

Lab 2 - OpenStack Multi-Tenants Guide

This guide explains in detail how to achieve all objectives of the lab using OpenStack, including screenshots and step-by-step instructions.

Objective 1: OpenStack Overview and Basic Networking

Before starting, here are a couple of Openstack terminologies.

Nova

Nova manages and provisions compute resources. It is responsible for launching and managing virtual machines and also handles scheduling, lifecycle management, and scaling of VMs.

Swift

Swift provides scalable, redundant, and distributed object storage. It stores unstructured data like files, backups, images, and media. It uses a replication mechanism to ensure data redundancy and availability.

Cinder

Cinder provides block-level storage for virtual machines. It allows users to create and attach volumes to instances. It also allows users to take volume snapshots and backups.

Neutron

Neutron manages the networking stuff of OpenStack. It helps in creation of virtual networks, subnets, and routers. It also supports features like firewalls, load balancing, and VPNs and integrates SDN solutions.

Glance

Glance provides a service for discovering, registering, and retrieving virtual machine images. It allows users to upload custom images or use pre-configured images and integrates well with other components of OpenStack to launch instances.

Keystone

Keystone handles authentication and authorization for OpenStack services. It acts as the central identity management system, ensuring secure access control across OpenStack. It manages user accounts, roles, and access policies and provides token-based authentication for secure interactions.

Horizon

Horizon is a component in OpenStack that provides a web-based graphical user interface reducing the reliance on CLI commands. It helps in managing storage, compute and networking resources and simplifies resource provisioning and monitoring.

Users & Roles

A **user** represents an individual account that interacts with OpenStack. A user has credentials to authenticate and can be associated with one or more projects/tenants.

A **role** is a set of permissions or access rights that define what actions a user can perform within a specific project or domain. A single user can have multiple roles, and roles can differ across projects.

Hypervisors

A **hypervisor** is a layer of software, firmware, or hardware that allows multiple VMs to run on a single physical machine (virtualization) by abstracting and allocating the underlying physical hardware resources, such as CPU, memory, network resources, etc. Each VM operates as if it has its own dedicated hardware.

In OpenStack, hypervisors are managed by Nova. OpenStack supports various hypervisors such as KVM, QEMU, VMware vSphere/ESXi, Hyper-V, etc.

Flavor

In OpenStack, a flavor is a preset configuration that defines the compute, memory, and storage capacity of an instance or a VM. It allows users to specify the size and capacity of the resources required for their VMs when they are launched. For example, OpenStack provides default flavors such as m1.small or m1.large, however, admins can create custom flavors tailored to specific workloads.

Create a Network with 64 IP Addresses with DHCP enabled

```
openstack network create internal-network-1

openstack subnet create --network internal-network-1 \
    --subnet-range 192.168.10.0/24 \
    --dhcp --allocation-pool start=192.168.10.2,end=192.168.10.33 \
    internal-network-1-subnet
```

```
[stack@nvo-sneha:/home/sneha]$ openstack network show internal-network-1
+-----+-----+
| Field | Value |
+-----+-----+
| admin_state_up | UP |
| availability_zone_hints | |
| availability_zones | |
| created_at | 2025-01-17T20:00:01Z |
| description | None |
| dns_domain | |
| id | 76db7440-ad6c-4abe-a32e-8fe559a4ae04 |
| ipv4_address_scope | None |
| ipv6_address_scope | None |
| is_default | None |
| is_vlan_transparent | None |
| mtu | 1442 |
| name | internal-network-1 |
| port_security_enabled | True |
| project_id | ac46a871fd9440c2a48b6e528be4a5b5 |
| provider:network_type | geneve |
| provider:physical_network | None |
| provider:segmentation_id | 58363 |
| qos_policy_id | None |
| revision_number | 5 |
| router:external | Internal |
| segments | None |
| shared | False |
| status | ACTIVE |
| subnets | |
| tags | |
| updated_at | 2025-01-17T20:27:25Z |
+-----+-----+
```

Screenshot 1: Creating a network “internal-network-1”

Field	Value
allocation_pools	192.168.10.2-192.168.10.33
cidr	192.168.10.0/25
created_at	2025-01-17T20:28:21Z
description	
dns_nameservers	8.8.8.8
dns_publish_fixed_ip	None
enable_dhcp	True
gateway_ip	192.168.10.1
host_routes	
id	14671e5c-90ec-48cb-90b2-d9c8d4778ac8
ip_version	4
ipv6_address_mode	None
ipv6_ra_mode	None
name	internal-network-1-subnet
network_id	76db7440-ad6c-4abe-a32e-8fe559a4ae04
project_id	ac46a871fd9440c2a48b6e528be4a5b5
revision_number	0
router:external	False
segment_id	None
service_types	
subnetpool_id	
tags	
updated_at	2025-01-17T20:28:21Z

Screenshot 2: Creating a subnet “internal_network_1_subnet” with 64 usable IP addresses, with 32 allocated for DHCP leasing.

