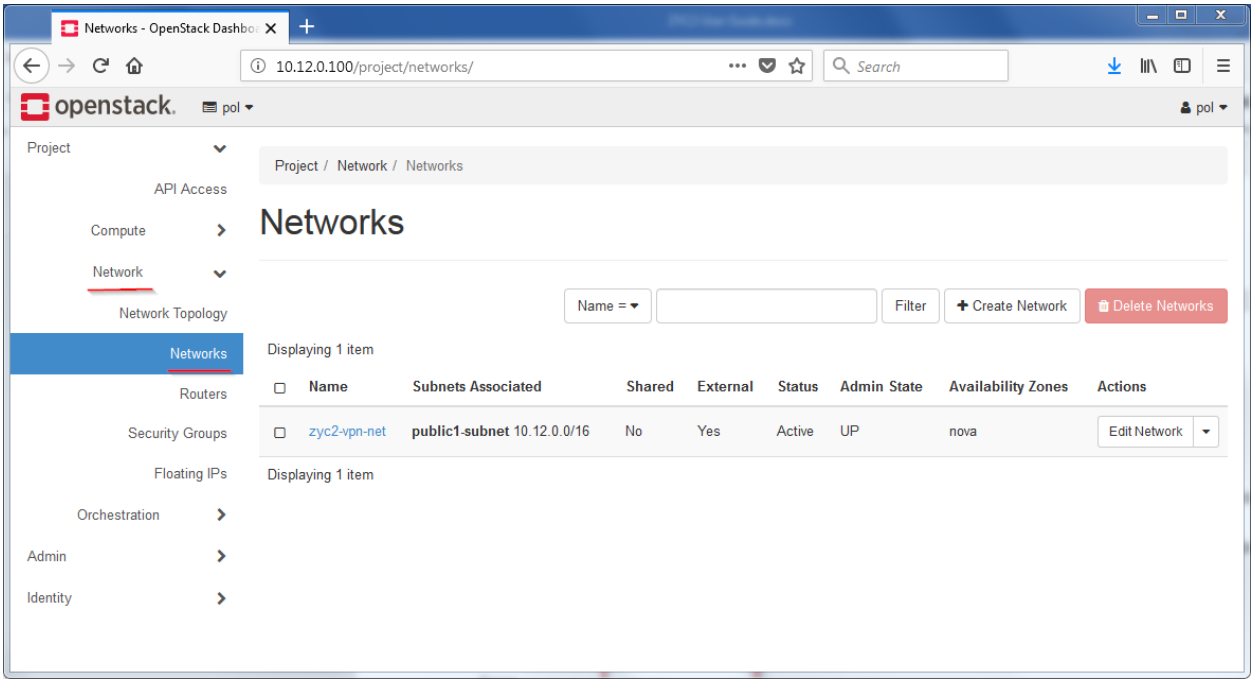
On the top left of the dashboard you can see the project you currently work with. If you click on it a drop-down list will appear of all projects, you are a member of. All members of a project can see and modify the virtual infrastructure of the project. A project also has its own resource limits. You will always be a member of your own project which is named as your username.



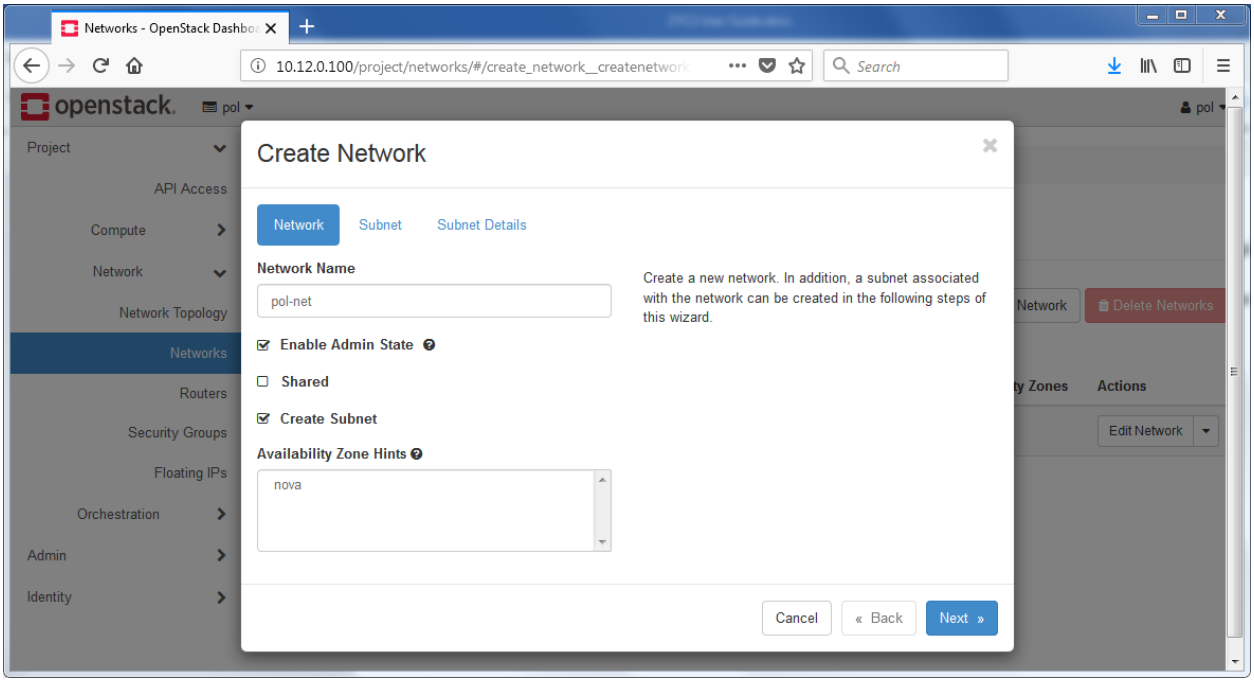
Networks

An important aspect of the infrastructure is the network. In OpenStack you may create multiple private networks that belong to your project. Think of them as your “home network”. After creation of your network you may add virtual machines to it. All machines on the same network can communicate with each other.

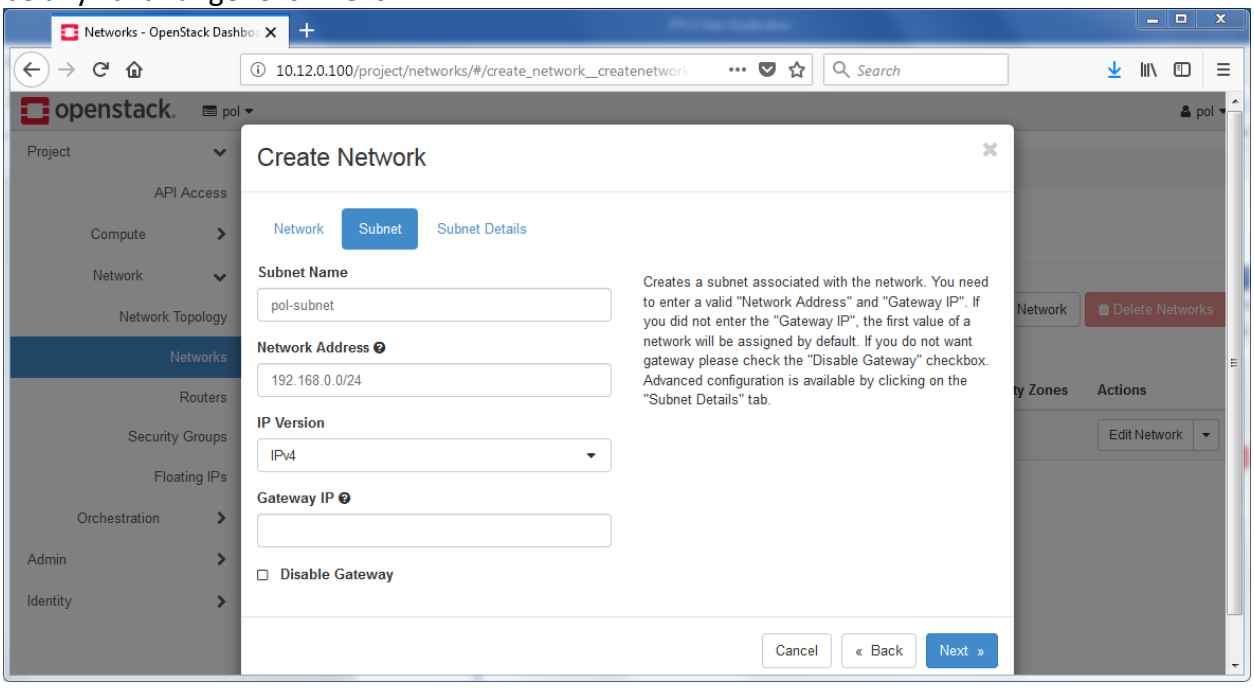
Let’s create our first network. In the dashboard menu on the left select Network and then click Networks. An overview of networks available to you is shown. The zyc2-vpn-net is available to all projects as the external network. It will come into play later. Click on Create Network to start the wizard.



