Uppsala University

Applied Cloud Computing

1TD265 Autumn 2020

Assignment 1

Henrik Schulze

12th October 2020

ABSTRACT

The aim of this computer assignment is to give hands-on experience with the cloud computing infrastructure used in this course, the SNIC Science Cloud (SSC) (`https://cloud.snic.se`). SSC is a resource that provide Infrastructure-as-a-Service (IaaS) based on the OpenStack cloud software (Newton release) and Ceph storage and offers the following basic services:

1. Compute
2. Storage (Volume and Object)
3. Identity management
4. Image
5. Network
6. Orchestration

1 Task-1 (Provisioning a Virtual Machine)

1. Start an instance of Ubuntu 18.04 with 1 VCPUs [1].

   Answer.

   Before logging in to `https://east-1.cloud.snic.se` I open up a linux terminal. Since I am on Windows 10, this will be a Windows Subsystem for Linux (WSL) terminal window. In the linux terminal I type:
   `ssh-keygen`
   to create an SSH key-pair for convenient logins to the virtual machine(s) that I will create below.

```
henke@hp:/mnt/c/sz/dit/acc$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/henke/.ssh/id_rsa):
/home/henke/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/henke/.ssh/id_rsa.
Your public key has been saved in /home/henke/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:aue91rLHArLQcWgFD/oZunNmOUQKjbk0nmkap4QR4ZI henke@hp
The key's randomart image is:
+---[RSA 2048]----+
|o.    o.         |
|.o + . o.        |
|E.* o oo.        |
|o+ * =+o.        |
|o.B oo+oS        |
|.*  .ooo.        |
|o   o.Oo.. o     |
|     *.+ .+ +    |
|        ..+*     |
+----[SHA256]-----+
```

   Then I use **File Explorer** to copy the files `id_rsa` and `id_rsa.pub` from
   `%LocalAppData%\lxss\home\henke\.ssh`
   to some other suitable place on `C:\` so that I don't loose this SSH key-pair.

   I now log in to `https://east-1.cloud.snic.se/`, choosing project `UPPMAX`. In the openstack dashboard I click `Compute > Key Pairs`. Then I click on the button/rectangle displaying `Import Public Key`. In the upcoming window, under `Key Pair Name*` I type the desired name of my new SSH key-pair, in my case `Henke-KeyPr-200916a`. As `Key Type*` I choose `SSH Key`, and under `Load Public Key from a file` I click `Choose File` and then double-click on the file `id_rsa.pub` that I previously saved to my hard drive (see above). Finally I click on the blue button down to the right labelled `Import Public Key`. I have now imported the SSH key-pair needed in the next step to create a new instance of `Ubuntu 18.04`.

   Next I click `Compute > Instances`.

   If applicable: in the leftmost button's drop-down menu I click to choose `Instance Name =` and in the field next to it I type a few letters of an instance that I created a few days ago, in my case `Henk` (to see if it is still there or if it has been deleted).

   Next I click on the button `Launch Instance`. In the window that comes up, the left pane has items such as `Details, Source*, Flavor*, Networks`, and so on.

In the left pane of the openstack dashboard `Details` is now high-lighted. Under `Instance Name*` I type `Henke-200916d`. Under `Description` I type `Henke-200916d-Description`. Under `Availability Zone` I leave the value `nova`, and finally under `Count*` I type `1` (the digit one).

In the left pane I now click on `Source*`. Under `Select Boot Source` I choose `Image`. Under `Create New Volume` I click `No`. Under `Available` I now see just one row. The name is `Ubuntu 18.04`. Rightmost on the row there is an arrow pointing upwards. I click on the arrow to make the Ubuntu image move up to the `Allocated` section.

In the left pane I now click on `Flavor*`. Under `Available` I click on the upwards arrow of the row with name `ssc.xsmall`.

If `Networks` has a star(*) next to it, I click on `Networks*` and then select the `UPPMAX` network.

Finally, because I have (previously) imported more than one SSH key-pairs, in the left pane I click on `Key Pair` (even if it has no * attached). I select one of the available key-pairs. At last I create the instance by clicking the blue button (not the red pill!) `Launch Instance`. I am now done with this task. The dashboard displays my instance. It currently has a private IP address (192.168.2.253) but no floating/public IP.