Create a Router and Connect the Network to the Public Network

```
openstack router create internal-network-1-router
```

```
openstack router set --external-gateway public internal-network-1-router
```

```
openstack router add subnet internal-network-1-router internal-network-1-subnet
```

Field	Value
admin_state_up	UP
availability_zone_hints	
availability_zones	
created_at	2025-01-17T20:18:44Z
description	
enable_default_route_bfd	False
enable_default_route_icmp	False
enable_ndp_proxy	None
external_gateway_info	[{"network_id": "ecabe00b-e09e-4d3b-bfe3-c8b2e89bc59f", "external_fixed_ips": [{"subnet_id": "149a2f72-e6bc-4ab8-a4de-f037ea28b56f", "ip_address": "172.24.4.172"}, {"subnet_id": "f048facb-3933-4a36-b426-3a2cca40ee1b", "ip_address": "2001:db8:44:1"}], "enable_snat": true}
external_gateways	[{"network_id": "ecabe00b-e09e-4d3b-bfe3-c8b2e89bc59f", "external_fixed_ips": [{"ip_address": "172.24.4.172", "subnet_id": "149a2f72-e6bc-4ab8-a4de-f037ea28b56f"}, {"ip_address": "2001:db8:44:1", "subnet_id": "f048facb-3933-4a36-b426-3a2cca40ee1b"}]]
flavor_id	None
ha	True
id	f048facb-3933-4a36-b426-3a2cca40ee1b
interfaces_info	[{"port_id": "a3b99e4x-356e-43d7-95d3-b060c4cf9db", "ip_address": "192.168.10.1", "subnet_id": "14671e5c-90ec-48cb-90b2-d9c8d4778ac8"}]
name	internal-network-1-router
project_id	ac46a871fd9440c2a48b6e528be4a5b5
revision_number	4
routes	
status	ACTIVE
tenant_id	ac46a871fd9440c2a48b6e528be4a5b5
updated_at	2025-01-17T20:33:33Z

Screenshot 3: The router’s internal interface IP is configured as the default gateway for internal-network-1-subnet, and the public network is set as the router’s external gateway.

Screenshot 4: The same configuration as Screenshot 3, displayed in the Horizon dashboard

Launch Cirros Instances

```
openstack server create --flavor m1.tiny --image cirros-0.6.3-x86_64-disk --network
internal-network-1 cirros-instance-1
```

```
openstack server create --flavor m1.tiny --image cirros-0.6.3-x86_64-disk --network
internal-network-1 cirros-instance-2
```

openstack server list						
ID	Name	Status	Networks	Image	Flavor	
77286dec-a808-4bbb-96e6-42ba4c1dc69c	cirros-instance-2	ACTIVE	internal-network-1=192.168.10.17	cirros-0.6.3-x86_64-disk	m1.tiny	
17823a4d-c4e6-4084-9127-655a98376726	cirros-instance-1	ACTIVE	internal-network-1=192.168.10.19	cirros-0.6.3-x86_64-disk	m1.tiny	

Screenshot 5: Cirros server instances dynamically allocated IP address from internal-network-1-subnet

Assign Floating IPs

Allocate a Floating IP

- Go to **Project > Network > Floating IPs**.
- Click **Allocate IP to Project** and select the public

Project / Network / Floating IPs

Floating IPs

Floating IP Address = Filter Allocate IP To Project Release Floating IPs

Displaying 6 items

<input type="checkbox"/>	IP Address	Description	DNS Name	DNS Domain	Mapped Fixed IP Address	Pool	Status	Actions
<input type="checkbox"/>	172.24.4.79				cirros-auto-4 192.168.10.31	public	Active	<input type="button" value="Disassociate"/> <input type="checkbox"/>
<input type="checkbox"/>	172.24.4.25				cirros-auto-1 192.168.10.19	public	Active	<input type="button" value="Disassociate"/> <input type="checkbox"/>
<input type="checkbox"/>	172.24.4.66				cirros-instance-2 192.168.10.17	public	Active	<input type="button" value="Disassociate"/> <input type="checkbox"/>
<input type="checkbox"/>	172.24.4.197				cirros-instance-1 192.168.10.25	public	Active	<input type="button" value="Disassociate"/> <input type="checkbox"/>
<input type="checkbox"/>	172.24.4.186				cirros-auto-2 192.168.10.21	public	Active	<input type="button" value="Disassociate"/> <input type="checkbox"/>
<input type="checkbox"/>	172.24.4.135				cirros-auto-3 192.168.10.28	public	Active	<input type="button" value="Disassociate"/> <input type="checkbox"/>

Displaying 6 items

Screenshot 6: Allocating floating IP

Associate Floating IP with an Instance:

- Click **Associate** next to the allocated floating IP.
- Select cirros-instance-1 from the dropdown.

