Introduction to Cloud Computing – Exercise 3

Scope: Docker Swarm.

Introduction:

Docker Swarm enables distributed operation of the application environment, including microservices, on many machines. There is a dominant machine 'Manager' and sub-machines called 'Workers'. Follow the instructions to complete the task.

1. Environment preparation:

Import three VM instances (you can find the import files in the Downloads folder). Give each a different name. Before you start the machines, you need to configure the network settings of the machines. To do this, go to Settings -> Network.

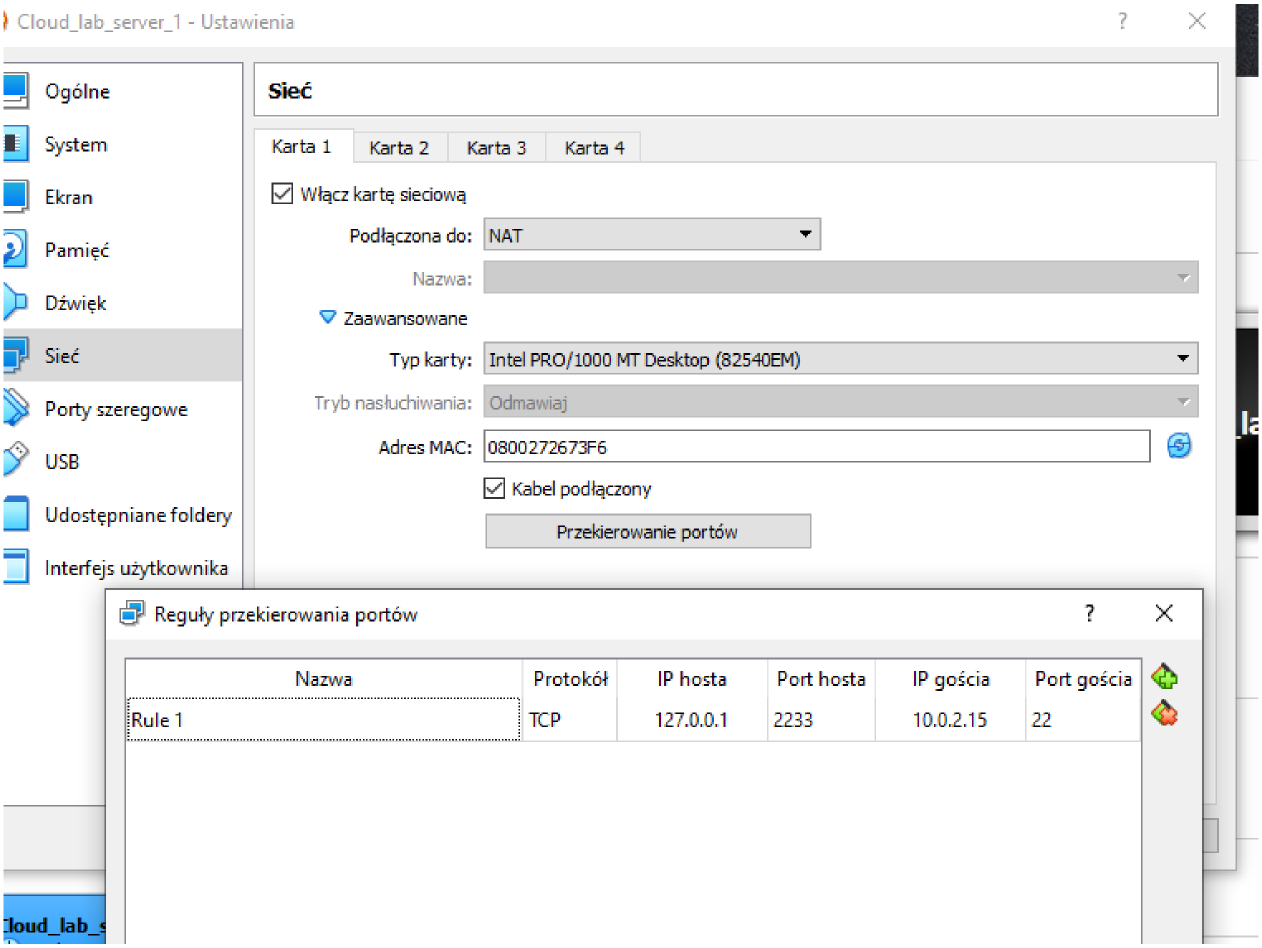
Set the first network card in NAT, and in the advanced settings, enter Port forwarding. This is required to enable ssh connection to our virtual machine.

Add a new rule, TCP protocol, Host IP enter 127.0.0.1, select a port other than 22 (e.g. 2233) and then Guest IP 10.0.2.15 and port 22. In this case, port 22 (i.e. ssh) from the virtual machine is redirected to port 2233 of our machine where Virtualbox is running.

Picture below.

For other machines, set different Host times (e.g. 2244 for the second machine and 2255 for the third machine).

Then go to the Tab 2 tab and there select the Internal network settings. Remember that each machine should have a different MAC address. This setting will allow VMs to communicate.