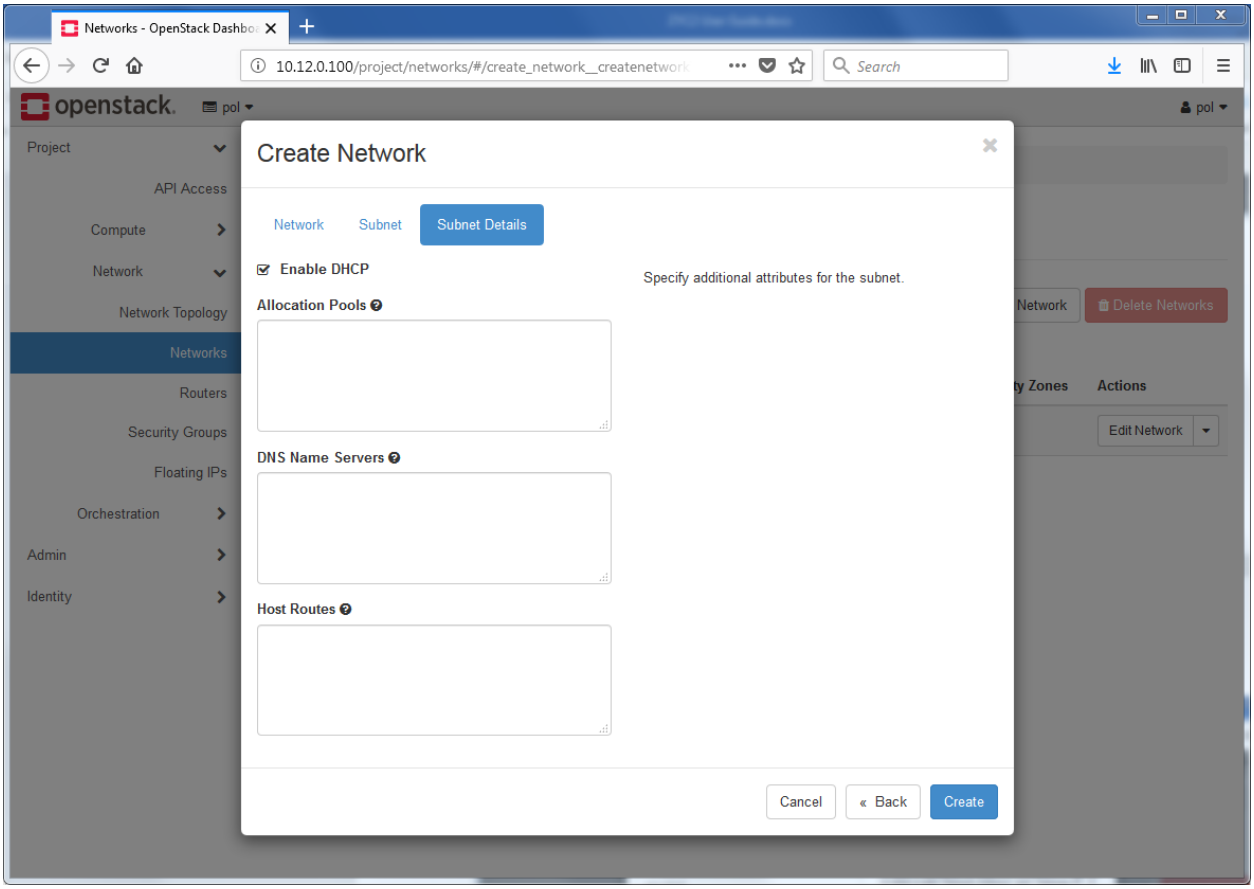
Enter a network name, leave the rest default (as in screenshot) and click **Next**.



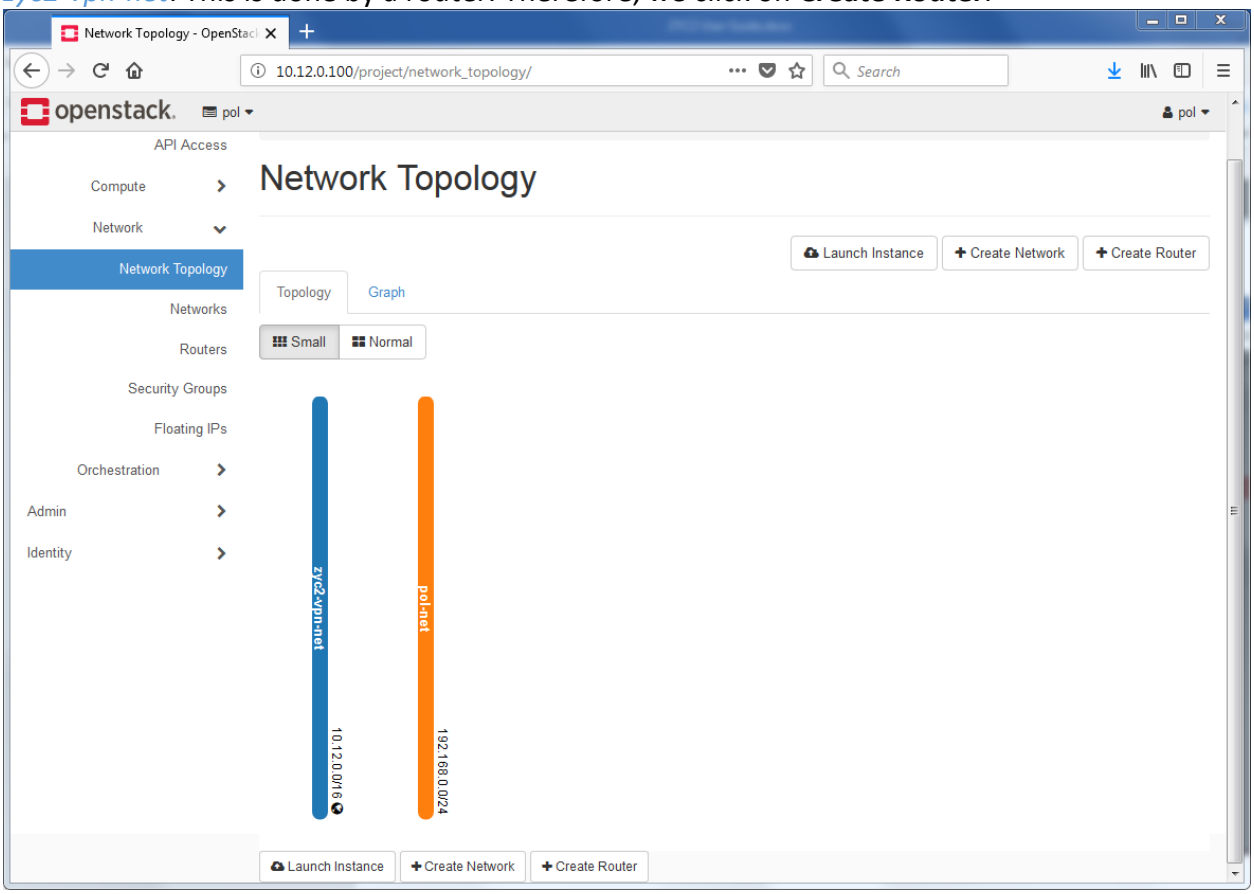
Enter a subnet name and a network address range. *192.168.0.0/24* is a good option but it can be any valid range. Click **Next**.



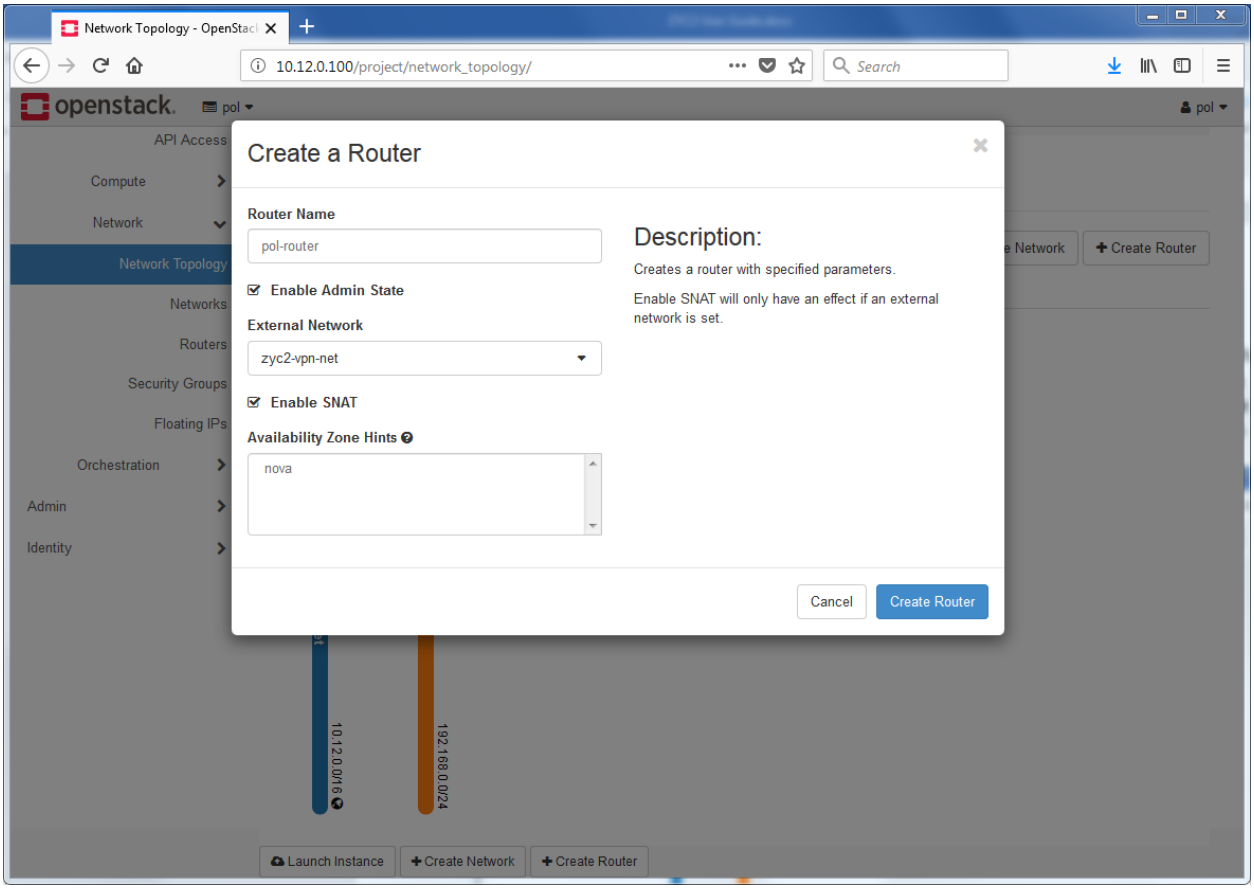
You can leave all field blank on the details page and click **Create**.