2. Assign a floating IP to the instance.

   Answer.

   On the row displaying my instance, the rightmost column is named `Actions`. In the dropdown menu of `Create Snapshot` I click on the down arrow, and then on `Associate Floating IP`. A window titled `Manage Floating IP Associations` comes up. Under `IP Address*` I click on an arbitrary row (in my case 130.238.29.233), and then on the `Associate` button down to the right. The column `IP address` now displays both the private IP (192.168.2.253) and the floating or public IP (130.238.29.233).

3. Access the instance using the SSH client (on Windows, use WSL) and install the program "cowsay". The user name on the Ubuntu image is "ubuntu".

   Answer.

   In the linux terminal I type:

   ```
   ssh -i id_rsa ubuntu@130.238.29.233
   ```

   to login to my virtual machine at SNIC/UPPMAX. I get a message starting with:

   ```
   The authenticity of host '130.238.29.233 (130.238.29.233)' can't be established.
   ```

   The message ends with the question:

   ```
   Are you sure you want to continue connecting (yes/no)?
   ```

   I answer by typing `yes` and pressing `Enter`. (For some reason I had to do the above twice.)
   The command prompt displays:

   ```
   ubuntu@henke-200916d:~$
   ```

   This confirms that I am now logged in to the virtual machine `henke-200916d` as user `ubuntu`.

   I do ***not*** install the program `cowsay` at this point.

4. Open port 4567 on the instance.

   Answer.

   In the left pane of the dashboard I click `Network > Security Groups`. In the search field I type `def`. By doing this, only the `UPPMAX 2020/1-2 default security group` is displayed. In the rightmost column `Actions` I click on `Manage Rules` which takes me to `Manage Security Group`

`Rules:default`. If I click on the column `Port Range` I can see that one of the rows has the value 4567 set as port. This shows that port 4567 has already been opened.

5. Create a snapshot of the instance.

   Answer.

   I click `Compute > Instances` and filter by `Henke` to display my instance. In the rightmost column `Actions` I click on `Create Snapshot` to see a window titled `Create Snapshot`. Under `Snapshot Name*` I type `Henke-Snapshot` and then click on the button `Create Snapshot`. My snapshot is now queued under `Compute > Images`. I click `Delete Image` to immediately delete the snapshot.

1.1 Questions and Answers for Task-1:

1. What is the difference between the private IP and the floating IP?

   Answer.

   A *private* IP address is assigned to an instance's network-interface by the DHCP server and is visible only from within the private network. A *floating* or *public* IP address is a service provided by Neutron and is not using any DHCP service or being set statically within the virtual machine. Actually, the virtual machine's operating system does not know it was assigned a floating IP address [2].

2. Can you access the Internet from the VM without assigning a floating IP to the machine?

   Answer.

   No. - A virtual machine must have a floating IP adress in order to reach the internet. The floating IP address assigned to the VM is its identifier for communication with the outer world. Cloud users need to explicitly "grab" a floating IP from the pool configured by the OpenStack administrator and then attach them to their instance [3].

3. What is the difference between image, instance and snapshot?

   Answer.

   Both a disk image and a disk snapshot reflect the contents of a persistent disk at a point in time. An image also includes an operating system and a boot loader program which can be used to boot an instance. Image and snapshot are two examples of instances [4].

4. What is the name of the OpenStack service responsible for providing the:
   a) Image Service
   b) Compute Service

   Answer.

   a) The Image service is called "glance", and enables users to discover, register, and retrieve virtual machine images. It offers a REST API that enables querying the virtual machine image metadata and retrieve an actual image. [5]
   b) The Compute service is called "nova". It supports creating virtual machines [6].

2 Task-2 (Block Storage)

1. Create a volume of size 1 GiB.

   Answer.

   In the openstack dashboard I click `Volumes > Volumes`. Then I click `Create Volume`. A new window titled `Create Volume` is displayed. Under `Volume Name` I type `Henke-Vol-200917a`. I note that `Size (GiB)*` is set to `1 GiB`. Without changing any settings (other than the volume name) I click on the blue button `Create Volume` down to the right. My `1 GiB` has now been created.

2. Attach your newly created volume to your instance.

   Answer.

