UNIVERSITY OF TARTU
Institute of Computer Science

(/)

Basics of Cloud Computing 2018/19 spring

Cus

- Main (https://courses.cs.ut.ee/2019/cloud/spring/Main/HomePage)
- Lectures (https://courses.cs.ut.ee/2019/cloud/spring/Main/Lectures)
- Practicals (https://courses.cs.ut.ee/2019/cloud/spring/Main/Practicals)
- Results (https://courses.cs.ut.ee/2019/cloud/spring/Main/Results)
- Submit Homework (https://courses.cs.ut.ee/2019/cloud/spring/Main/Homework)

Practice 1 - Introduction to Infrastructure as a Service

In this practice session you learn how to access cloud services that we will be using in the rest of the course. We will be mainly using a private university cloud - which is a Cloud infrastructure running on the hardware of the University of Tartu and is managed by the High Performance Computing Center. In this lab we are working on the OpenStack cloud platform, located at: https://stack.cloud.hpc.ut.ee/ (https://stack.cloud.hpc.ut.ee/)

- To access the local university cloud resources your computer has to be inside the Institute network. So you should either use lab computers, Eduroam Wifi (inside the institute building) or set up a VPN connection to university network.
  - VPN (English on the right side column) - https://wiki.ut.ee/pages/viewpage.action?pageId=17105590 (https://wiki.ut.ee/pages/viewpage.action?pageId=17105590)
  - Eduroam Wifi (English on the right side column) - https://wiki.ut.ee/display/AA/Eduroam (https://wiki.ut.ee/display/AA/Eduroam)

NB! Students have previously reported that using Eduroam in dormitories will not give access to the University cloud. You will have to use VPN in such cases.

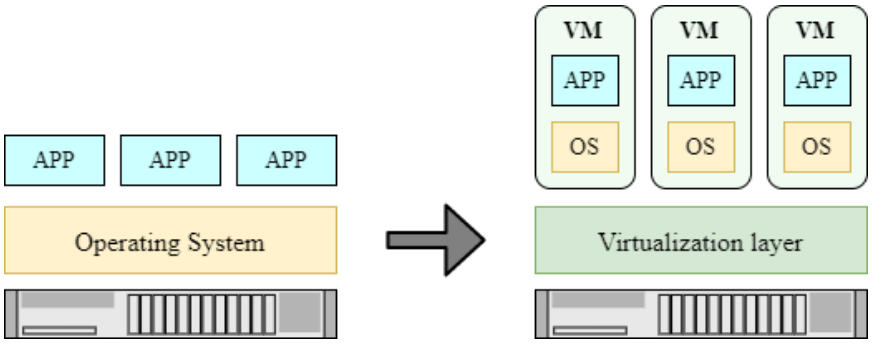
Lab supevisors:

- Pelle Jakovits (jakovits ät ut . ee)
- Alo Peets (alo . peets ät ut . ee)

Feel free to send us an email if you have any questions about the lab materials or issues with submitting deliverables.

Introduction

Infrastructure as a Service (IaaS) is model of Cloud computing, in which Virtualized computing resources are provided to users over the internet. In comparison to using physical servers, computing resources can be provisioned **on-demand** and in **real-time** and applications running on same hardware can be separated into different secure environments, each containing their own OS, software libraries and kernels.



Working with IaaS model of Cloud usually consists of the following steps:

1. Register an account to access the cloud services
2. Select appropriate virtual machine image to run (Ubuntu, Debian, Windows, etc.)
3. Start a new instance of the selected virtual machine image. Login into the instance as a root user over the internet and configure it to meet your requirements. I.e. install needed software, upload your own application, perform any required configuration actions as you would do in any real computer.
4. As you will lose all your work when instance will be terminated -- you have three options on how to persist the changes you made:
    1. Save all your configuration steps to a script that will launch and configure the instance automatically for you.
    2. Bundle a new image from your running instance and next time launch your custom image.
    3. Save the running instance as a snapshot, and next time launch new instances from there.

First option is more flexible as you can easily change the script than bundle a new image if something changes. Second and third option are simpler to use once you have stable configuration or when launching large number of instances.

In this lab we are working on the OpenStack cloud platform.

Exercise 1.1. Accessing the cloud services

In this exercise you will log into the institute OpenStack cloud and create a secure access key.

Verify that you have access to the university OpenStack cloud resources and familiarize yourself with the available cloud functionality.