<input type="checkbox"/>	cirros-instance-1	cirros-0.6.3-x86_64-di-sk	192.168.10.25, 172.24.4.197	m1.tiny -	Active <input type="checkbox"/>	nova	None	Running	2 days, 4 hours	<input type="button" value="Create Snapshot"/> <input type="checkbox"/>
<input type="checkbox"/>	cirros-instance-2	cirros-0.6.3-x86_64-di-sk	192.168.10.17, 172.24.4.66	m1.tiny -	Active <input type="checkbox"/>	nova	None	Running	3 days, 8 hours	<input type="button" value="Create Snapshot"/> <input type="checkbox"/>

Screenshot 7: Floating IP assigned to the instances

Objective 2: Auto-Scaling with Python

Python Script for Monitoring CPU Utilization

[Link to the code]

I am using regular ssh, netmiko to extract CPU Utilization. Here's how the output looks like:

```
[stack@nvo-sneha:~/git/Network-Virtualization-and-Orchestration$ python3 cpu_util.py
  cirros-instance-1 CPU Utilization: 0%
  cirros-instance-2 CPU Utilization: 10%
```

Screenshot 6: Output of cpu_util.py script that extracts CPU usage of the cirros instances

I am utilizing the OpenStack Python module to create instances. To increase CPU load, I use the yes command, which continuously outputs the letter 'y' to the command line. My code also assigns a floating IP address to each Cirros instance, enabling internet accessibility and monitoring. A CPU utilization threshold of 30% has been set, and if any instance exceeds this threshold, the code automatically spawns a new Cirros instance. Once the total number of instances reaches 4, the code stops creating additional instances and exits as seen in the below screenshot.

```
stack@nvo-sneha:/git/Network-Virtualization-and-Orchestration$ python3 cpu_util.py
2025-01-20 02:33:42.898 | INFO    | __main__:main:79 - Starting instance CPU utilization monitoring...
2025-01-20 02:33:48.637 | WARNING | __main__:monitor_instances:59 - CPU utilization for cirros-instance-1 exceeds 30%. Creating a new instance...
2025-01-20 02:33:48.637 | INFO    | create_instance:create_cirros_instance:20 - Creating new instance: cirros-auto-1
2025-01-20 02:34:07.642 | SUCCESS | create_instance:create_cirros_instance:44 - Instance cirros-auto-1 created successfully!
2025-01-20 02:34:10.918 | SUCCESS | create_instance:create_cirros_instance:48 - Floating IP 172.24.4.25 associated with instance cirros-auto-1.
2025-01-20 02:34:10.919 | SUCCESS | create_instance:create_cirros_instance:52 - sshInfo.csv updated with instance cirros-auto-1.

2025-01-20 02:34:16.893 | WARNING | __main__:monitor_instances:59 - CPU utilization for cirros-instance-1 exceeds 30%. Creating a new instance...
2025-01-20 02:34:16.893 | INFO    | create_instance:create_cirros_instance:20 - Creating new instance: cirros-auto-2
2025-01-20 02:34:40.970 | SUCCESS | create_instance:create_cirros_instance:44 - Instance cirros-auto-2 created successfully!
2025-01-20 02:34:44.564 | SUCCESS | create_instance:create_cirros_instance:48 - Floating IP 172.24.4.186 associated with instance cirros-auto-2.
2025-01-20 02:34:44.564 | SUCCESS | create_instance:create_cirros_instance:52 - sshInfo.csv updated with instance cirros-auto-2.

2025-01-20 02:34:51.012 | WARNING | __main__:monitor_instances:59 - CPU utilization for cirros-instance-1 exceeds 30%. Creating a new instance...
2025-01-20 02:34:51.012 | INFO    | create_instance:create_cirros_instance:20 - Creating new instance: cirros-auto-3
2025-01-20 02:35:10.309 | SUCCESS | create_instance:create_cirros_instance:44 - Instance cirros-auto-3 created successfully!

2025-01-20 02:35:12.520 | SUCCESS | create_instance:create_cirros_instance:48 - Floating IP 172.24.4.135 associated with instance cirros-auto-3.
2025-01-20 02:35:12.520 | SUCCESS | create_instance:create_cirros_instance:52 - sshInfo.csv updated with instance cirros-auto-3.

2025-01-20 02:35:19.380 | WARNING | __main__:monitor_instances:59 - CPU utilization for cirros-instance-1 exceeds 30%. Creating a new instance...
2025-01-20 02:35:19.380 | INFO    | create_instance:create_cirros_instance:20 - Creating new instance: cirros-auto-4
2025-01-20 02:35:42.677 | SUCCESS | create_instance:create_cirros_instance:44 - Instance cirros-auto-4 created successfully!
2025-01-20 02:35:45.103 | SUCCESS | create_instance:create_cirros_instance:48 - Floating IP 172.24.4.79 associated with instance cirros-auto-4.
2025-01-20 02:35:45.103 | SUCCESS | create_instance:create_cirros_instance:52 - sshInfo.csv updated with instance cirros-auto-4.
2025-01-20 02:36:52.510 | DEBUG   | __main__:monitor_instances:63 - CPU utilization threshold breached. 4 instances have already been created. No additional instances will be spun up.
```

Screenshot 8: Output of the cpu utilization code running and creating new instances on demand