   In the openstack dashboard I now click `Compute > Instances`. In the leftmost button's dropdown menu I choose `Instance Name =` and in the field just to the right of it I type `Henke` and then click `Filter`. The only item displayed now is my instance `Henke-200916d`. Next to the rightmost button labeled `Create Snapshot` I click on the downwards arrow and then `Attach Volume`. A new window titled `Attach Volume` comes up. In the dropdown menu under `Volume ID*` I choose the volume that starts with `Henke-Vol-200917a` and then the blue button labeled `Attach Volume`. The `1 GiB` volume has now been attached to the instance `Henke-200916d`.

3. Access the volume and copy a file to the attached volume.

   Answer.

   My course mate Tabea Haverkamp provided me with at helpful link for this task [7].

   As in Task-1.3 above, I log in to my virtual Ubuntu machine from a linux terminal:

   ```
   ssh ubuntu@130.238.29.233
   ```

   Next I follow the suggestions in the github link that Tabea kindly gave me [7]:

   ```
   sudo lsblk -f
   ```

   to see all the volumes that are currently mounted. I then format the volume `Henke-Vol-200917a`, created in Task-2.1 above, with an ext4 file system:

   ```
   sudo mkfs.ext4 /dev/vdb
   ```

   Next I create the folder `volume` and mount `/dev/vdb` to this folder:

   ```
   mkdir volume
   sudo mount /dev/vdb volume
   ```

   By once again running `sudo lsblk -f`, I can see that the mountpoint of `vdb` is now: `/home/ubuntu/volume` which confirms that the mounting was successful.

   I now create a file to be copied to the volume attached to my Ubuntu 18.04 instance:

   ```
   echo "Hello world!">hi.txt
   ```

   and then do the actual copying:

   ```
   sudo cp hi.txt volume/
   ```

   To confirm that the file was indeed copied to the volume, I run:

   ```
   cat volume/hi.txt
   ```

   which outputs:

   ```
   Hello world!
   ```

   To take it one step further, I can unmount the volume and then try to output the contents of the text file:

   ```
   sudo umount /dev/vdb
   sudo lsblk -f (just for checking)
   ```

```
cat volume/hi.txt
```

The last command results in:
`cat: volume/hi.txt: No such file or directory` which confirms that the volume at `/dev/vdb` is no longer mounted.

4. Modify the size of the volume created in step 1.

   Answer.

   To resize a volume, I must first detach it from the server [8].

   In the openstack dashboard I click `Compute > Instances` and filter to find my instance. In the rightmost column `Actions`, in the dropdown menu I click `Extend Volume`. Under `New Size (GiB)*` I type `2` and then click on the blue `Extend Volume` button. By filtering again, I can confirm that the size of the volume is now `2 GiB`.

2.1 Questions and Answers for Task-2:

1. What is the technology used to provide volumes in OpenStack? Is it RAID or LVM?

   Answer.

   It is LVM. The Logical Volume Manager (LVM) provides a method of allocating space on mass-storage devices that is more flexible than conventional partitioning schemes [9]. RAID stands for 'Redundant Array of Independent Drives' and is a way of combining multiple, physical drive units into one, logical unit [10].

2. What is LVM? Explain the advantage(s) of using LVM.

   Answer.

   The Logical Volume Manager (LVM) is a Linux-based system that provides an abstraction layer on top of physical disks to expose logical volumes to the operating system [11].

   LVM abstracts the physical disks away from the operating system. The main benefit of this is the ability to grow file systems on the fly [12].

3. Can one volume be attached to multiple instances or vice versa?

   Answer.

   The Volume multi-attach feature enables users to attach and access a single block storage volume to and from multiple servers [13].

   The reverse is not true - with the OpenStack Block Storage service, you can attach a volume to only one instance at a time [14].

4. Explain the main difference between Ephemeral Storage and Block-Storage. What are the major use-cases for the different storage types?

   Answer.

   In addition to image storage (glance) and object storage (swift), there is also block storage (cinder) and ephemeral storage (nova). With ephemeral storage - the OpenStack Compute service (nova) - users do not have access to any form of persistent storage. From the user's point of view the ephemeral disks disappear when a virtual machine is terminated [15] [11].

   The OpenStack Block Storage service uses the Logical Volume Manager (LVM) for Linux [14].