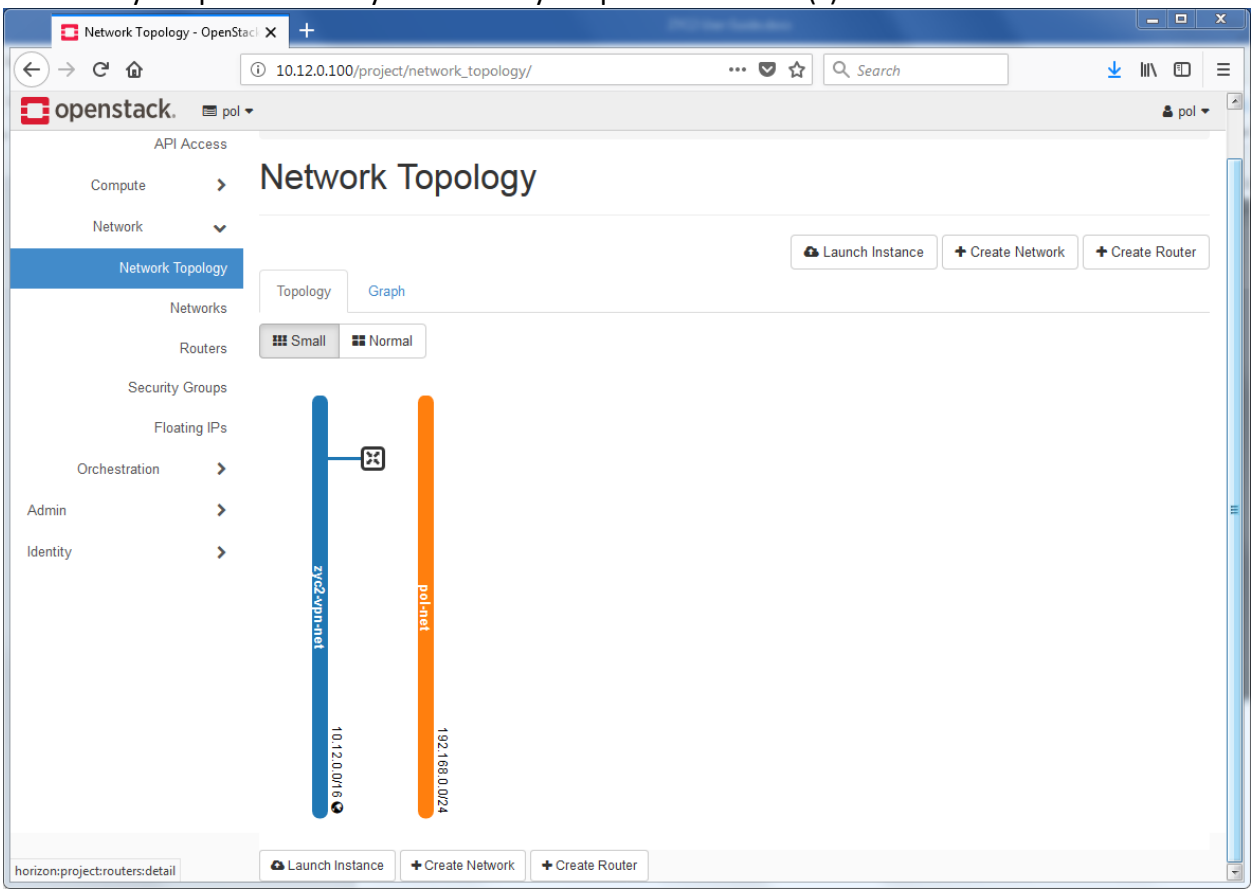
If you now go to **Network Topology** you can see the two networks available to your project. The external *zyc2-vpn-net* and your private network. The external network provides access to the VPN network. If you want to SSH from your VPN client into your virtual machines which reside on your private network, we must create a connection from your private network to the *zyc2-vpn-net*. This is done by a router. Therefore, we click on **Create Router**.



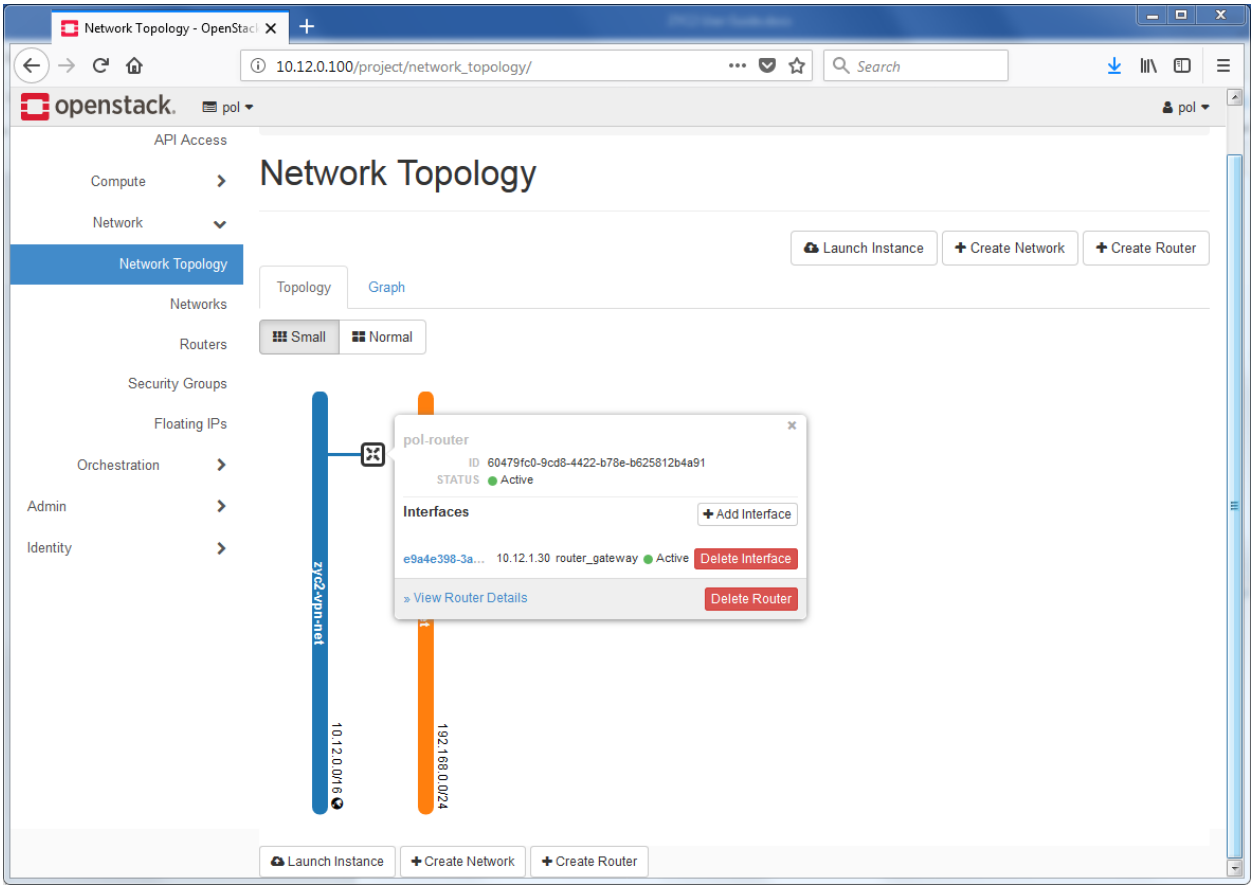
Provide a router name and select *zyc2-vpn-net* as the external network. Leave SNAT and Admin State checked and click **Create Router**.