<input type="checkbox"/>	Instance Name	Image Name	IP Address	Flavor	Key Pair	Status	Availability Zone	Task	Power State	Age	Actions
<input type="checkbox"/>	cirros-auto-4	cirros-0.6.3-x86_64-disk	192.168.10.31, 172.24.4.79	m1.tiny	-	Active	<input type="checkbox"/> nova	None	Running	1 minute	<button>Create Snapshot</button>
<input type="checkbox"/>	cirros-auto-3	cirros-0.6.3-x86_64-disk	192.168.10.28, 172.24.4.135	m1.tiny	-	Active	<input type="checkbox"/> nova	None	Running	2 minutes	<button>Create Snapshot</button>
<input type="checkbox"/>	cirros-auto-2	cirros-0.6.3-x86_64-disk	192.168.10.21, 172.24.4.186	m1.tiny	-	Active	<input type="checkbox"/> nova	None	Running	2 minutes	<button>Create Snapshot</button>
<input type="checkbox"/>	cirros-auto-1	cirros-0.6.3-x86_64-disk	192.168.10.19, 172.24.4.25	m1.tiny	-	Active	<input type="checkbox"/> nova	None	Running	3 minutes	<button>Create Snapshot</button>
<input type="checkbox"/>	cirros-instance-1	cirros-0.6.3-x86_64-disk	192.168.10.25, 172.24.4.197	m1.tiny	-	Active	<input type="checkbox"/> nova	None	Running	1 day, 2 hours	<button>Create Snapshot</button>
<input type="checkbox"/>	cirros-instance-2	cirros-0.6.3-x86_64-disk	192.168.10.17, 172.24.4.66	m1.tiny	-	Active	<input type="checkbox"/> nova	None	Running	2 days, 5 hours	<button>Create Snapshot</button>

Screenshot 9: Horizon dashboard displaying the instances created by the code

Objective 3: Multi-Tenants

Create a Project and User:

- **Using Horizon:**
 - Go to **Identity > Projects** and click **Create Project**
 - Enter:
 - **Project Name:** lab2
 - **Description:** Optional
 - **Enabled:** Checked
 - Click **Create Project**.
 - Navigate to **Identity > Users** and click **Create User**.
 - Enter:
 - **Username:** lab2_admin
 - **Email:** Optional
 - **Password:** Set your password
 - **Primary Project:** lab2
 - **Role:** admin
 - Click **Create User**:



Screenshot 10: Project lab2 created



Screenshot 11: User lab2_admin

- **Create a Flavor:**
 - Navigate to **Admin > Flavors** and click **Create Flavor**.
 - Enter:
 - **Flavor Name:** ngn.tiny
 - **vCPUs:** 1
 - **RAM:** 128 MB
 - **Root Disk:** 1 GB

- **Ephemeral Disk:** 1 GB
 - **Swap Disk:** 1 GB
- Click **Create Flavor**.
- **Upload an Image:**
 - Go to **Admin > Images** and click **Create Image**.
 - Enter:
 - **Image Name:** tinycore
 - **Image Source:** [link](#)
 - **Format:** ISO
 - **Visibility:** Public
 - Click **Create Image**

<input type="checkbox"/>	ngn.tiny	1	128MB	1GB	1GB	1024MB	1.0	2d9a7884-a00c-4ace-950f-b378fc15e260	Yes	No	Update Metadata	<input type="checkbox"/>
--------------------------	----------	---	-------	-----	-----	--------	-----	--------------------------------------	-----	----	---------------------------------	--------------------------

Screenshot 12: Created ngn.tiny flavor

Setup Virtual Networks

1. Create VN-A/VN-B

```
openstack network create VN-A

openstack subnet create --network VN-A --subnet-range 192.168.100.0/24 VN-A-Subnet

openstack network create VN-B

openstack subnet create --network VN-B --subnet-range 192.168.200.0/24 VN-B-Subnet
```