   Users who need fast access to many objects that do not change often, or who want to set a time-to-live (TTL) value on a file will benefit from block storage. On the other hand, users who only want something computed and do not need the data to persist beyond the life of the virtual machine can benefit from the ease and simplicity of ephemeral storage [16] [17].

5. Does your virtual machine have ephemeral storage?

   Answer.

   Yes - it is of type nova.

6. What is the name of the OpenStack service providing volumes?

   Answer.

   Cinder is the OpenStack Block Storage service for providing volumes to Nova virtual machines, Ironic bare metal hosts, containers and more [19].

3 Task-3 (Network)

3.1 Questions and Answers for Task-3:

1. Explain the picture in the tab "Network Topology"

   Answer.

   The blue vertical line to the left represents the public internet, while the orange vertical line to the right symbolizes the internal network at UPPMAX to which all the virtual machines (guests) are connected [18].

2. What is the subnet used by the Tenant?

   Answer.

   The subnet range is 192.168.2.0/24 or equivalently 192.168.2.2 − 192.168.2.254 [18].

3. What is the role of the router?

   Answer.

   A router is a logical component that forwards data packets between networks. It also provides network address translation to provide external network access for servers on a local network [20].

4. Explain the path of the traffic of the virtual machine to the Internet.

   Answer.

   The picture at `https://aptira.com/wp-content/uploads/2016/03/5-1.png` describes the traffic well. The figure shows how a virtual machine VM1, can reach some IP address on the public internet. VM1 is connected to the virtual router for external connectivity, a packet is able to reach the virtual router with a source IP of 10.10.10.5 and a target IP of 56.57.58.59. The virtual external router EXT_VR then changes the source IP to its own IP address (192.168.0.19) and sends the packet to the correct destination [21].

5. Find out the unique ID of the external network.

   Answer.

   In the left pane of the openstack dashboard I click:
   `Network > Networks > Public External IPv4 Network > Overview`

   where I can read the Network ID:
   `9187404b-b24b-4ee5-b5f4-22d9a15dc4e2`.

6. What is the name of the OpenStack service handling Networks?

   Answer.

   Neutron is an OpenStack project to provide "network connectivity as a service" between interface devices managed by other OpenStack services (for example, nova) [21].

4 Task-4 (Cowsay as a Service)

Answer.

As in Task-1.3 and Task-2.3 above, I log in to my virtual machine from a linux terminal.
At this point, I install `python3-pip` and `flask`:

```
sudo apt update               # takes just a few seconds
sudo apt install cowsay       # takes about half a minute
sudo apt install python3-pip # takes about a minute
pip3 install flask           # takes up to ten seconds
```

I now clone the *Cowsay as a Service* github repository, like so:

```
git clone https://github.com/TDB-UU/csaas.git
```

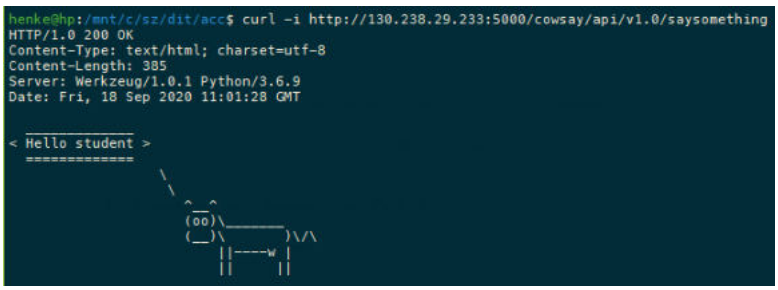
Next I execute:

```
python3 csaas/cowsay/app.py
```

From a completely different linux terminal (not being logged in to the VM), I now run the command:

```
curl -i http://130.238.29.233:5000/cowsay/api/v1.0/saysomething
```

This other terminal now displays:



I have deliberately chosen **not** to use the `screen` command.

4.1 Questions and Answers for Task-4:

1. Examine the code in app.py.
   What Python framework is used to provide the (extremely simplistic) RESTful service?

   Answer.

   The code in app.py [22] uses the `flask` Python framework [23].

2. What problem does "screen" solve?

   Answer.

   The `screen` command creates a new window with a shell in it, run a command, and then pushes the window to the background [24]. The problem this solves is that it allows for a server to be started, and then for a client to run a completely new command in another window.

3. Write a short description of the steps you followed to complete the Task-4.

   Answer.

