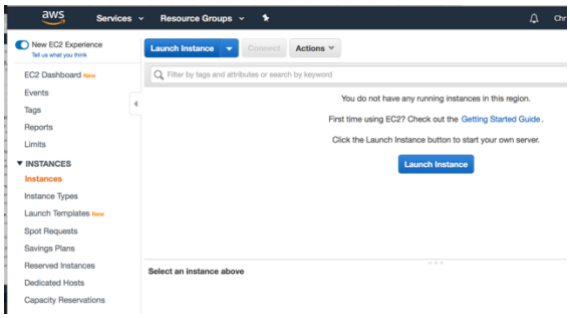
WEEK 4 ASSIGNMENT 1

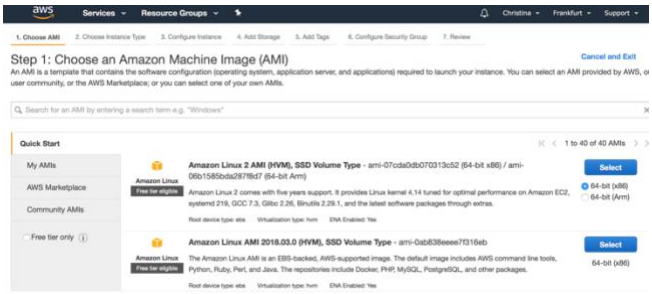
Large-Scale Data Storage Systems – DATA-5400 | Spring 2020

Christina Morgenstern

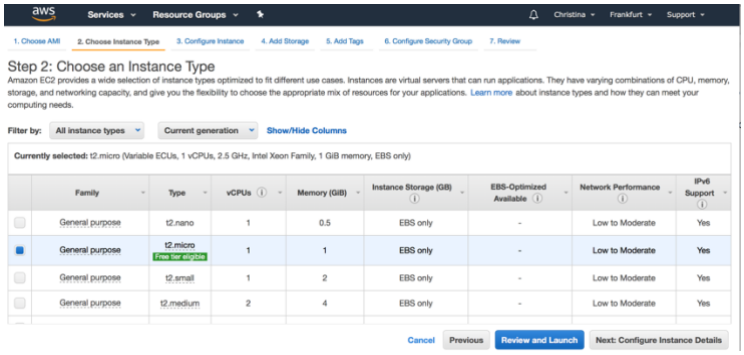
In this assignment, I created a Linux VM in the cloud and ran Linux and some commands on the newly created VM.
Using the Atom editor, I created three text files with .txt extension and saved them in my course directory. The contents of these files are a one-line sentence and the same in each case.
In order to create a Linux VM, I go to my AWS Management Console and select the EC2 service. On the left-hand menu, I choose Instances and then click the blue Launch Instance button, selecting launch instance. As for the region, I stick with the Europe (Frankfurt) eu-central-1 region, which is closest to my home.



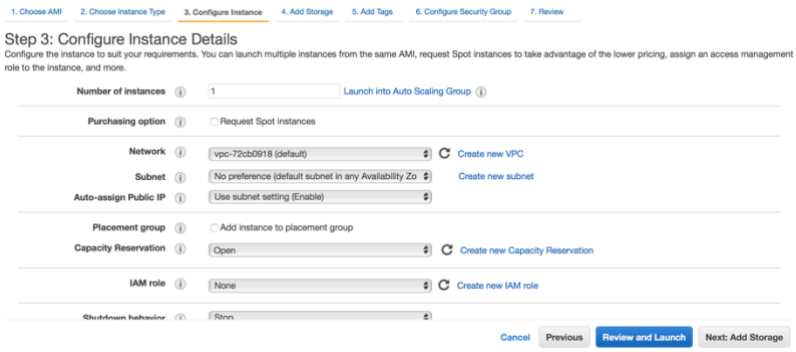
In the next step, you have to choose the type of Amazon Machine Image (AMI), a template containing the software configuration required to launch the instance. To create a Linux VM, I selected the Amazon Linux 2, Free tier, instance.