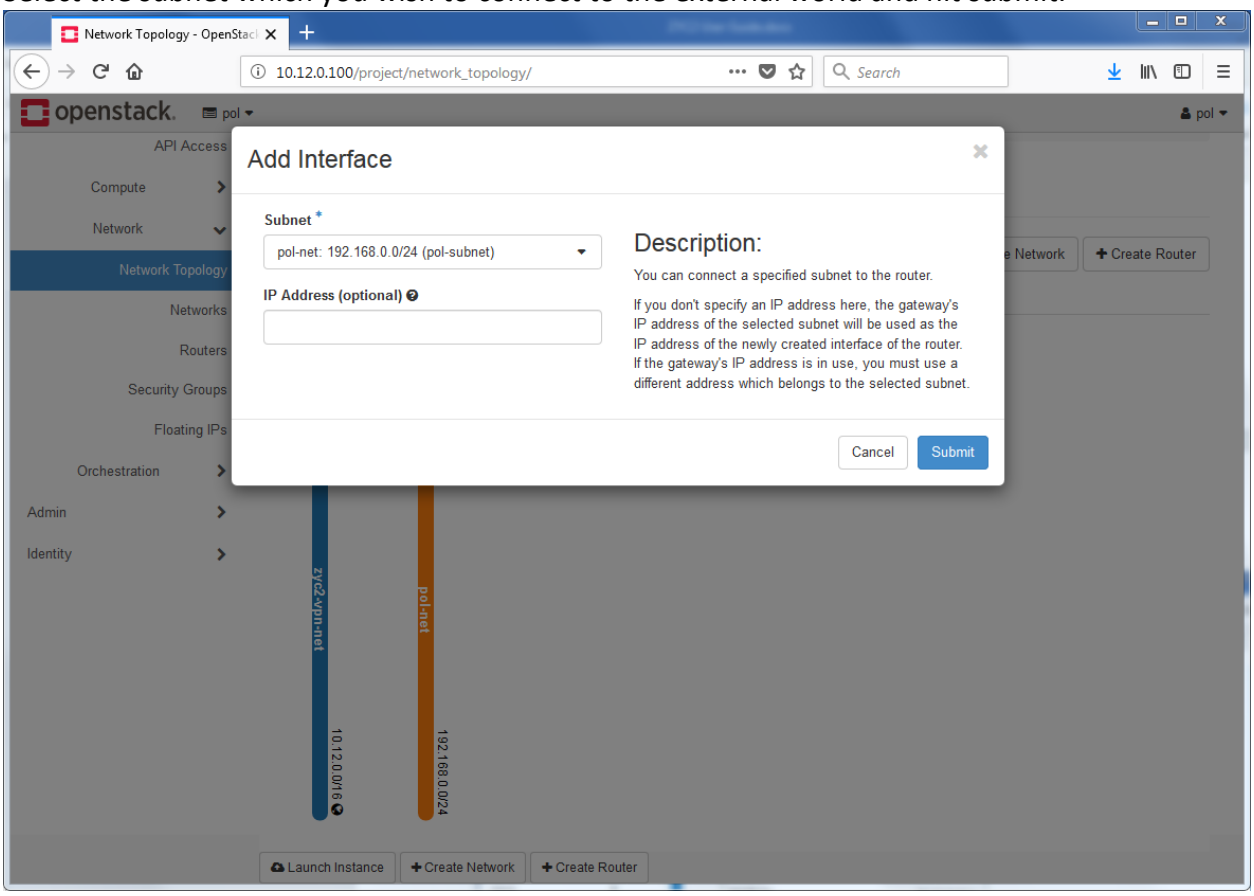
After creation of the router it will appear in the network topology and is connected to the external zyc2-vpn-net. Now you can add your private network(s) to this router.



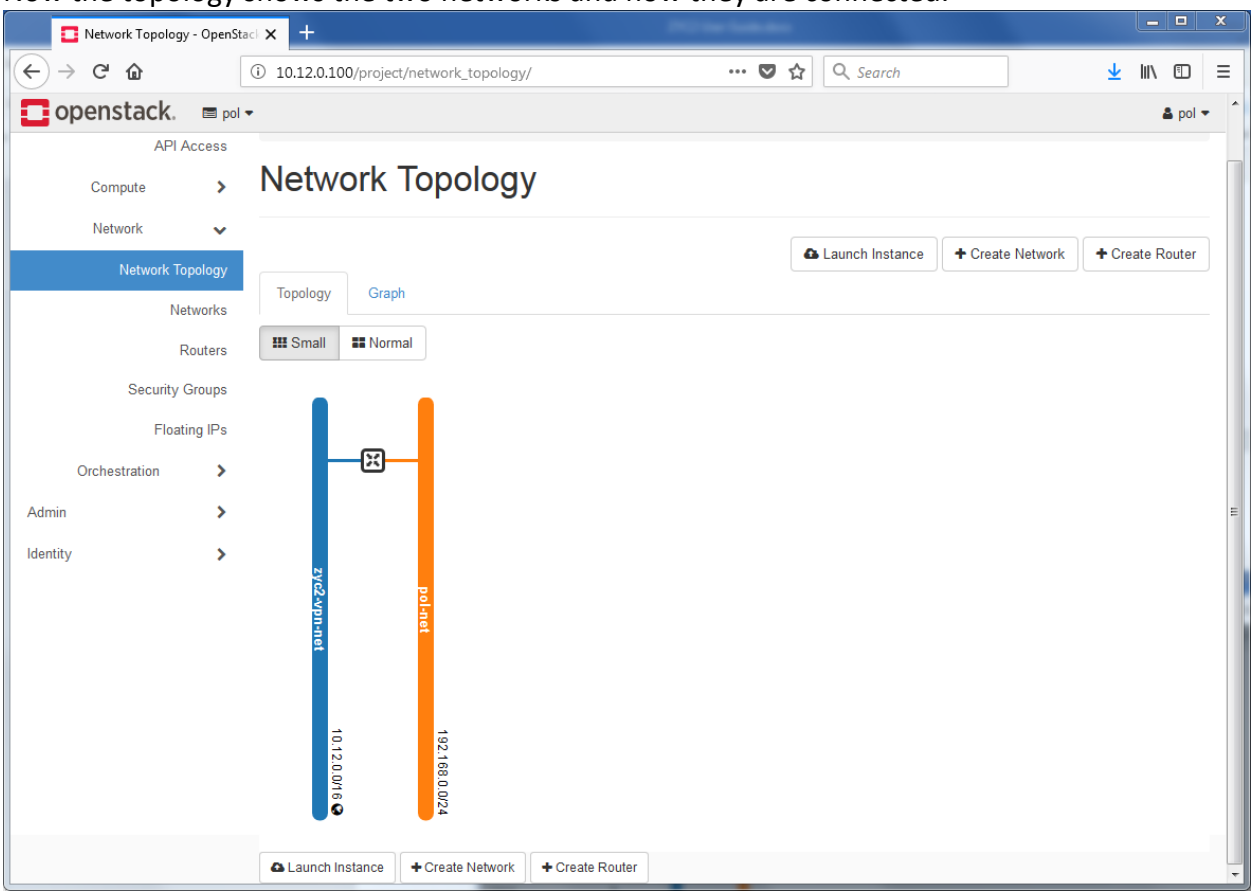
Hover over the router icon and a bubble will appear. Click on Add interface to add your network.



Select the subnet which you wish to connect to the external world and hit submit.



Now the topology shows the two networks and how they are connected.

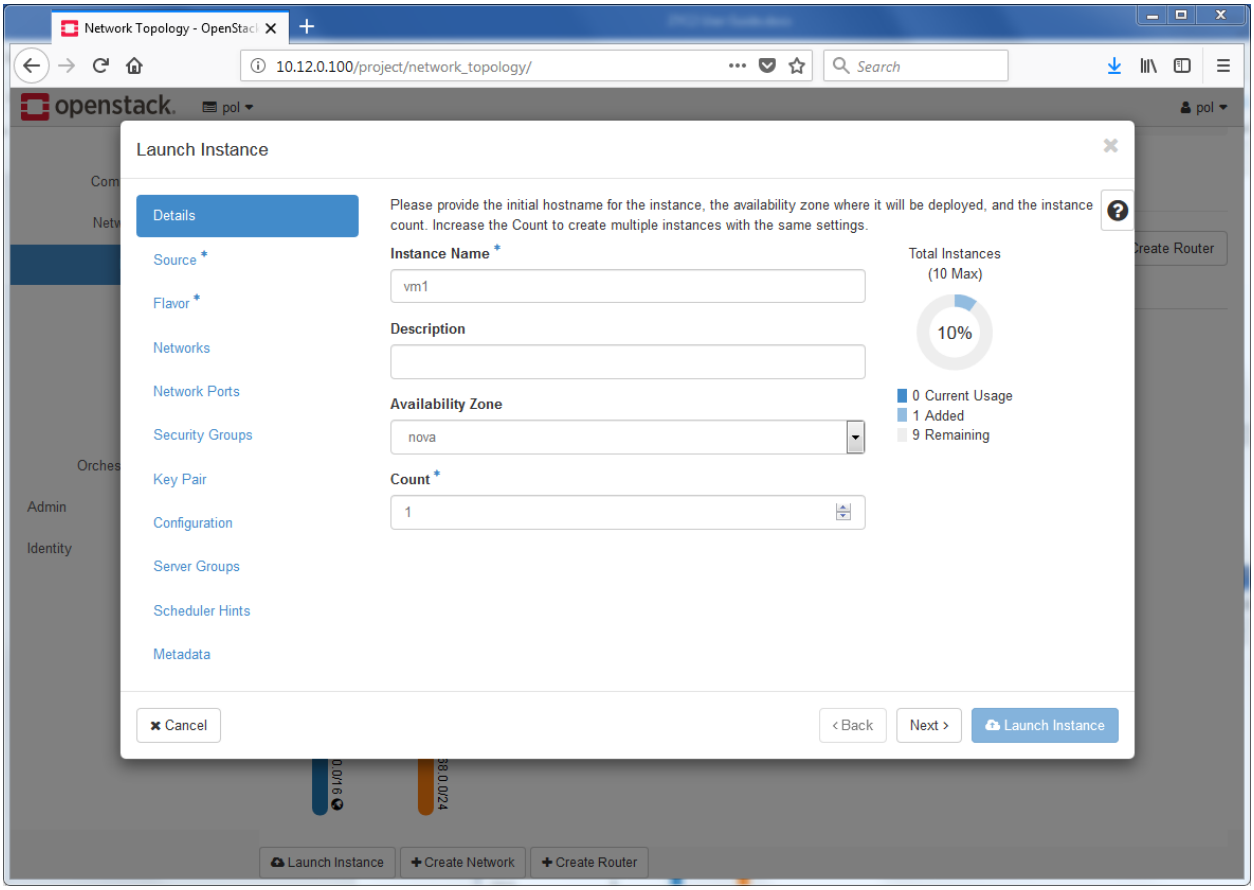


Now your private network can communicate with the outside world.