- Log into https://stack.cloud.hpc.ut.ee/ (https://stack.cloud.hpc.ut.ee/) using your university `username` and `password` and `ut.ee` as domain.
    - Contact lab supervisor if you have issues while logging in. Make sure to include your university username in the email, as it is required for configuring access.
    - You have to be in the university network to access this website. Use (https://wiki.ut.ee/pages/viewpage.action?pageId=17105590) VPN or log into eduroam, if you're using your own laptop
- Familiarize yourself with the available OpenStack cloud functionality.
- Create a ssh Key Pair for accessing Virutal Machines over the network. Make sure the name of the Key Pair includes your last name!
    - You will find this functionality under **Compute -> Key Pairs**
    - **NB!** `Copy private key to Clipboard` and save it to your computer in text file with extension `*.pem` into a location from where you can easily find it later.
    - If using `Putty` in Windows to connect to cloud instance with ssh you should use PuTTYgen

(https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html) to convert certificate into Putty specific `*.ppk` format. Use `Load` and `Save private key` functinality in PuttyGen program to do it.
    - PEM or Privacy Enhanced Mail is a Base64 encoded DER certificate. PEM certificates are frequently used for servers as they can easily be translated into readable data using a simple text editor.

Exercise 1.2. - Requesting computing resources from the cloud

In this exercise you will start a Cloud instance (or virtual machine) while specifying it's configuration and computing resources available for it.

- Use the OpenStack web interface
- Under the "Compute" tab go to "Instances" and start a new instance by clicking the "Launch Instance" button (If not specified leave the default values)
- Start a new instance of **Ubuntu 18.04** virtual machine image
    - Use your last name as the **Instance Name** under **Details** tab
    - Choose **ubuntu18.04** under **Source** tab & change the volume Size to 10GB
    - Also enable **Delete Volume on Instance Delete** under **Source** tab
    - Choose the capacity of the instance
        - Under **Flavor** tab, choose **m1.small** as the type of the instance
    - Choose network for the instance
        - Under **Networks** tab, choose **provider_64_net**
    - Specify what Key Pair to use under the **Key Pair** tab!
        - use the Key Pair that you created in the previous exercises. If you lose the downloaded file, you will have to create a new one!

Exercise 1.3. Accessing your Cloud instance over the internet

We will use Secure Shell (ssh) protocol to log into the started instance over the internet. Instances in the cloud can have multiple IP addresses. **Public IP** for accessing the instance from outside the cloud and **Private IP** for accessing the instance from inside the cloud (from other instances). However, our instances will only have a single IP in the current configuration.

- Log into the instance through **ssh** using SSH Key based authentication
    - On Linux:
        - `ssh –i path_to_my_key_pair_file ubuntu@<instance public ip address>`
        - For example: *ssh -i .hpc/jakovits_ldpc.pem ubuntu@172.17.64.63*
        - if you get an error, check that the path to the keyfile is correct and that it has correct rights (`chmod 400 <filename>`)
    - On Windows:
        - Either copy the private key pair file to a university linux server (like math.ut.ee) and use the previous ssh command.
        - Or user Putty, SSH secure Shell or WinSCP program to get a command line interface to a remote server through ssh
            - In windows, we first have to transform the private key file (_keyname_.pem) we downloaded from OpenStack into a `.ppk` file.
            - Follow the **To prepare to connect to a Linux instance from Windows using PuTTY** section @ https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/get-set-up-for-amazon-ec2.html#prepare-for-putty

(https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/get-set-up-for-amazon-ec2.html#prepare-for-putty)
    - Username for the SSH connection has to be `ubuntu`
        - Specify it in Putty under `Connection->Data->Login details`
    - Host must be the public IP of the instance you started
    - SSH Key must be the same `.ppk` key you converted with PyttyKeyGen
        - Specify it's location under: `Connection->SSH->Auth->Private key file...`

- **Log into your new instance through ssh**
- change the password of the user **ubuntu**
    - `sudo passwd ubuntu`
    - assign a password you can remember.
    - We have to use sudo because normal use of `passwd` command otherwise requires us to enter the current password, which we do not know.
- Now the user **ubuntu** has a password and we can use it to log into the instance through the web console instead of a ssh client.