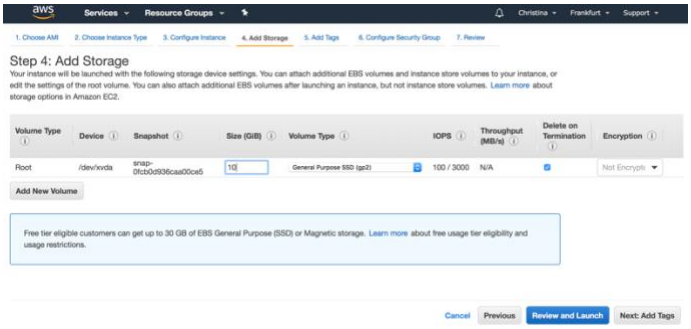
I selected the General Purpose t2.micro type instance with 1 vCPU, 2.5 GHz Intel Xeon and 1 GiB memory of EBS only.



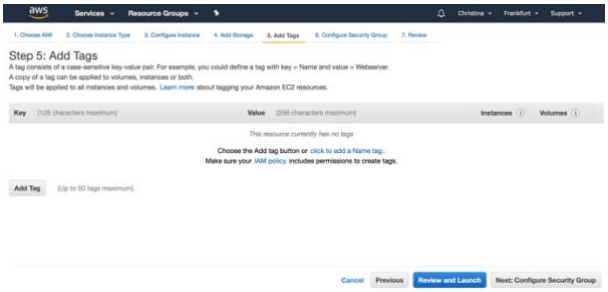
Next, further instance details were configured. Like the number of instances which was set to 1.

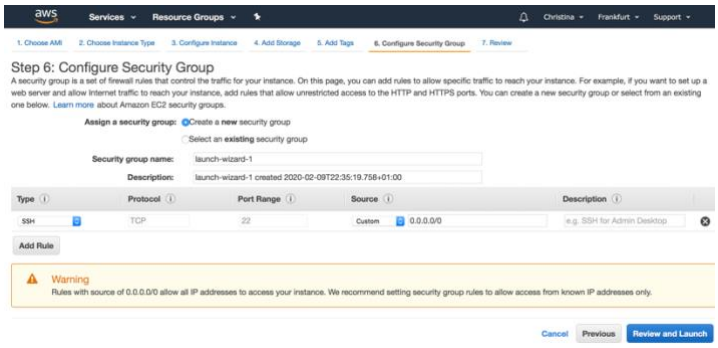


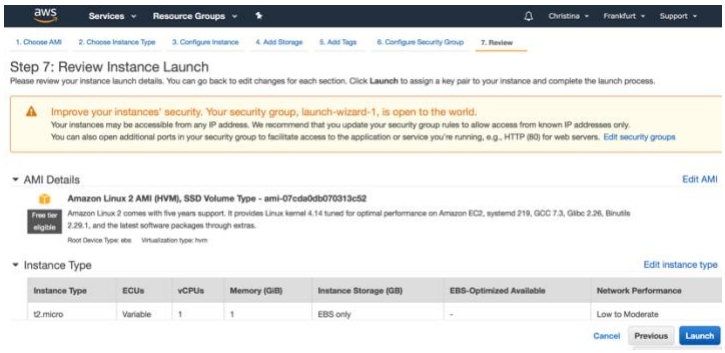
The storage was set to 10 GB.



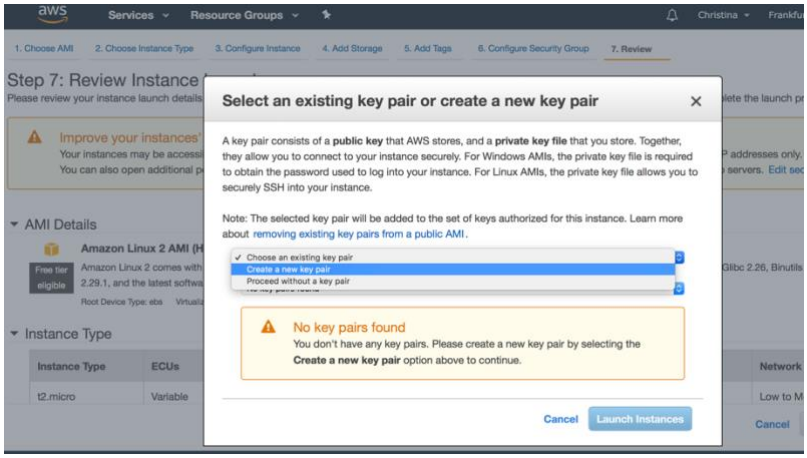
Tags allow you to identify instances. I used tags that identify these VMs with the Large Scale Storage course.