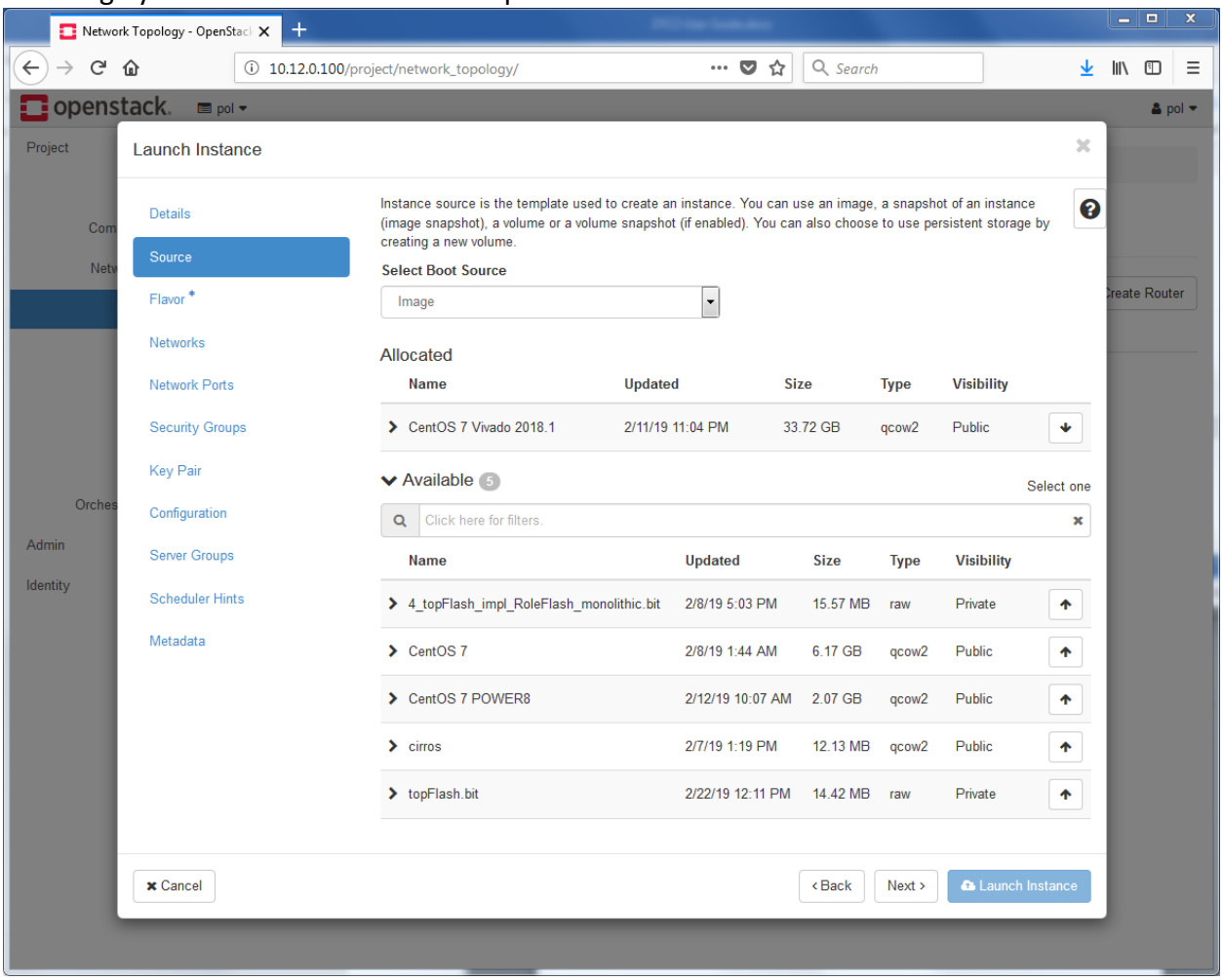
Compute

To create a virtual machine, you navigate to Compute and then Instances. This gives you an overview of the current virtual machines in your project.

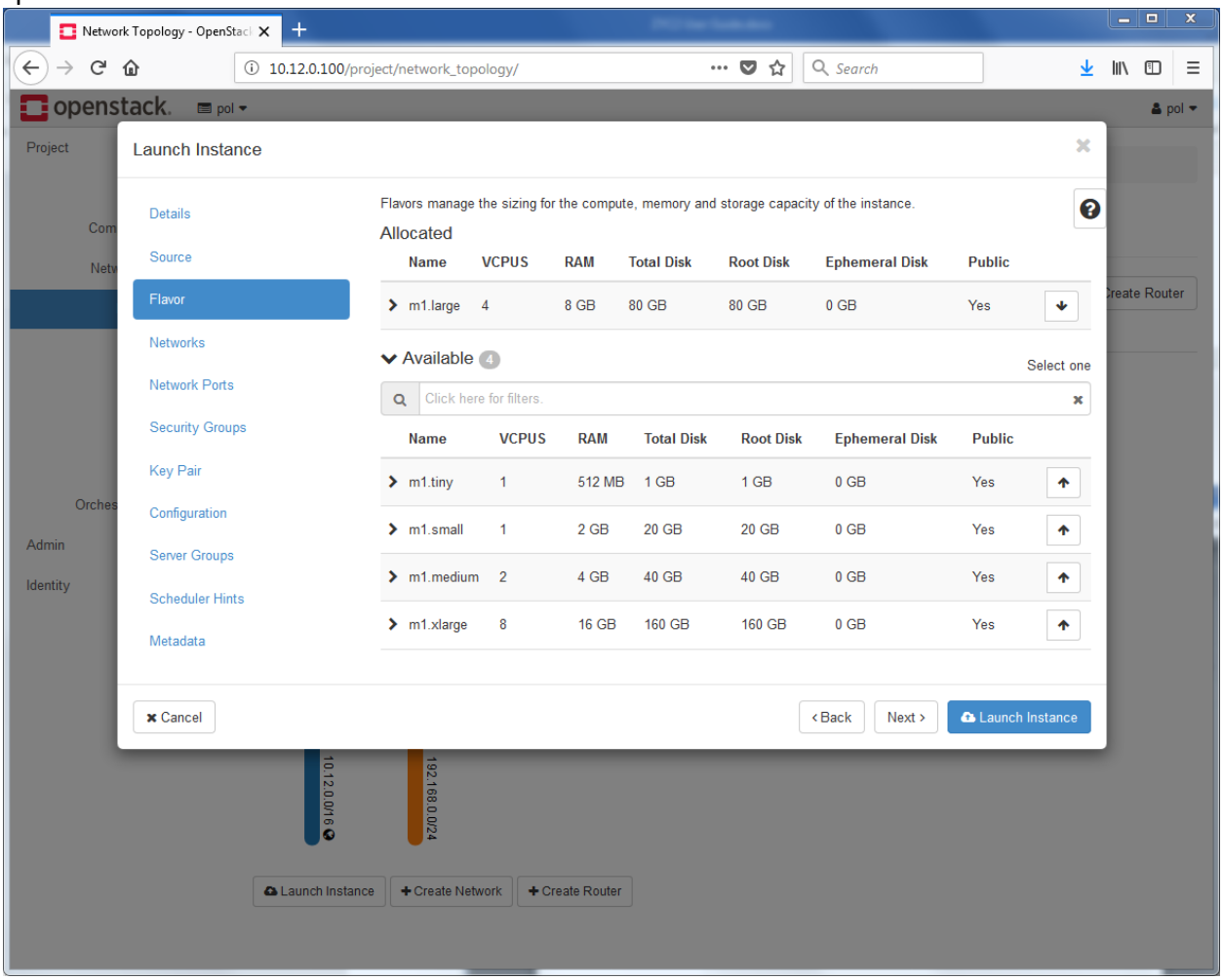
You can click on Launch instance to create a new VM. When the wizard pops up, give the instance a name and click next.