   See my answer above directly under **Task-4 (Cowsay as a Service)**!

4. Is the SNIC Science Cloud a Public, Community, Private or Hybrid cloud, and why?

   Answer.

   The Swedish National Infrastructure for Computing Science Cloud (SSC) is a *community cloud* and use of its resources is free of charge to Swedish principal investigators and their collaborators [25].

References

[1] "Launch and manage instances." https://docs.openstack.org/horizon/latest/user/launch-instances.html. Accessed on 2020-09-15.

[2] "Difference between Floating IP and private IP." https://www.rdoproject.org/networking/difference-between-floating-ip-and-private-ip/. Accessed on 2020-09-15.

[3] "Configuring Floating IP addresses in OpenStack." https://www.mirantis.com/blog/configuring-floating-ip-addresses-networking-openstack-public-private-clouds/. Accessed on 2020-09-15.

[4] "What is the difference between disk snapshot and disk image?." https://stackoverflow.com/questions/27290731#29634892. Accessed on 2020-09-17.

[5] "Image service." https://docs.openstack.org/newton/install-guide-rdo/glance.html. Accessed on 2020-09-17.

[6] "OpenStack Compute (nova)." https://docs.openstack.org/nova/latest/. Accessed on 2020-09-17.

[7] "Howto: Creating and using Volumes on a linux instance." https://github.com/naturalis/openstack-docs/wiki/Howto:-Creating-and-using-Volumes-on-a-Linux-instance. Accessed on 2020-09-17.

[8] "Resize a volume." https://docs.openstack.org/cinder/latest/cli/cli-manage-volumes.html#resize-a-volume. Accessed on 2020-09-17.

[9] "Logical Volume Manager (LVM)." https://docs.openstack.org/newton/admin-guide/common/glossary.html#term-logical-volume-manager-lvm. Accessed on 2020-09-18.

[10] "RAID storage: A guide." https://www.krcs.co.uk/news/a-guide-to-raid-storage. Accessed on 2020-09-18.

[11] "Storage concepts." https://docs.openstack.org/arch-design/design-storage/design-storage-concepts.html. Accessed on 2020-09-18.

[12] "What are the benefits of the Logical Volume Manager." https://serverfault.com/questions/272846. Accessed on 2020-09-18.

[13] "Meet volume multi-attach, a great new feature in OpenStack Queens." https://superuser.openstack.org/articles/meet-volume-multi-attach-great-new-feature-openstack-queens/. Accessed on 2020-09-18.

[14] "Manage volumes." https://docs.openstack.org/cinder/queens/admin/blockstorage-manage-volumes.html. Accessed on 2020-09-18.

[15] "Storage architecture." https://docs.openstack.org/project-deploy-guide/openstack-ansible/draft/overview-storage-arch.html. Accessed on 2020-09-18.

[16] "Storage architecture." https://docs.openstack.org/arch-design/design-storage/design-storage-arch.html. Accessed on 2020-09-18.

[17] "OpenStack Storage Options And Use Cases." https://docs.platform9.com/openstack/storage/storage-options. Accessed on 2020-09-18.

[18] "Network Topology." https://east-1.cloud.snic.se/project/network_topology/. Accessed on 2020-09-18.

[19] "OpenStack." https://en.wikipedia.org/wiki/OpenStack. Accessed on 2020-09-18.

[20] "router." https://docs.openstack.org/python-openstackclient/pike/cli/command-objects/router.html. Accessed on 2020-09-18.

[21] "Everything you need to know to get started with Neutron." https://superuser.openstack.org/articles/everything-you-need-to-know-to-get-started-with-neutron-f90e2797-26b7-4d1c-84d8-effef03f11d2/. Accessed on 2020-09-18.

[22] "app.py." https://github.com/TDB-UU/csaas/blob/master/cowsay/app.py. Accessed on 2020-09-18.

[23] "Flask (web framework)." https://en.wikipedia.org/wiki/Flask_(web_framework). Accessed on 2020-09-18.

[24] "How to Use Linux's screen Command." https://www.howtogeek.com/662422/how-to-use-linuxs-screen-command/. Accessed on 2020-09-18.

[25] "SNIC Science Cloud." https://www.snic.se/allocations/ssc/. Accessed on 2020-09-18.