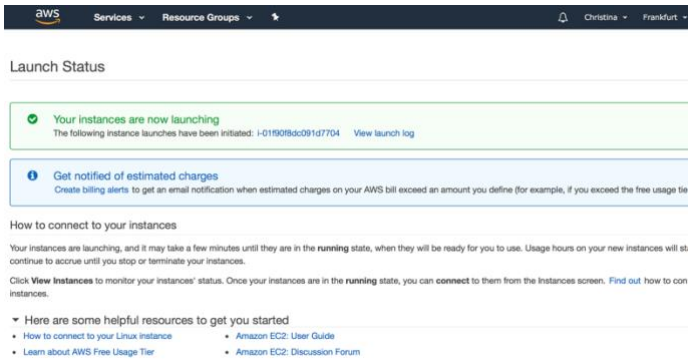




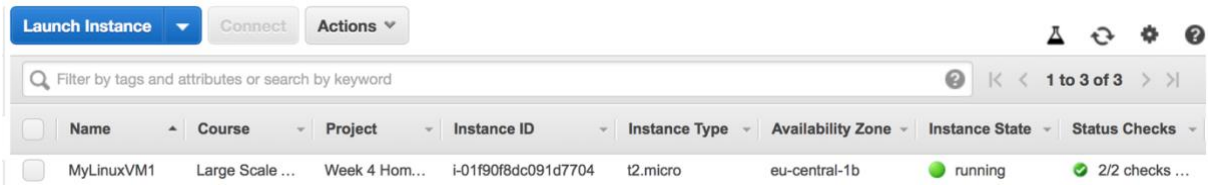


In Step 7, when reviewing the details of the instance, a key pair for authorization of the instance needs to be generated. The key pair file needs to be downloaded as .pem file and was saved on my drive in the same course directory.





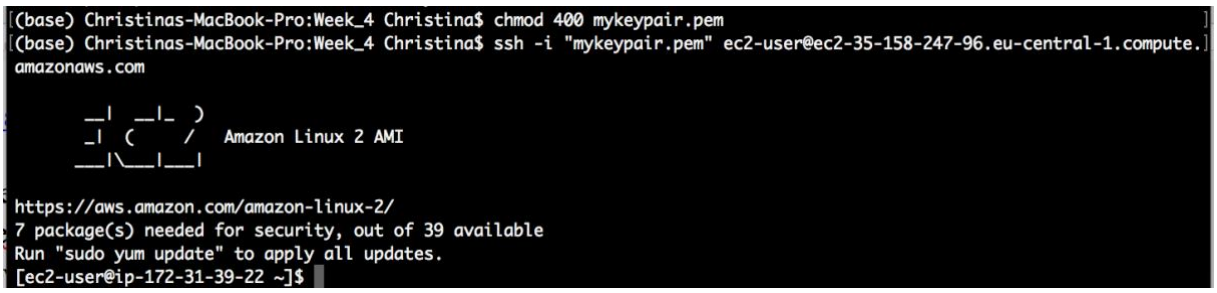
I successfully created a Linux VM, called MyLinuxVM1, using AWS EC2 service. Selecting the VM on the left-hand side lets you explore the details of this instance. Under the description tab, you can find the private and public IP addresses.

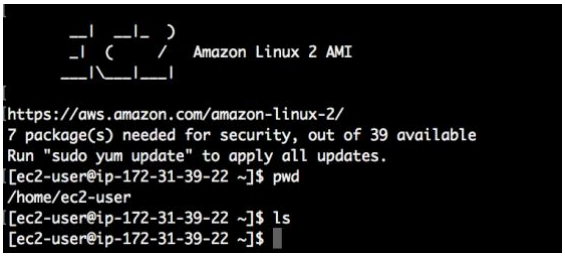


To connect to MyLinuxVM1, I opened the Terminal window on my Mac Laptop and followed the instructions in the AWS documentation (https://docs.aws.amazon.com/quickstarts/latest/vmlaunch/step-2-connect-to-instance.html#sshclient) using SSH:

1. Using the `cd` command, I navigated to the directory containing the private key file and the created text files.
2. The following `chmod` command makes sure that the private key file isn´t publicly viewable:
   `chmod 400 mykeypair.pem`
3. With the following SSH command, I connected to my instance:
   `ssh -i "mykeypair.pem" ec2-user@ec2-35-158-247-96.eu-central-1.compute.amazonaws.com`

These commands can also be found in the EC2 console window upon selecting the instance and pressing the connect button.