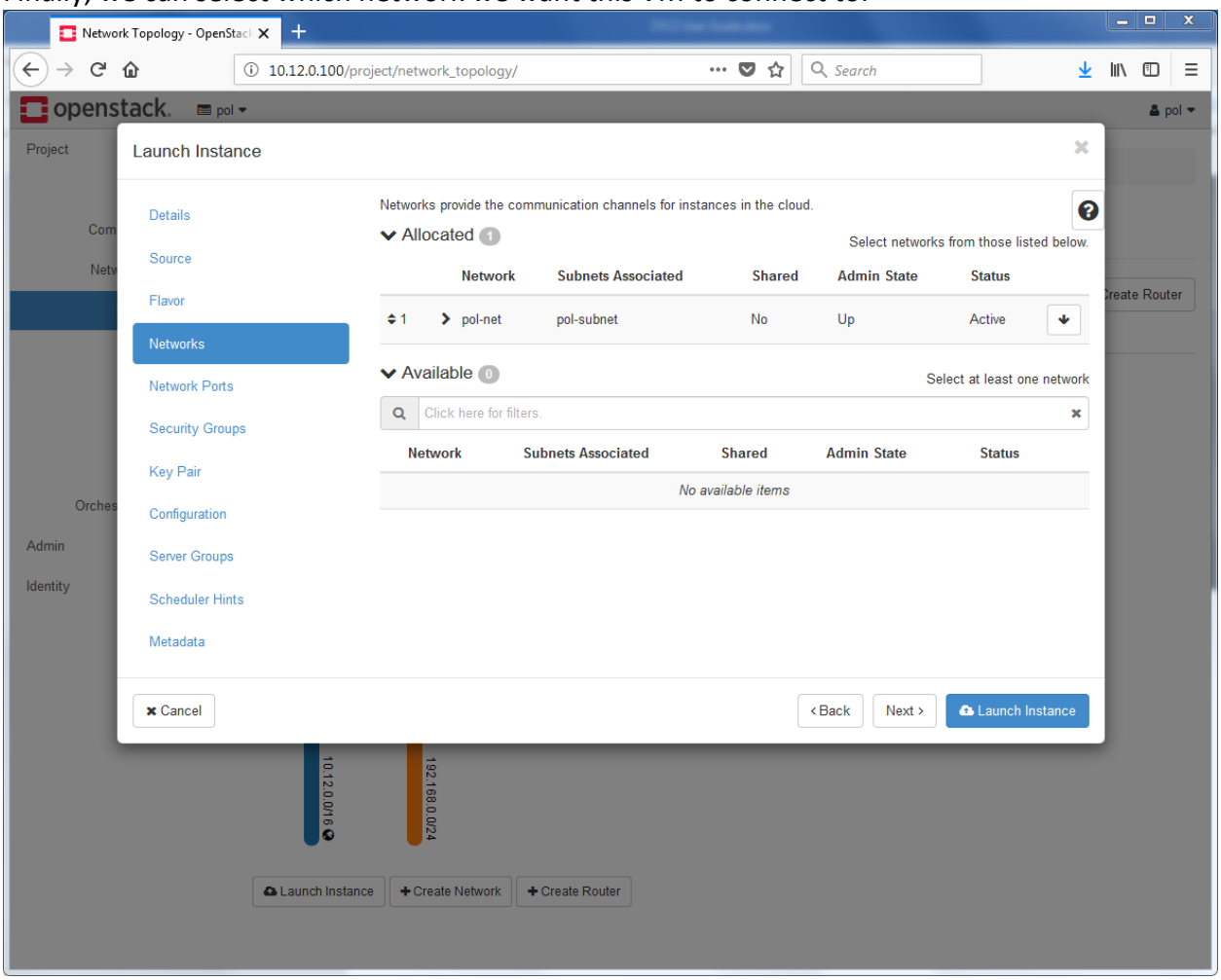
Next you can select the image you want the VM to run. Click on the arrow on the right to select the image you want to run. In the example CentOS with installed Xilinx tools will be launched.



Now we can select a flavor of the virtual machine, meaning how many CPUs, memory and disk space it should have.

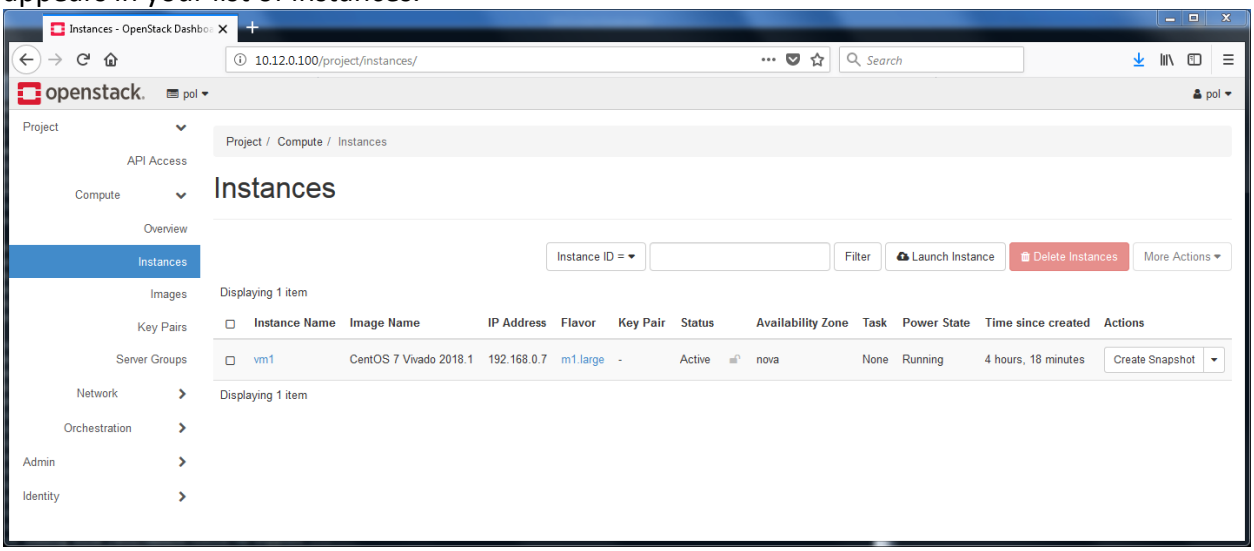


Finally, we can select which network we want this VM to connect to.

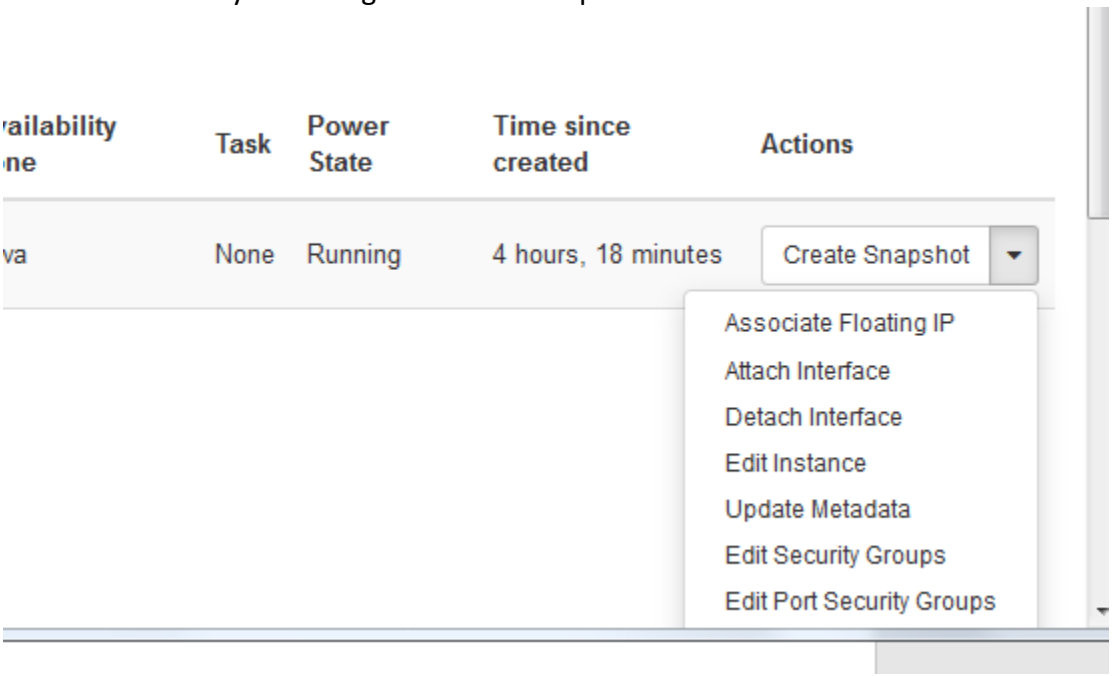


All provided images use public key authentication for accessing the VMs. A public key is injected during launch of an instance. Under **Key Pair** you need to add (or generate) the key you want to use.

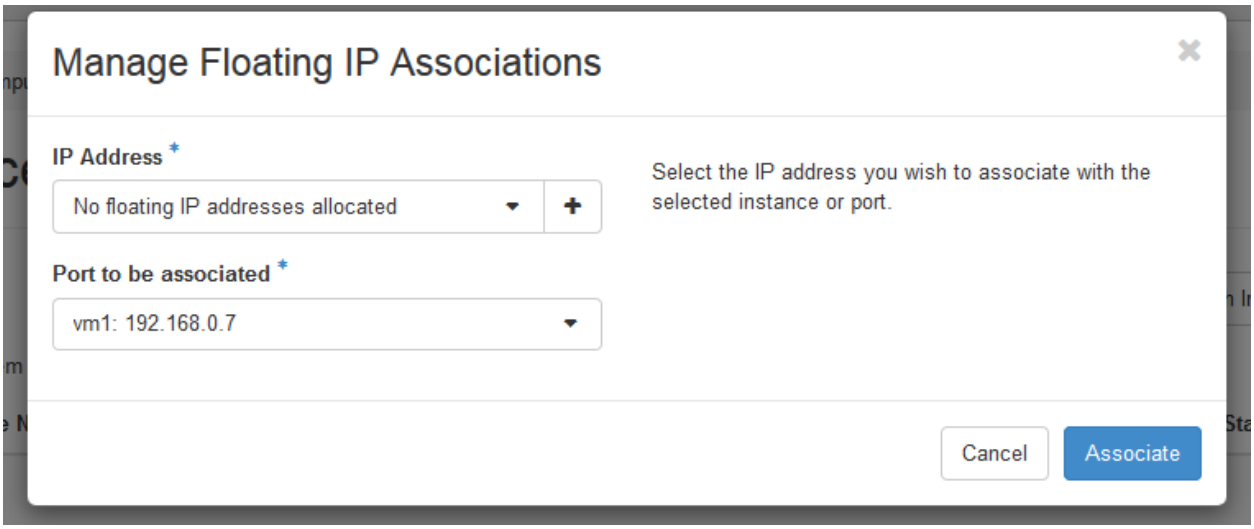
Click **Launch Instance** to start the VM. This may take a few minutes. If successful, the VM appears in your list of instances.



