Lab CC-IAAS - Using OpenStack

Introduction and Prerequisites

This laboratory is to:

- Explore various features of OpenStack IaaS cloud framework.
- Learn how to use the Horizon dashboard to interact with various services such as Nova, Keystone, Cinder, Neutron and Glance.
- Learn how to create and manage virtual machines in OpenStack and access it.

In this lab you will explore the OpenStack Horizon dashboard and the On-Demand and Self-Service capabilities offered through it. You will create virtual machines, connect them over virtual network and use OpenStack Volumes.

The following resources and tools are required for this laboratory session:

- Any modern web browser.
- Any modern SSH client application:
  - Windows users[1]: https://www.ssh.com/ssh/putty/windows/puttygen
    - Putty: https://www.ssh.com/ssh/putty/download
  - Mac / Unix / Linux users: `ssh` and `ssh-keygen` commands are needed
    - OpenSSL packages provide these
- OpenStack Horizon dashboard: https://ned.cloudlab.zhaw.ch
- OpenStack account details: please contact the lab assistant in case you already have not received your access credentials.

Time

The entire session will take 90 minutes.

Task 1: Setup and Basics

Subtask 1.1 Keypairs

Create your own public/private key-pair:

- Windows users - use puttygen tool to generate an SSH key-pair (If possible choose RSA as your key-pair type)
  - https://www.ssh.com/ssh/putty/windows/puttygen#sec-Creating-a-new-key-pair-for-authentication
  - https://devops.ionos.com/tutorials/use-ssh-keys-with-putty-on-windows/
- Linux/Unix/Mac users
  - Use `ssh-keygen` system utility in a terminal window to generate a key-pair
    - Use RSA as key pair type
  - By default, the generated key pair can be found within your `$HOME/.ssh/` directory.
- The generated key pair has a public part (file typically ending with `.pub` extension) and a private part which is meant never to be shared with anyone.

[1] Note that SSH is also available on Windows 10 through Windows Subsystem for Linux

Register your public key with OpenStack

- Use your credentials to login to the OpenStack Horizon dashboard.
- Register your created key pair with your account.
    - Use Compute → Key Pairs → Import Key Pair

Subtask 1.2 Security Groups

Every OpenStack project comes with a Default security rule that allows all traffic from inside a virtual machine to go out, but blocks all external traffic from coming in.

The goal of this subtask is to permit SSH and HTTP traffic to reach the VM.

- Use Network → Security Groups → Create Security Group.
- Add rules to allow SSH traffic and save.

Subtask 1.3 Understanding Quotas

- Go to Compute → Overview page,
    - Analyze what you see and discuss what quota is allocated for your project and how many VMs can you realistically create with such quota limitations.

Subtask 1.4 Create your VM

- Create the VM (Compute → Instances → Launch Instance) with a name of your choice, from a basic VM image (Boot Source → Image) named '`2019-08_Ubuntu_Bionic_with_Docker`'.
- To this VM add the '`m1.small`' Flavor.
- Attach to the '`default_internal`' network. You do not need to create a router as there is a hidden router associated with the '`default_internal`' network and is provided 'as a Service'.
- Make sure the key name and security groups are those you defined earlier.
- Launch the VM.
- Once the VM is created, Associate a Floating IP from the '`public1`' network.

Subtask 1.5 Basic VM Management

- Look at the boot log of your VM in the horizon dashboard. What is the IP address assigned to the VM. What is the device name that the IP address is associated with? Is the assigned IP the same as the Floating IP? If not, why?
- SSH to the VM[2] and create a file named '`ccp1-lab.txt`'.
- Try suspending and resuming the VM.
    - Are these actions are relevant in a real world scenario? Where?
    - Upon resuming, make sure the file "`ccp1-lab.txt`" is still in the VM.
- Do a VM rebuild (again from the Actions drop down menu).
    - Is your created file "`ccp1-lab.txt`" still available in your VM?
- Create a VM Snapshot
    - Ensure "`ccp1-lab.txt`" is present, if not create it.
    - Give the snapshot a meaningful name.

[2] The username is '`ubuntu`' e.g. ssh -i $PRIV_KEY ubuntu@$IP_ADDRESS

    - Once completed, the Snapshot should be available under Compute → Images tab. Verify that it exists.
- Terminate the VM and create a new VM using the Snapshot that you previously created
    - Is the file that was created (‘`ccp1-lab.txt`’) still available once the VM is SSH’ed into?
    - Discuss the importance of snapshots while working with VMs.

Task 2: Advanced Management

In this task we will create all our own infrastructure, from networking, storage and compute.