Exercise 1.4. Accessing your instance through the web interface

To have acces to your VM when external network connection is down or there is a problem connecting over SSH we can use OpenStack web interface and VNC. Make sure you changed password earlier for user `ubuntu` so you can log into your VM using username:password though the OpenStack web console. Virtual Network Computing (VNC) is a graphical desktop sharing system that uses the Remote Frame Buffer protocol (RFB) to remotely control another computer. It transmits the keyboard and mouse events from one computer to another, relaying the graphical screen updates back in the other direction, over a network.
https://en.wikipedia.org/wiki/Virtual_Network_Computing
(https://en.wikipedia.org/wiki/Virtual_Network_Computing)

- Log into the OpenStack web interface at https://stack.cloud.hpc.ut.ee/ (https://stack.cloud.hpc.ut.ee/)
- Go to Instances page and click on the name of your instance.
- Go to the `Console` Tab and click on the **Click here to show only console** link.
- A command line interface should show up in a few moments. Refresh the page if it does not show up. If you see only black screen try hitting `ENTER` few times.
- Log into the instance using **ubuntu** as the username and the password you previously specified.

- FIX ERROR: "sudo: unable to resolve host <your_machine_name_here>"
    - If you try entering any sudo commands i.e `sudo free` ; `sudo du` you should get and error "unable to resolve ..."
    - In order fix it edit `/etc/hosts` file and add your hostname to the end of first line like this `127.0.0.1 localhost <your_machine_hostname_here>`. You can use `nano` with `sudo` rights to do it.
- Take a screenshot of the web command line interface after you have successfully logged in and executed a `sudo` command without error. Browser should stay visible in the screenshot.

It is up to your preferences whether to use the web interface or the ssh client to access your instances. Web client allows you to avoid using additional ssh software but may not work as well when you need to use a lot of copy/paste commands or transfer files.

Exercise 1.5. Attach a volume to an instance

Volumes are virtual hard disks, which can be used as a more permanent storage of files or extending the available disk space of an instance. We will create a new volume and attach it to our instance to increase the available disk space.

- Create a new `2GB` size volume (Volumes -> Volume -> Create Volume)
  - Use your last name as the name of your volume.
- Attach the volume to your running instance.
  - Go to the list of instances
  - On the right side, next to the **Create Snapshot** button, choose **Attach Volume** from the `drop down menu` and choose the volume you just created.
- Create a new disk partition on the attached volume
  - Log into your instance through ssh
  - Check the list of available disks and their partitions using the `lsblk` command line command
  - The name of the new disk added as a result of attaching your cloud volume should be `vdb`
  - Create a new partition on the disk
    - Run `sudo fdisk /dev/vdb` command to start the partitioning process
    - Press `n` to create a new disk partition.
    - Press `p` to create a primary disk partition.
    - Press `1` to denote it as 1st disk partition.
    - Press `ENTER` twice to accept the default of 1st and last cylinder – to convert the remainder of hard disk to a single disk partition
    - Press `t` to choose a type for the new partition.
    - Press `83` change your new partition to Linux partition type.
    - Finally, press `w` to commit changes.
- Mount the volume inside the instance to `/data` folder
  - Create the `/data` folder: `sudo mkdir /data`
  - Create a ext4 type filesystem on the new partition: `sudo mkfs.ext4 /dev/vdb1`
  - Mount the partition under `/data` folder: `sudo mount –t ext4 /dev/vdb1 /data`
- Create a new file in the mounted folder to verify that everything is working correctly.
  - You can use the `nano` command line text editor: `sudo nano /data/myfile.txt` (**CTR+X** keys to exit the editor)
- Run `lsblk` command in the command line interface inside your cloud instance and take a screenshot of the output.

NB! Once you are done, you must delete your instance and the Volume you created! Also, be careful you do not delete work of other students.

Deliverables:

- Your instance must have been be terminated and the additional volume deleted!
- Screenshots created in exercise 1.4 and 1.5
  - Pack the screenshots into a single zip file and upload them through the following submission form.
- Submit an answer for the following questions:

- What happens if you lose your ssh KeyPair file? What happens to existing instances which were started with the lost ssh key?
- What are the advantages of utilizing cloud Volumes? Briefly describe at least two scenarios, where using volumes simplifies working with cloud instances.

Task lab 1

If your homework consists of multiple files (for example HTML + CSS + JS) it should be archived before submitting.

File Choose File no file selected

Comment

Submit


Study IT in .ee
sponsored by Skype™