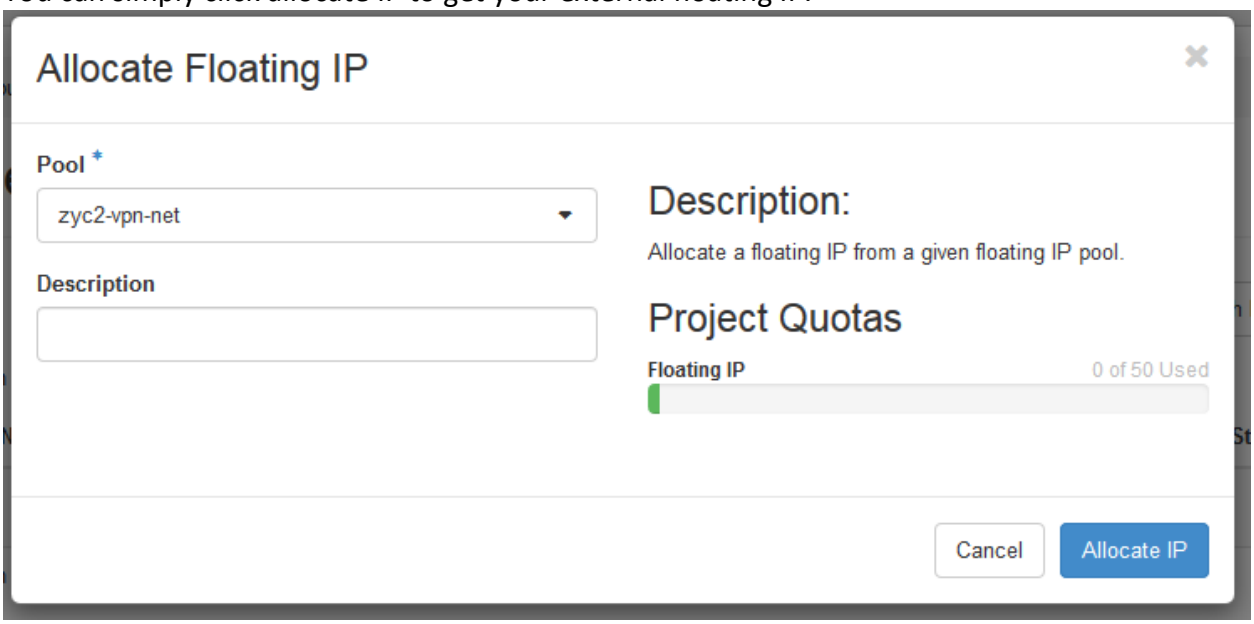
The VM receives an IP address from the private network it has been associated to. To be able to communicate via the external *zyc2-vpn-net* network you need to associate a floating IP to it. You can do that by unfolding the actions drop down menu and click on associate floating IP.



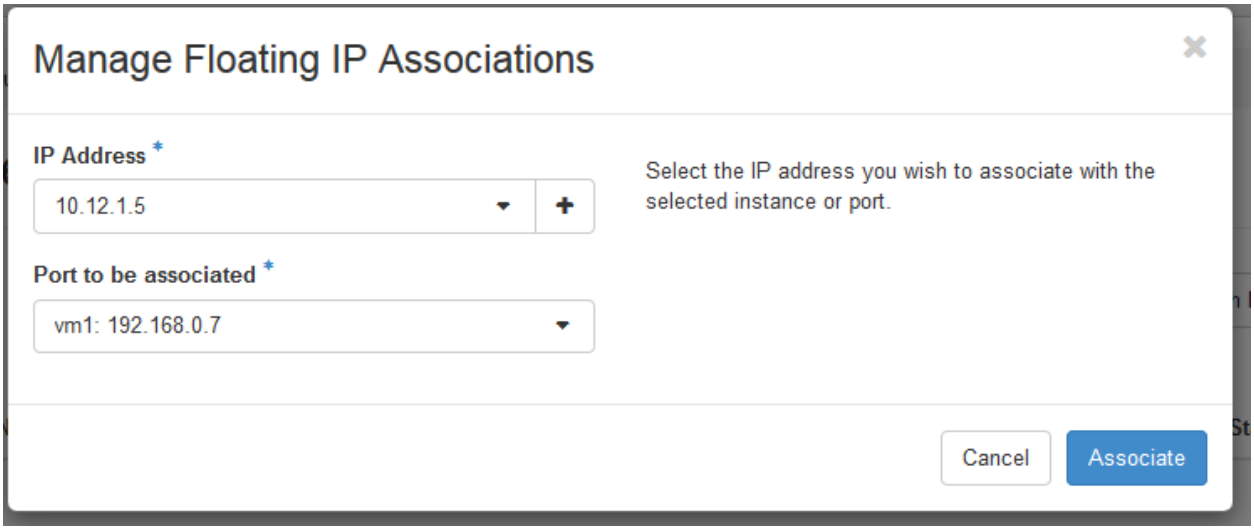
A project can receive a limited number of floating IPs from the external network. These floating IPs can then be assigned to specific VMs in the project. If you open this wizard for the first time, there is no floating IP allocated to the project. Click on the plus (+) symbol to allocate one now.



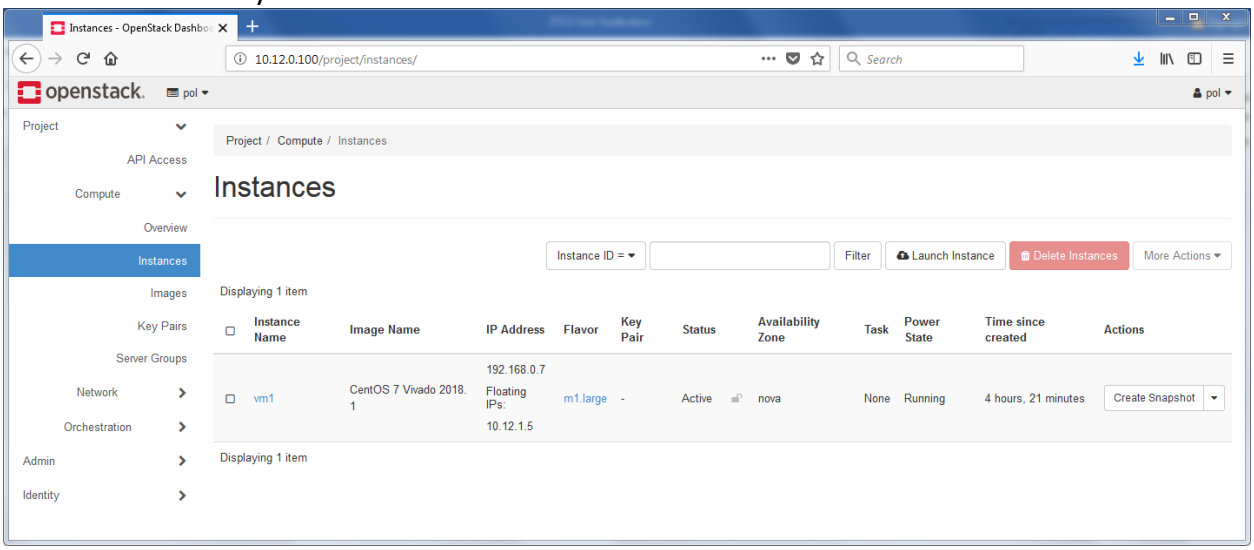
You can simply click allocate IP to get your external floating IP.



The wizard returns to the previous one and now shows the newly allocated floating IP. You can now associate this IP to a port of your VM. Since there is only one port for now this is a 1-to-1 mapping. Click associate to complete this step.



Now the floating IP will also appear in the list of your instances. This is the IP you can use to communicate with your VM.



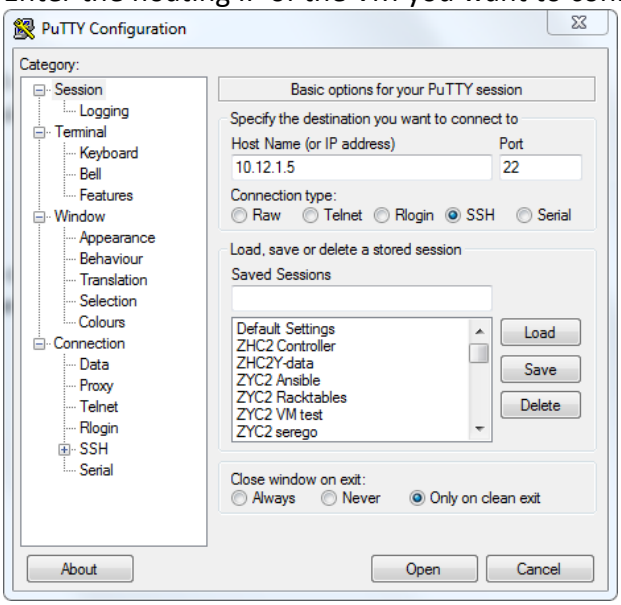
A last but important step is to adjust or review the security group rules associated with the VM. At VM creation you can select which security group to apply. In our case this was the default security group as it is the only one. Navigate to **Network → Security Groups** and select the group to review or modify. By default, the rules are very restrictive. Click on Add Rule and add at least the all ICMP and SSH rules to the security group. This will allow you to ping the VMs and use ssh for communication.

