



Cloud Computing

4th laboratory

Tasks

1.

Create a student account on AWS Academy Canvas LMS, and create and connect to an EC2 instance.

1. Log into your @student.tuiasi.ro email account and click the **Get started button** in the email from **AWS Academy** entitled **Course Invitation** in order to access the Canvas LMS.
The authentication URL is <https://awsacademy.instructure.com/login/canvas>
2. From AWS Canvas access the **AWS Academy Learner Lab - Foundation Services (ALLFv1-11643)** course, click on **Modules -> Learner Lab - Foundational Services**.
If you use the browser in private mode you will not be able to access the Laboratory.
3. Click on **Start Lab** and wait for the circle to the left of the AWS link to turn green .
4. To access the **AWS Management Console**, click on the  link in the upper left part of the webpage.
5. From the Console Home, click on the **EC2** service (or from the upper left menu -> Service -> Compute -> EC2).
6. Click on **Instances** on the left-hand side menu.
7. Click on **Launch Instance** -> select **Red Hat Enterprise Linux 8 (HVM), SSD Volume Type** -> click on Review and Launch -> Launch.
8. Create a new key pair and store it locally (save it on moodle/email and delete it from the faculty computer after the laboratory is finished).
9. After the instance is created, select the instance and click Connect. Select the **SSH client** tab and follow the instructions to connect from your own terminal to the newly created instance.

Connect to the AWS instance on Windows

1. Open Windows PowerShell (press the windows key and write PowerShell)
2. Navigate to the location of your AWS key pair
3. Execute the following commands to change the permissions of the key pair and connect to the AWS instance (replace *keyPairName* with the your AWS key pair filename):

```
icacls.exe keyPairName /reset  
icacls.exe keyPairName /GRANT:R "$($env:USERNAME):(R)"  
icacls.exe keyPairName /inheritance:r  
ssh -i "keyPairName" ec2-user@publicAddressFromAWS
```

* The last line is from AWS, when you try to connect to an instance.

The *publicAddressFromAWS* is something like ec2-3-89-193-193.compute-1.amazonaws.com

Do not forget to stop your instances after you are done with them

On AWS -> EC2 -> Instances -> select your instance -> Actions -> Instance State -> Stop.

Connect using EC2 Instance Connect

<https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-instance-connect-methods.html>

2.

Install Docker on the newly created instance by running the instructions:

```
sudo yum install -y yum-utils device-mapper-persistent-data lvm2
sudo yum-config-manager --add-repo
https://download.docker.com/linux/centos/docker-ce.repo
sudo yum install -y --nobest docker-ce docker-ce-cli containerd.io
sudo yum install -y iptables
sudo systemctl start docker
sudo docker run hello-world
```

3.

A. Install Docker Compose on your Amazon EC2 instance by following the tutorial from Docker Compose – Install: <https://docs.docker.com/compose/install/>

When you install docker compose do not forget to execute the following command:

```
sudo ln -s /usr/local/bin/docker-compose /usr/bin/docker-compose
```

B. Follow the tutorial from Docker Compose Getting started and test the example on your Amazon EC2 instance: <https://docs.docker.com/compose/gettingstarted/>

The Python container from the example exposes port 5000 and the docker compose YAML file forwards port 5000 from the container to port 8000 from the host (i.e., the AWS VM instance).

In order to start the docker composer: in detached mode execute:

```
sudo docker-compose up -d
```

C. Access the web server (localhost:8000) using curl from the EC2 instance.

D. Access a webpage using curl from the host operating system (your Linux/Windows)

On the AWS EC2 website -> Security Groups -> Select the launch-wizard-1 security group -> from Actions select Edit inbound rules -> Add Rule: **Type: Custom TCP, Protocol: TCP, Port Range: 8000, Source: Custom, 0.0.0.0/0**

For example, in the browser should work something like: *publicAddressFromAWS:8000*

Extra: <https://kubernetes.io/docs/tutorials/hello-minikube/>