Sieć

☑ Włącz kartę sieciową

Podłączona do: NAT

Nazwa:

Zaawansowane

Typ karty: Intel PRO/1000 MT Desktop (82540EM)

Tryb nasłuchiwania: Odmawiaj

Adres MAC: 0800272673F6

☑ Kabel podłączony

Reguły przekierowania portów

Nazwa	Protokół	IP hosta	Port hosta	IP gościa	Port gościa
Rule 1	TCP	127.0.0.1	2233	10.0.2.15	22

After starting, you should have 3 machines with internet access and access to each other. At this point, you can start the VMs and log into them via ssh.

To do this, run the cmd command line and then connect to the virtual machines

```
ssh student@127.0.0.1 -p 2233
```

The password for the student account is: student.

In the same way, connect via ssh to other machines, specifying a different port.

The next step is to change the hostname and hosts on our servers. We want to have 3 servers named node1, node2 and node3. To change the hostname type

```
sudo nano /etc/hostname
```

edit name to node1 save file. Do:

```
sudo nano /etc/hosts
```

edit the file renaming from node1. Save the file, restart the machine

```
sudo reboot
```

Do the same for UbuntuServer2, UbuntuServer3 just type node2 and node3.

Result: 3 Linux machines with hostnames: node1, node2, node3

2. Network configuration

In order for the virtual machines to see each other, we must set the appropriate network configuration. For this purpose, we will use the second card selected in the settings of virtual machines and internal network. After starting the machines, we will enter the network settings. Execute the command

```
ip addr show
```

You should see all network interfaces. You will notice that some are disabled. To configure on startup, we need to edit the configuration file. Make changes carefully as wrong changes can damage your network settings. The file we want to edit is in the /etc/netplan folder, to know the name execute

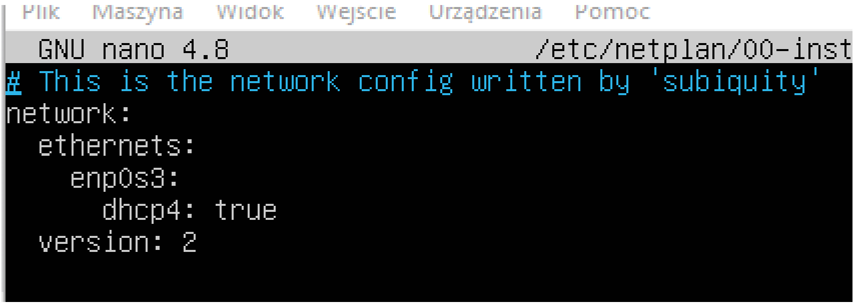
```
ls /ect/netplan/
```

You should see the configuration file name

Then edit the file

```
Sudo nano /ect/netplan/nazwa_pliku.yaml
```

By default, the file should look similar to the dump shown below:



```
  GNU nano 4.8                /etc/netplan/00-inst
# This is the network config written by 'subiquity'
network:
  ethernets:
    enp0s3:
      dhcp4: true
  version: 2
```

We want to make additional settings for another network interface. To do this, add the appropriate settings in the ethernets section (add new ones while keeping the existing ones)

```
enp0s8:
  dhcp4: no
  addresses:
  - 192.168.56.103/24
  gateway4: 192.168.56.1
```

Where in place of enp0s8 we must enter the name of our internal network interface and in place of addresses the address that our machine should have.

After saving the file, check if the configuration is correct

```
$ sudo netplan try
```

If all goes well, next excecute

```
ip addr show
```

You should notice that the web interface is initialized. At this point you can do

```
sudo netplan apply
```

The machine prepared in this way will have the appropriate address of the internal network after starting. Do this for the other machines by giving them a different IP address.

After configuring the machines, check if the machines can see each other with the ping command.

3. Create Docker Swarm:

On UbuntuServer1, type in the console:

```
sudo docker swarm init
```

a token will be generated copy the whole line to notepad

```
sudo docker swarm join –token „token razem z IP:port”
```

Paste the line into the console of Ubuntu Server 2 and UbuntuServer3. On the Ubuntu Server1 machine, type

```
sudo docker node ls
```

read the information displayed see on other machines if you can see node. Note the hostname.

What conclusions can you point to?

4. Create network from overlay

Create a virtual network that will allow communication between services installed on multiple machines:

```
sudo docker network create --driver overlay drupal_network
```

Check if the network has been created

```
sudo docker network ls
```

5. Postgresql db installation

Install the postgres server, pay attention to the name and password, the username is 'postgres', it will be useful later when configuring Drupal.

```
sudo docker service create --name psql --network drupal_network -e
POSTGRES_PASSWORD=mypass postgres
```

Check if the psql service has been installed.

```
sudo docker service ls
sudo docker service ps psql
```

Try to check the logs if you don't see UbuntuServer1 on the other machines:

```
sudo docker container logs psql
```

What conclusions can you point to?

6. Drupal installation

Install Drupal service pay attention to port forwarding.

```
sudo docker service create --name drupal --network drupal_network -p 80:80 drupal
```

Try to write the IP numbers of the individual Linux servers in the browser in 3 separate tabs, you should see the Drupal configuration page, pay attention to the overlay managing address redirection from the 3 machines to the main machine.

Present the result to the teacher.

Additional task

Add port forwarding to access drupal from the machine where the virtual machines are running.