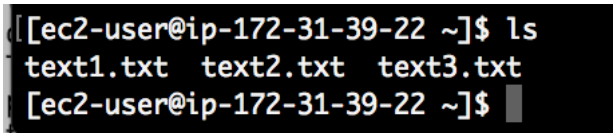


Using the SCP client on my macOS, I transferred the previously created text files to the Linux VM. For that, I opened another terminal window and navigated to the directory with the text files and the secure keys. Then using the following general SCP command, the files were copied to the VM:

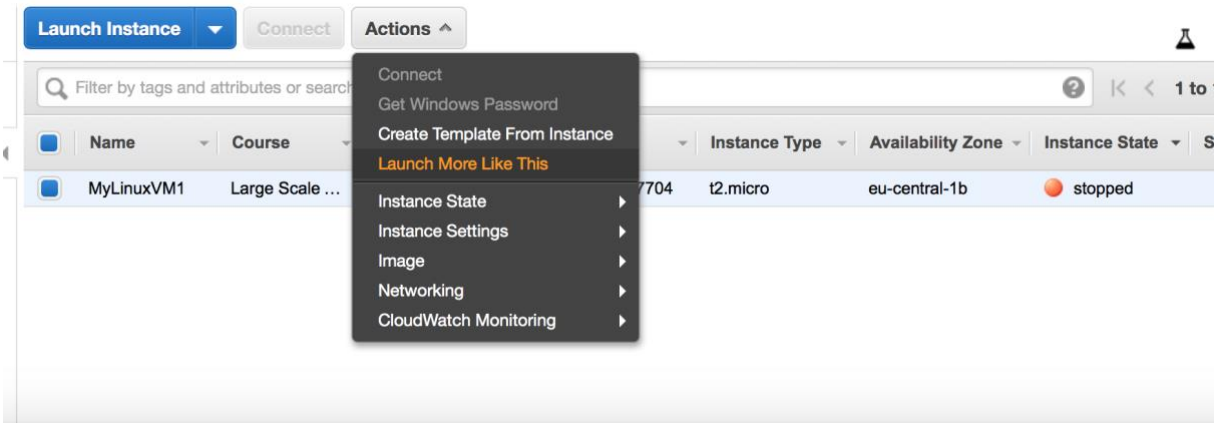
```
scp -i path/to/key file/to/copy user@ec2-xx-xx-xxx-xxx.compute-1.amazonaws.com:path/to/file
```

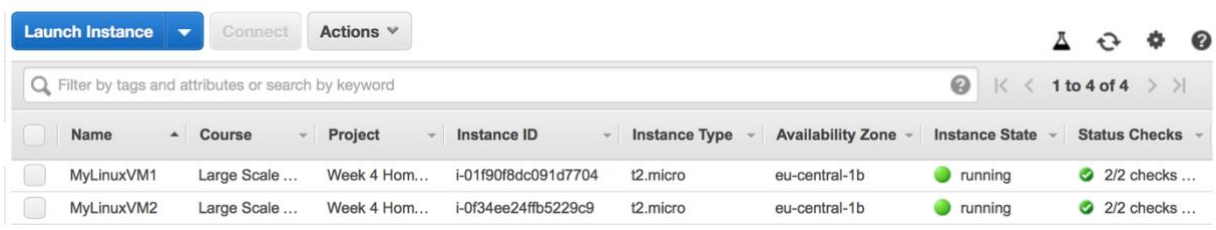
Successful transfer of text1.txt file:

And verification of the transfer of three text files in the terminal window connected to the VM using the `ls` command.