Accessing your VMs

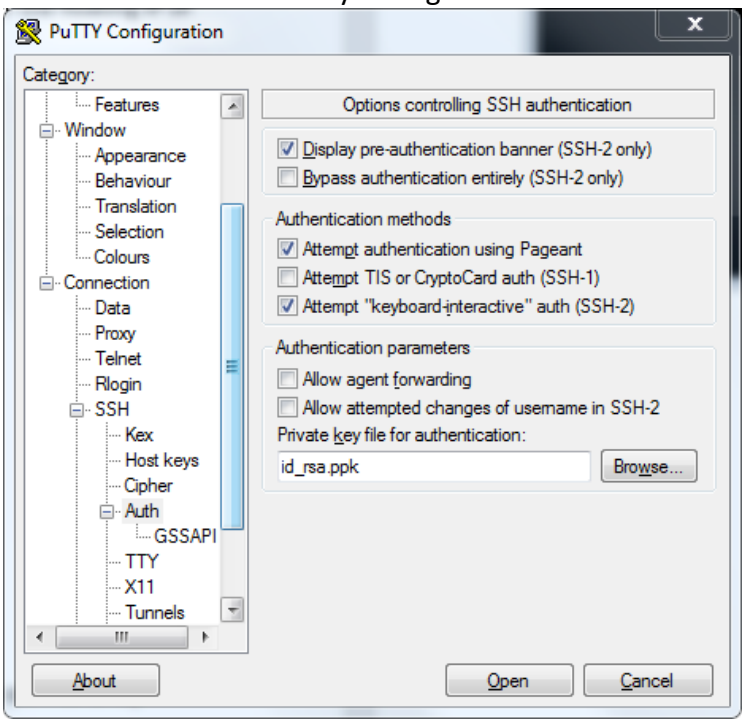
There are two ways of accessing your virtual machines. One is through the Console interface of the OpenStack dashboard. The other is by using SSH and potentially tunneling a VNC session over it.

Windows: SSH using PuTTY

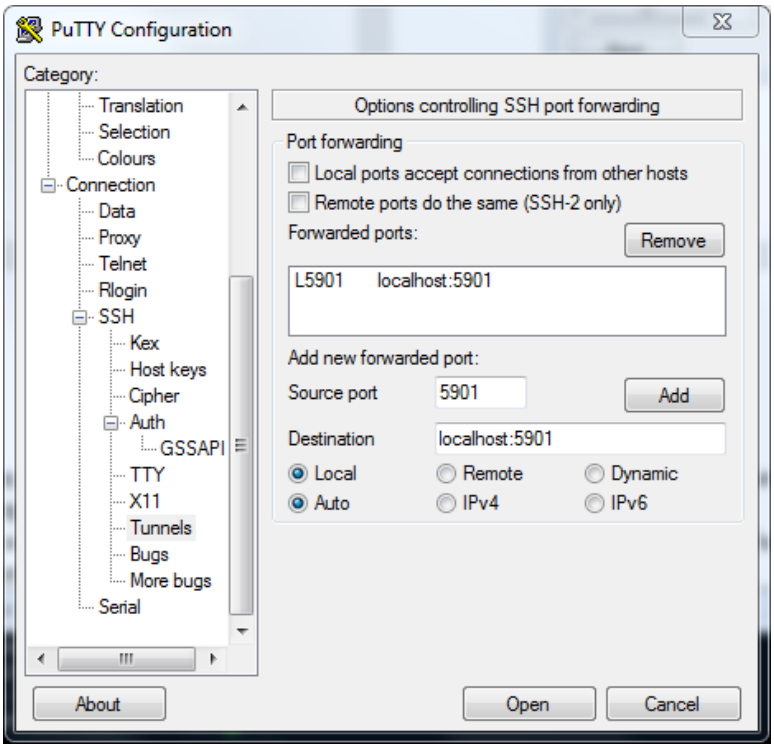
On Windows you can use the *putty* client to connect to your VM via SSH.
Link: https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html
Enter the floating IP of the VM you want to connect to.



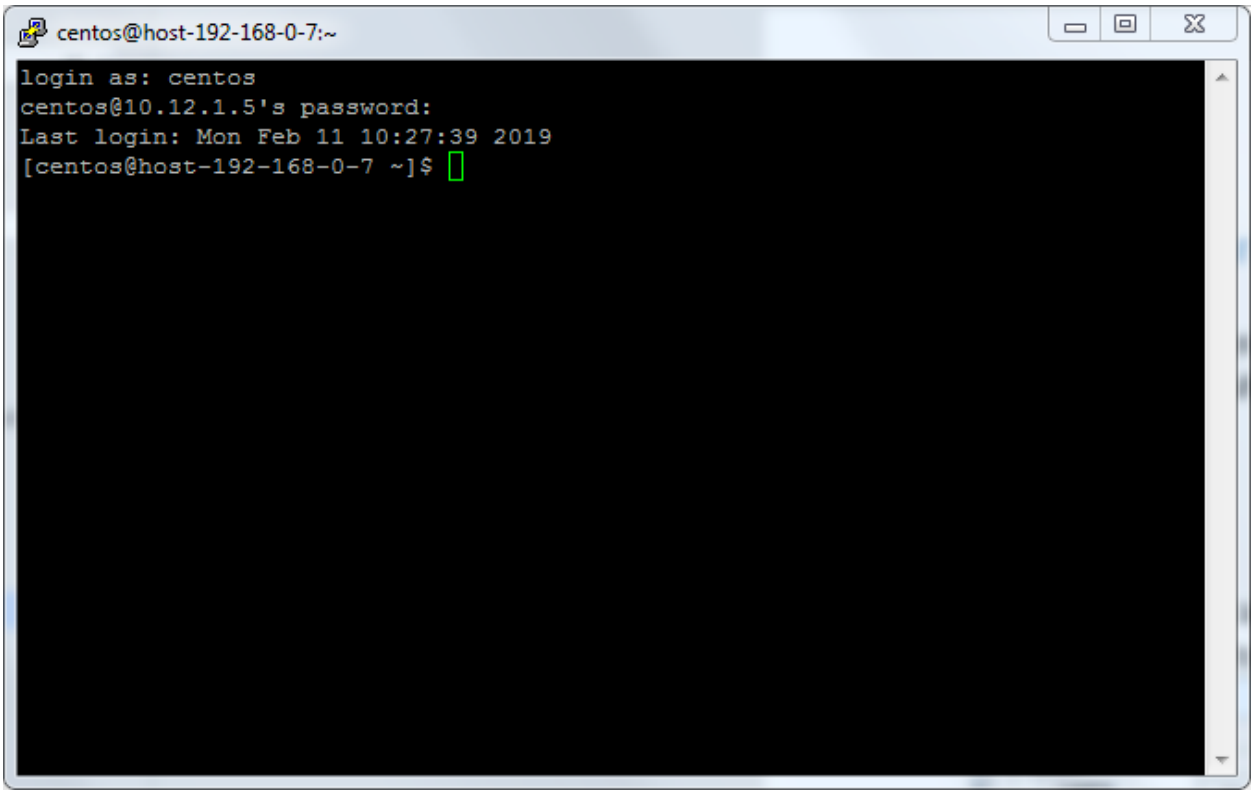
Next navigate to **Connection → SSH → Auth** to indicate and select your private key file. Putty has its own format which you might need to convert to using the *puttygen* tool.



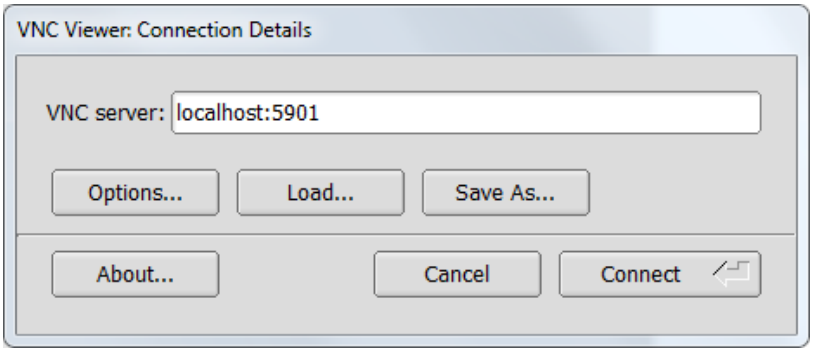
Optional: If the image provides a graphical session then you must use port forwarding to be able to access the VNC session. Go to **Connection → SSH → Tunnels** and enter *5901* as **Source port** *and localhost:5901* as **Destination** and click **Add**.



A warning may appear asking whether this is the correct host key. Click yes to permanently accept this finger print.
The terminal window will open and ask you for a username. For all CentOS images the username is set to *centos*.



To access the VNC session you can use TigerVNC
Link: https://bintray.com/tigervnc/stable/tigervnc/1.9.0



Linux: SSH

On most Linux distributions you will find ssh already installed.

To use the graphical VNC session you must enable port forwarding with a command similar as follows

```
ssh -L 5901:localhost:5901 -N -f -l centos 10.12.1.15
```

This will forward port 5901 from/to your localhost. You can then connect to your localhost:5901 using your favorite VNC client.