stack@nvo-sneha:~/devstack\$ openstack network create VN-A	stack@nvo-sneha:~\$ openstack subnet create \ --network VN-A \ --subnet-range 192.168.100.0/24 \ --ip-version 4 \ VN-A-Subnet
+-----+ Field Value +-----+ admin_state_up UP availability_zone_hints availability_zones created_at 2025-01-20T20:36:49Z description dns_domain None id e68fd3a8-56bb-42b3-a5c6-f1faa3ef7d30 ipv4_address_scope None ipv6_address_scope None is_default None is_vlan_transparent None mtu 1442 name VN-A port_security_enabled True project_id ac46a871fd9440c2a48b6e528be4a5b5 provider:network_type geneve provider:physical_network None provider:segmentation_id 5494 qos_policy_id None revision_number 1 router:external Internal segments None shared False status ACTIVE subnets tags updated_at 2025-01-20T20:36:49Z +-----+	+-----+ Field Value +-----+ allocation_pools 192.168.100.2-192.168.100.254 cidr 192.168.100.0/24 created_at 2025-01-20T20:37:43Z description dns_nameservers dns_publish_fixed_ip None enable_dhcp True gateway_ip 192.168.100.1 host_routes id 035dd38f-9614-4382-900c-bfe62861ea64 ip_version 4 ipv6_address_mode None ipv6_ra_mode None name VN-A-Subnet network_id e68fd3a8-56bb-42b3-a5c6-f1faa3ef7d30 project_id ac46a871fd9440c2a48b6e528be4a5b5 revision_number 0 router:external False segment_id None service_types subnetpool_id None tags updated_at 2025-01-20T20:37:43Z +-----+
stack@nvo-sneha:~\$ openstack network create VN-B	stack@nvo-sneha:~\$ openstack subnet create \ --network VN-B \ --subnet-range 192.168.200.0/24 \ --ip-version 4 \ VN-B-Subnet
+-----+ Field Value +-----+ admin_state_up UP availability_zone_hints availability_zones created_at 2025-01-20T20:37:54Z description dns_domain None id f00ade63-75b6-43f0-a497-a0fd0253f680 ipv4_address_scope None ipv6_address_scope None is_default None is_vlan_transparent None mtu 1442 name VN-B port_security_enabled True project_id ac46a871fd9440c2a48b6e528be4a5b5 provider:network_type geneve provider:physical_network None provider:segmentation_id 65194 qos_policy_id None revision_number 1 router:external Internal segments None shared False status ACTIVE subnets tags updated_at 2025-01-20T20:37:54Z +-----+	+-----+ Field Value +-----+ allocation_pools 192.168.200.2-192.168.200.254 cidr 192.168.200.0/24 created_at 2025-01-20T20:39:59Z description dns_nameservers dns_publish_fixed_ip None enable_dhcp True gateway_ip 192.168.200.1 host_routes id 03de1fd5-ceee-472a-b7c6-def4bcf5b106 ip_version 4 ipv6_address_mode None ipv6_ra_mode None name VN-B-Subnet network_id f00ade63-75b6-43f0-a497-a0fd0253f680 project_id ac46a871fd9440c2a48b6e528be4a5b5 revision_number 0 router:external False segment_id None service_types subnetpool_id None tags updated_at 2025-01-20T20:39:59Z +-----+

Screenshot 13: Creating the VN-{A,B} networks and subnets

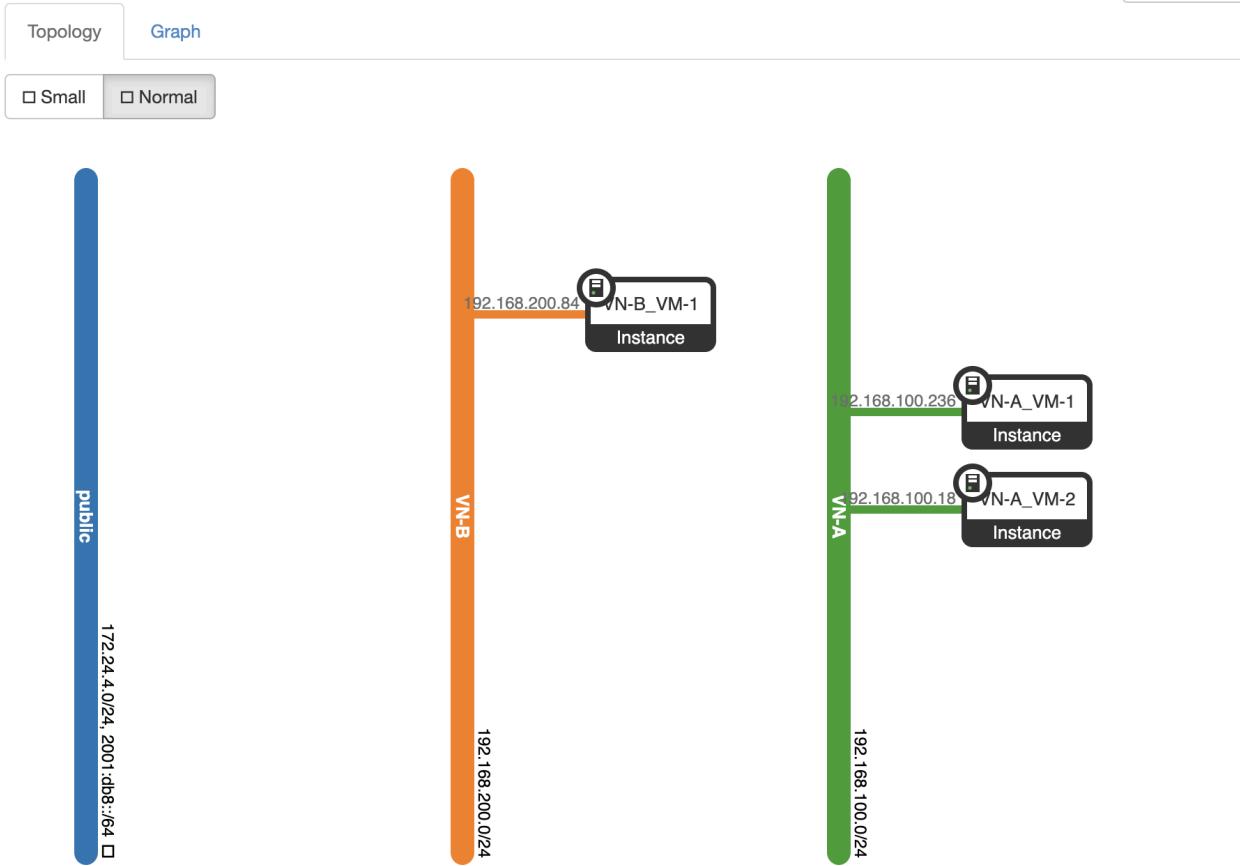
Launch VM Instances

Launch VN-A_VM-1 and VN-A_VM-2 into VN-A

```
openstack server create --flavor ngn.tiny --image tinycore --network VN-A VN-A_VM-1  
  
openstack server create --flavor ngn.tiny --image tinycore --network VN-A VN-A_VM-2
```