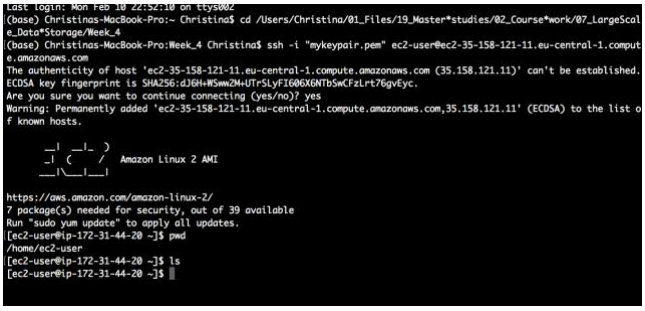


A second Linux virtual machine, named MyLinuxVM2, was created using the Actions button and the Launch More Like This option.



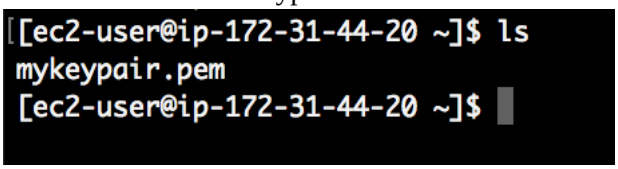


In a third terminal window, I connected to MyLinuxVM2.

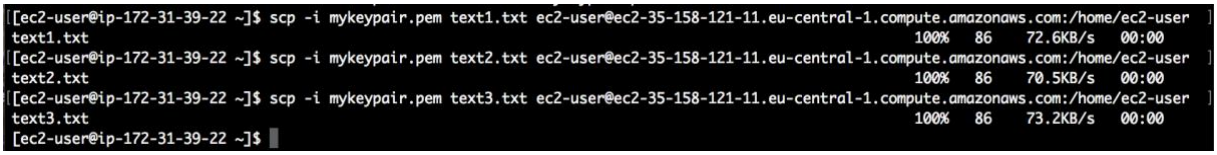


Transfer of the keypair file to VM2 (and accidental transfer of text3.txt file, which was immediately removed using the `rm` command.)

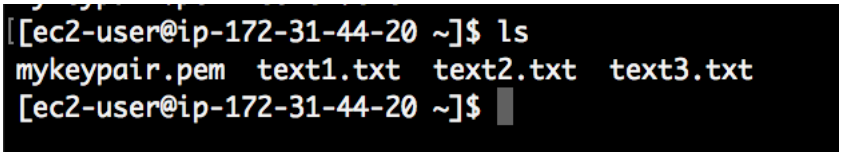
Verification that the keypair file resides within the VM2.



Transferring the three text files from MyLinuxVM1 to the MyLinuxVM2 using the `scp` command.



Verification that the text files were copied to MyLinuxVM2 using the `ls` command.



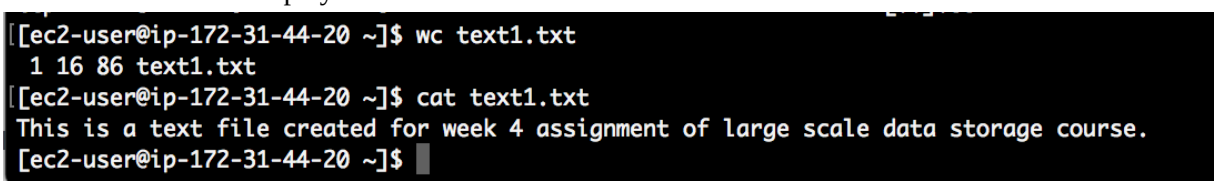
I tested several Linux commands using MyLinuxVM2, like the following:

`ss` command displays network statistics. I have used the `ss` command without any arguments as well as with different arguments (https://www.linux.com/tutorials/introduction-ss-command/).

The `wc` command outputs the number of lines, number of words, number of bytes and the filename (https://shapeshed.com/unix-wc/)

```
$ wc text1.txt
1 16 86 text1.txt
```

The `cat` command displays the file contents:



The `history` command shows the previously used commands:

```
[ec2-user@ip-172-31-44-20 ~]$ history
    1  pwd
    2  ls
    3  rm text3.txt
    4  ls
    5  ss
    6  ss -l
    7  wc text1.txt
    8  cat text1.txt
    9  history
[ec2-user@ip-172-31-44-20 ~]$
```

The `which` command helps to locate executable files, like where python resides.

```
[ec2-user@ip-172-31-44-20 ~]$ which python
/usr/bin/python
```

Using the `exit` command, the terminal windows were closed, and the instances stopped within the AWS console window.