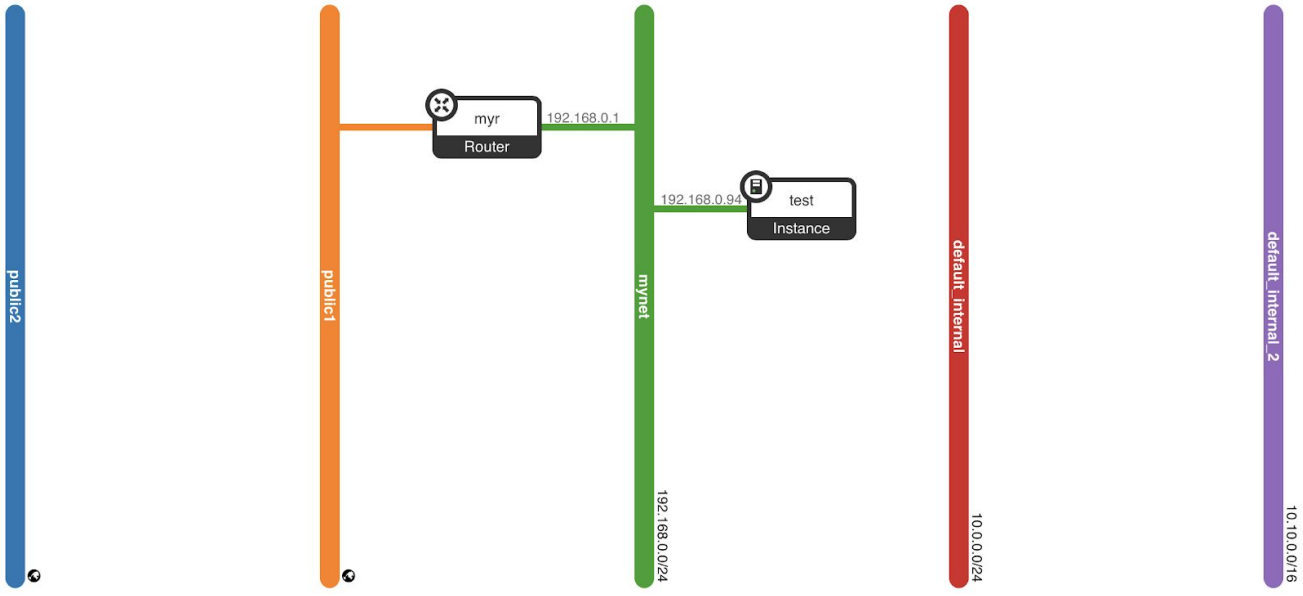
Subtask 2.1 Networking

- Create a Network & Subnet.
- Create a Router.
- Attach the router to a public network (‘`public1`’) and also attach it to the Network you just created.

Subtask 2.2 Virtual Machine

- As in Subtask 1.3, create a VM, with the same specification, but that is attached to the **new** network you have just created.
- Assign a Floating IP and open an SSH session to the VM. You can reuse the Floating IP address you previously allocated.

Your network topology should now look like this:



Subtask 2.3 Volumes

- Create an empty Volume of size 2GB and attach it to the VM.
- With an SSH session to the VM, validate that the Volume is available to the VM. Hint: see if the “Attached To” device is present in the VM. Alternatively you can check with ‘`sudo fdisk -l`’ or ‘`lsblk -f`’.

- As the disk is not mounted and formatted it cannot be used by the VM. Format (`sudo mkfs.ext4 $DEVICE_NAME`), create a directory to mount (attach) the disk to and mount the disk (`sudo mount $DEVICE_NAME $DIRECTORY`) and create a new file (e.g. '`myfile.txt`' - you'll have to use `sudo` for this) within that new disk. Unmount (`umount`) the disk and remove the association with the Volume. Confirm the volume is no longer attached using `lsblk` or `fdisk` as above. Reattach the Volume and mount it. Is your file still there? What would happen to the file on your volume if the VM is destroyed?
- Can the Volume be resized? If so how? Is the data still there? Are there advantages to this over snapshotting?

Subtask 2.4 Management

- Find the resource ID of the VM. What is its value? Why is it useful?
- Again, look at the boot log of your VM. What is the IP address assigned to the VM. Is the IP address from your subnet? Is it the same as the Floating IP? If not, why?
- Is there any additional information (metadata) available about the VM?

Subtask 2.5 VM Scaling

One of the powers that Cloud Computing gives you is the power to scale - both horizontal as well as vertical.

- What is the difference between horizontal and vertical scaling? Discuss with your partner and/or lab assistants.
- Now vertically scale an existing VM. Identify all stages of the workflow and verify the result.
    - For instance: Use the linux '`df`' command to verify that the new VM has more disk space compared to the original VM or the command "`cat /proc/cpuinfo`" to see added or removed CPU resources.
- When do you foresee vertical scaling to be useful in a practical scenario?
- Is the Volume still attached? Is the Disk of the Volume mounted? Is the data in the attached Volume affected?

Task 3: Cleanup - Stop the bills!

As cloud computing, IaaS in our case, is pay per use, you will be billed while your resources are created - even if not used. Prompt deletion of unused resources is a must. Otherwise you will pay for things that you are not using! The resources to be cleaned up include:

- Delete the Volume.
- Remove the SSH keys ad Security Groups and rules that you created.
- Delete the Network and Subnet.
- Release the Floating IPs.
- Delete the Router.
- Delete the VM.

Is the order of deletion important? Why?

Additional Documentation

- OpenStack Horizon documentation can be found on the following pages:

    - User Guide: https://docs.openstack.org/horizon/latest/
- Creating SSH key pair for Windows:
    - https://www.ssh.com/ssh/putty/windows/puttygen#sec-Creating-a-new-key-pair-for-authentication
    - https://devops.ionos.com/tutorials/use-ssh-keys-with-putty-on-windows/