Launch VN-B_VM-1 into VN-B

```
openstack server create --flavor ngn.tiny --image tinycore --network VN-B VN-B_VM-1
```



Screenshot 14: Network topology after spinning up 2 VMs in VN-A and 1 VM in VN-B

Ping testing

```
$ ping 192.168.100.18
PING 192.168.100.18 (192.168.100.18) 56(84) bytes of data.
64 bytes from 192.168.100.18: icmp_seq=1 ttl=64 time=0.058 ms
64 bytes from 192.168.100.18: icmp_seq=2 ttl=64 time=0.048 ms
64 bytes from 192.168.100.18: icmp_seq=3 ttl=64 time=0.051 ms
64 bytes from 192.168.100.18: icmp_seq=4 ttl=64 time=0.052 ms
64 bytes from 192.168.100.18: icmp_seq=5 ttl=64 time=0.050 ms
^C
--- 192.168.100.18 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4088ms
rtt min/avg/max/mdev = 0.048/0.051/0.058/0.003 ms
```

Screenshot 15: Ping within the same Virtual Networks

```
$ ping 192.168.200.84
PING 192.168.200.84 (192.168.200.84) 56(84) bytes of data.
From 192.168.100.18 icmp_seq=1 Destination Host Unreachable
From 192.168.100.18 icmp_seq=2 Destination Host Unreachable
From 192.168.100.18 icmp_seq=3 Destination Host Unreachable
^C
--- 192.168.200.84 ping statistics ---
5 packets transmitted, 0 received, +3 errors, 100% packet loss, time 4082ms
```

Screenshot 16: Pinging from VN-A to VN-B

Floating IPs

<input type="checkbox"/>	IP Address	Description	DNS Name	DNS Domain	Mapped Fixed IP Address	Pool	Status	Actions
<input type="checkbox"/>	172.24.4.61				VN-A_VM-1 192.168.100.236	public	Active	<button>Disassociate</button> <input type="checkbox"/>
<input type="checkbox"/>	172.24.4.106				VN-A_VM-2 192.168.100.18	public	Active	<button>Disassociate</button> <input type="checkbox"/>
<input type="checkbox"/>	172.24.4.189				VN-B_VM-1 192.168.200.84	public	Active	<button>Disassociate</button> <input type="checkbox"/>

Screenshot 17: Assigned floating IP addresses on all the VMs

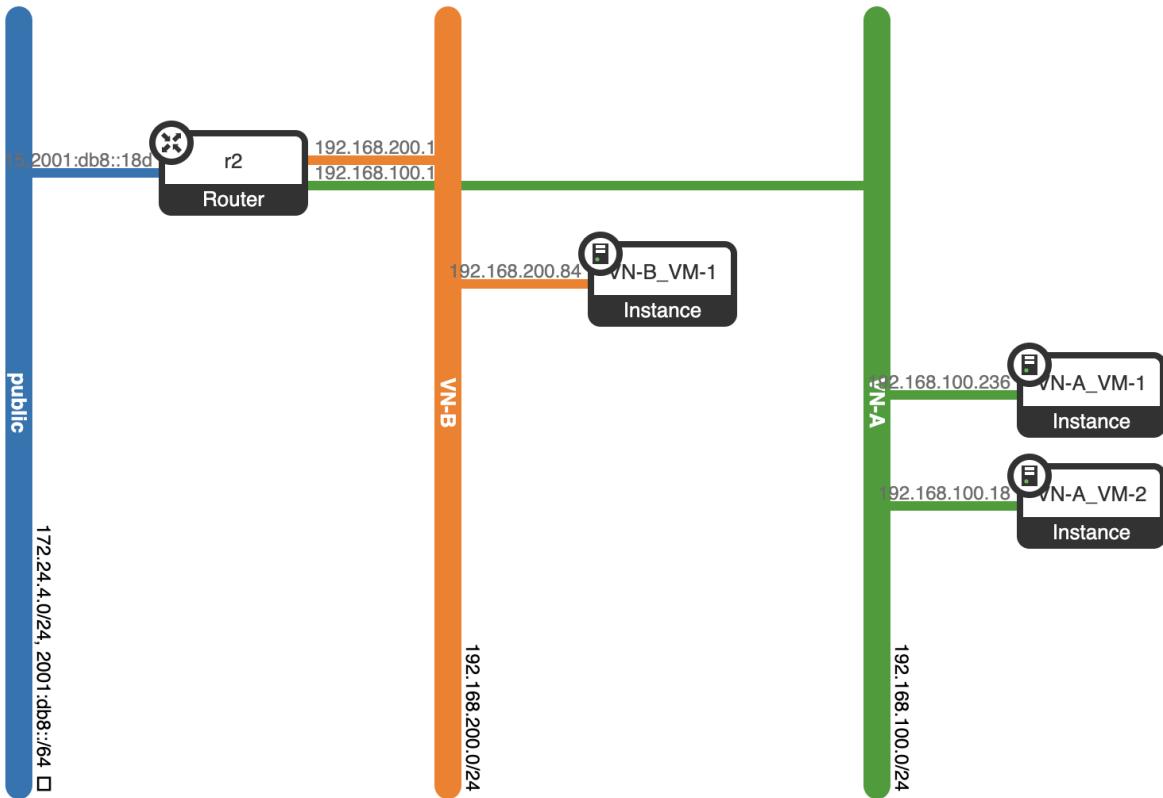
```
$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=114 time=9.01 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=114 time=5.41 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=114 time=2.67 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=114 time=2.75 ms
^C
--- 8.8.8.8 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3003ms
rtt min/avg/max/mdev = 2.672/4.959/9.006/2.582 ms
```

Screenshot 18: Pinging the internet

Objective 4: Network Policy Management

In the previous steps, we achieved internet connectivity between the VMs. I added a router that connects to the public network as well as the virtual networks to enable routing. Post adding the router, I was able to ping from VN-A to VN-B.

Here's the network topology.



```
$ ping 192.168.200.84
PING 192.168.200.84 (192.168.200.84) 56(84) bytes of data.
64 bytes from 192.168.200.84: icmp_seq=1 ttl=63 time=3.86 ms
64 bytes from 192.168.200.84: icmp_seq=2 ttl=63 time=1.15 ms
64 bytes from 192.168.200.84: icmp_seq=3 ttl=63 time=0.354 ms
64 bytes from 192.168.200.84: icmp_seq=4 ttl=63 time=0.294 ms
^C
--- 192.168.200.84 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 0.294/1.415/3.862/1.452 ms
```

Screenshot 19: Inter-VN communication from VN-A-VM_1 enabled after adding a router

Configure Network Policies

openstack security group list --project 10f88bc267634e47aa99db58ce94e63					
ID	Name	Description	Project	Tags	
0f3c3a39-7509-44bb-8eb0-0766b5685fd7	sg-vnb-vm1	Security group for VN-B_VM-1	10f88bc267634e47aa99db58ce94e63	[]	
28843bb0-aa5d-4653-94ff-e41d20f3966f	default	Default security group	10f88bc267634e47aa99db58ce94e63	[]	
4b13ce87-bc08-44c2-a9a3-6b4e365ce202	sg-vna-vm1	Security group for VN-A_VM-1	10f88bc267634e47aa99db58ce94e63	[]	
cab647b4-d3e4-496f-b7fb-b326a7fae053	sg-vna-vm2	Security group for VN-A_VM-2	10f88bc267634e47aa99db58ce94e63	[]	

For each of the security groups, I created a custom rule.

Case 1: VN-A_VM-1

Manage Security Group Rules: sg-vna-vm1 (4b13ce87-bc08-44c2-a9a3-6b4e365ce202)

<input type="checkbox"/>	Direction	Ether Type	IP Protocol	Port Range	Remote IP Prefix	Remote Security Group	Description	Actions
<input type="checkbox"/>	Egress	IPv4	ICMP	Any	192.168.100.18/32	-	-	<button>Delete Rule</button>
<input type="checkbox"/>	Egress	IPv4	ICMP	Any	192.168.200.84/32	-	-	<button>Delete Rule</button>
<input type="checkbox"/>	Ingress	IPv4	ICMP	Any	192.168.200.84/32	-	-	<button>Delete Rule</button>
<input type="checkbox"/>	Ingress	IPv4	ICMP	Any	192.168.100.18/32	-	-	<button>Delete Rule</button>

Displaying 4 items

Screenshot 20: Security group for VN-A_VM-1

1. Can ping VN-B_VM-1

VN-A_VM-1

```
Connected to QEMU (instance-00000026)
$ ping 192.168.200.84
PING 192.168.200.84 (192.168.200.84) 56(84) bytes of data.
64 bytes from 192.168.200.84: icmp_seq=1 ttl=63 time=6.34 ms
64 bytes from 192.168.200.84: icmp_seq=2 ttl=63 time=1.16 ms
64 bytes from 192.168.200.84: icmp_seq=3 ttl=63 time=0.344 ms
64 bytes from 192.168.200.84: icmp_seq=4 ttl=63 time=0.326 ms
^C
--- 192.168.200.84 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3030ms
rtt min/avg/max/mdev = 0.326/2.041/6.338/2.503 ms
$
```

2. Cannot ping the internet

```
Project / Compute / Instances / VN-A_VM-1

Connected to QEMU (instance-00000026)
$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
^C
--- 8.8.8.8 ping statistics ---
5 packets transmitted, 0 received, 100% packet loss, time 4085ms
$ -
```

Case 2: VN-A_VM-2

Manage Security Group Rules: sg-vna-vm2 (cab647b4-d3e4-496f-b7fb-b326a7fae053)

Displaying 2 items								<input type="button" value="Add Rule"/>	<input type="button" value="Delete Rules"/>
Direction	Ether Type	IP Protocol	Port Range	Remote IP Prefix	Remote Security Group	Description	Actions		
Egress	IPv4	ICMP	Any	172.24.4.61/32	-	-	<input type="button" value="Delete Rule"/>		
Ingress	IPv4	ICMP	Any	172.24.4.61/32	-	-	<input type="button" value="Delete Rule"/>		

Displaying 2 items

Screenshot 21: Security group for VN-A_VM-2

1. Cannot ping VN-B_VM-1

VN-A_VM-2

Overview Interfaces Log **Console** Action Log

Instance Console

If console is not responding to keyboard input: click the grey status bar below. [Click here to show only console](#)
To exit the fullscreen mode, click the browser's back button.

```
Connected to QEMU (instance-00000025)
$ ping 192.168.200.84
PING 192.168.200.84 (192.168.200.84) 56(84) bytes of data.
^C
--- 192.168.200.84 ping statistics ---
4 packets transmitted, 0 received, 100% packet loss, time 3061ms
$ -
```

2. Cannot ping the internet

VN-A_VM-2

Overview Interfaces Log **Action Log**

Instance Console

If console is not responding to keyboard input: click the grey status bar below. [Click here to show only console](#)
To exit the fullscreen mode, click the browser's back button.

```
Connected to QEMU (instance-00000025)
$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
^C
--- 8.8.8.8 ping statistics ---
4 packets transmitted, 0 received, 100% packet loss, time 3073ms
$ -
```

Case 3: VN-B_VM-1

Manage Security Group Rules: sg-vnb-vm1 (0f3c3a39-7509-44bb-8eb0-0766b5685fd7)

Displaying 4 items								<input type="button" value="Add Rule"/>	<input type="button" value="Delete Rules"/>
□	Direction	Ether Type	IP Protocol	Port Range	Remote IP Prefix	Remote Security Group	Description	Actions	
□	Egress	IPv4	Any	Any	0.0.0.0/0	-	-	<input type="button" value="Delete Rule"/>	
□	Egress	IPv4	ICMP	Any	192.168.100.236/32	-	-	<input type="button" value="Delete Rule"/>	
□	Egress	IPv6	Any	Any	::/0	-	-	<input type="button" value="Delete Rule"/>	
□	Ingress	IPv4	ICMP	Any	192.168.100.236/32	-	-	<input type="button" value="Delete Rule"/>	

Displaying 4 items

Screenshot 22: Security group for VN-B_VM-1

1. Can ping VN-A_VM-1

VN-B_VM-1

Overview Interfaces Log **Console** Action Log

Instance Console

If console is not responding to keyboard input: click the grey status bar below. [Click here to show only console](#)
To exit the fullscreen mode, click the browser's back button.

```
Connected to QEMU (instance-00000027)
$ ping 192.168.100.236
PING 192.168.100.236 (192.168.100.236) 56(84) bytes of data.
64 bytes from 192.168.100.236: icmp_seq=1 ttl=63 time=1.38 ms
64 bytes from 192.168.100.236: icmp_seq=2 ttl=63 time=1.07 ms
64 bytes from 192.168.100.236: icmp_seq=3 ttl=63 time=0.289 ms
64 bytes from 192.168.100.236: icmp_seq=4 ttl=63 time=0.366 ms
64 bytes from 192.168.100.236: icmp_seq=5 ttl=63 time=0.294 ms
^C
--- 192.168.100.236 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4037ms
rtt min/avg/max/mdev = 0.289/1.278/4.377/1.577 ms
$
```

2. Cannot ping VN-A_VM-2

VN-B_VM-1

Overview Interfaces **Log** Console Action Log

Instance Console

If console is not responding to keyboard input: click the grey status bar below. [Click here to show only console](#)
To exit the fullscreen mode, click the browser's back button.

```
Connected to QEMU (instance-00000027)
$ ping 192.168.100.18
PING 192.168.100.18 (192.168.100.18) 56(84) bytes of data.
^C
--- 192.168.100.18 ping statistics ---
5 packets transmitted, 0 received, 100% packet loss, time 4080ms
$
```

3. Can ping the internet

VN-B_VM-1

Overview Interfaces Log **Console** Action Log

Instance Console

If console is not responding to keyboard input: click the grey status bar below. [Click here to show only console](#)
To exit the fullscreen mode, click the browser's back button.

```
Connected to QEMU (instance-00000027)

$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=114 time=7.06 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=114 time=3.13 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=114 time=2.37 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=114 time=4.23 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=114 time=2.56 ms
^C
--- 8.8.8.8 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4006ms
rtt min/avg/max/mdev = 2.368/3.869/7.062/1.723 ms
$ -
```