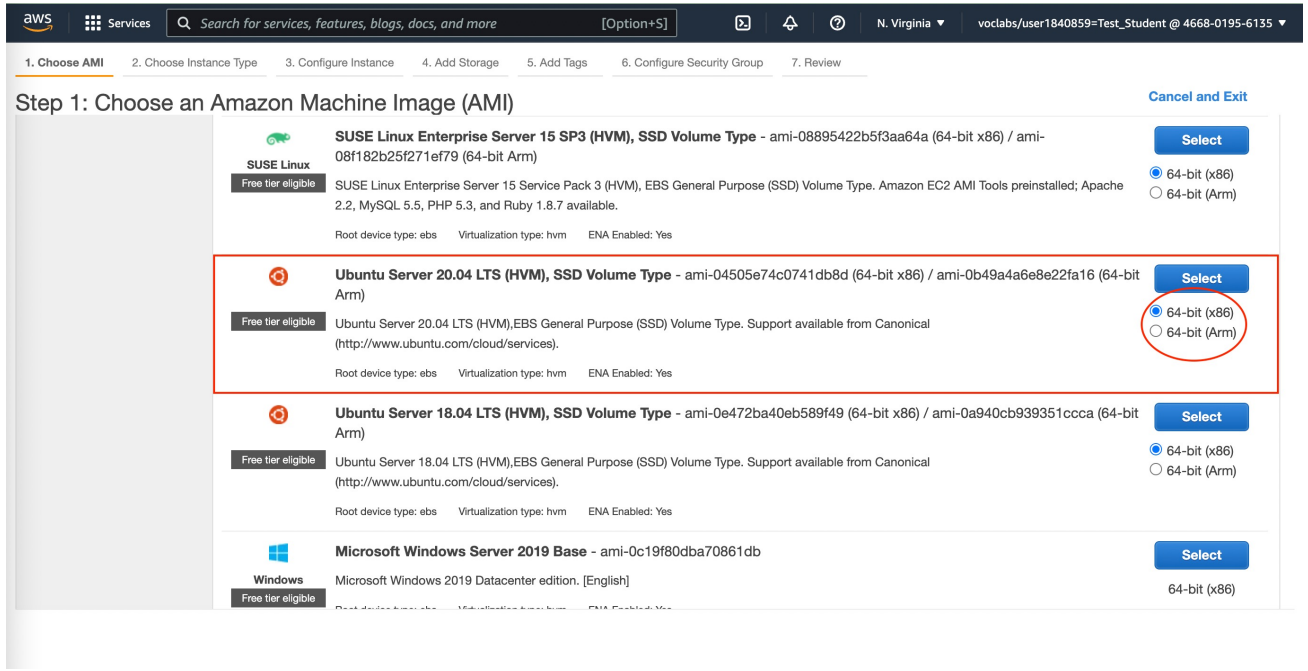
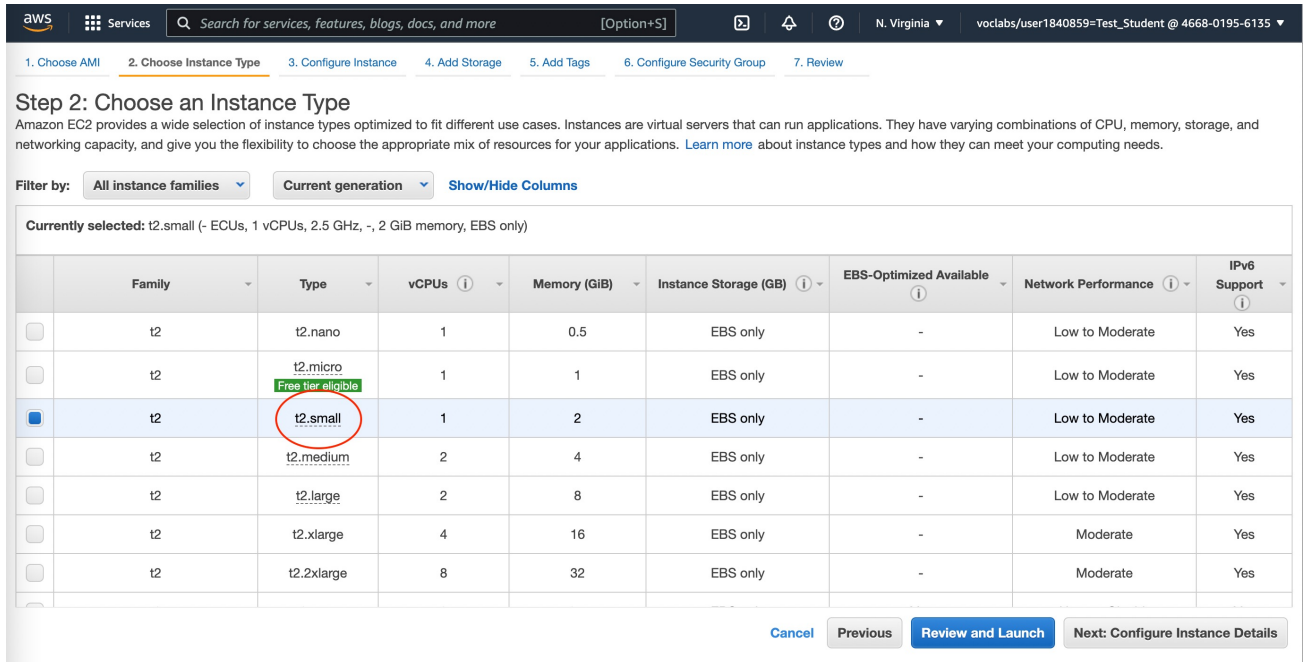
Ne logam pe site-ul AWS si cream o instanta EC2:

Hardware: t2.small

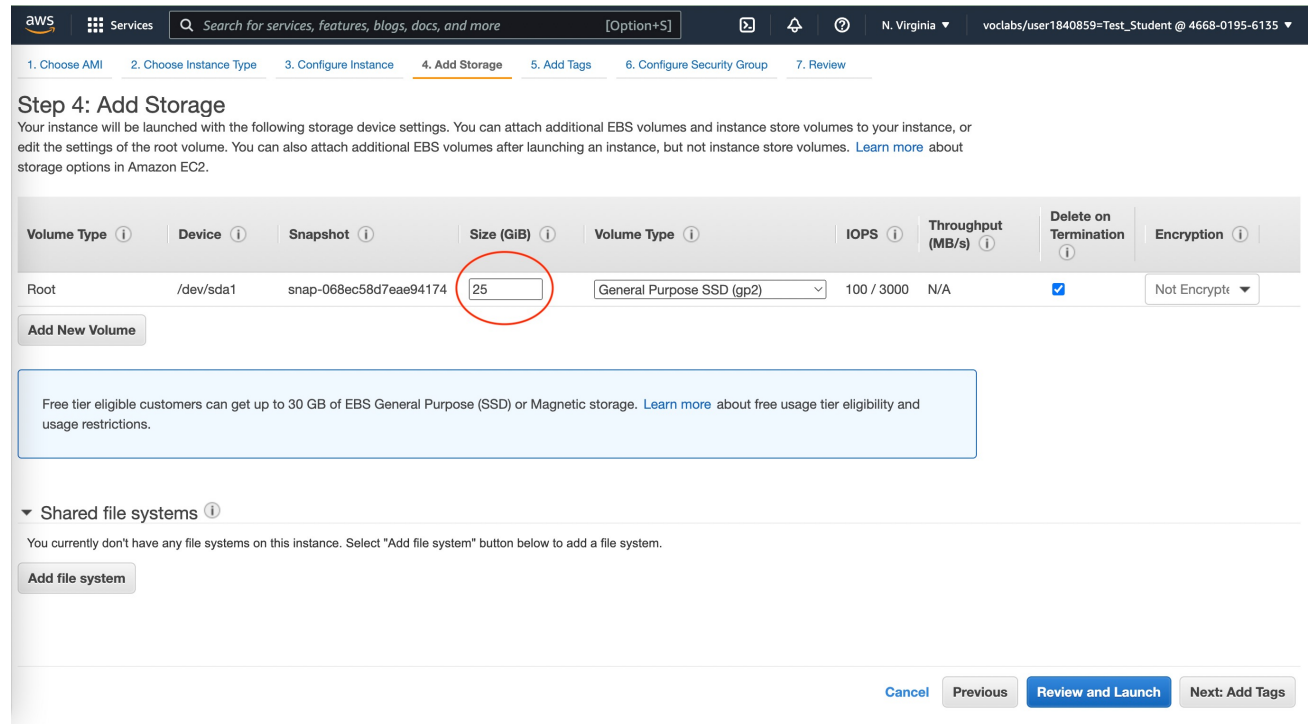
Software: Ubuntu 20.04 / **AMD64 / X86_64**



Avem nevoie de minim 2G RAM asadar alegem modelul **t2.small**

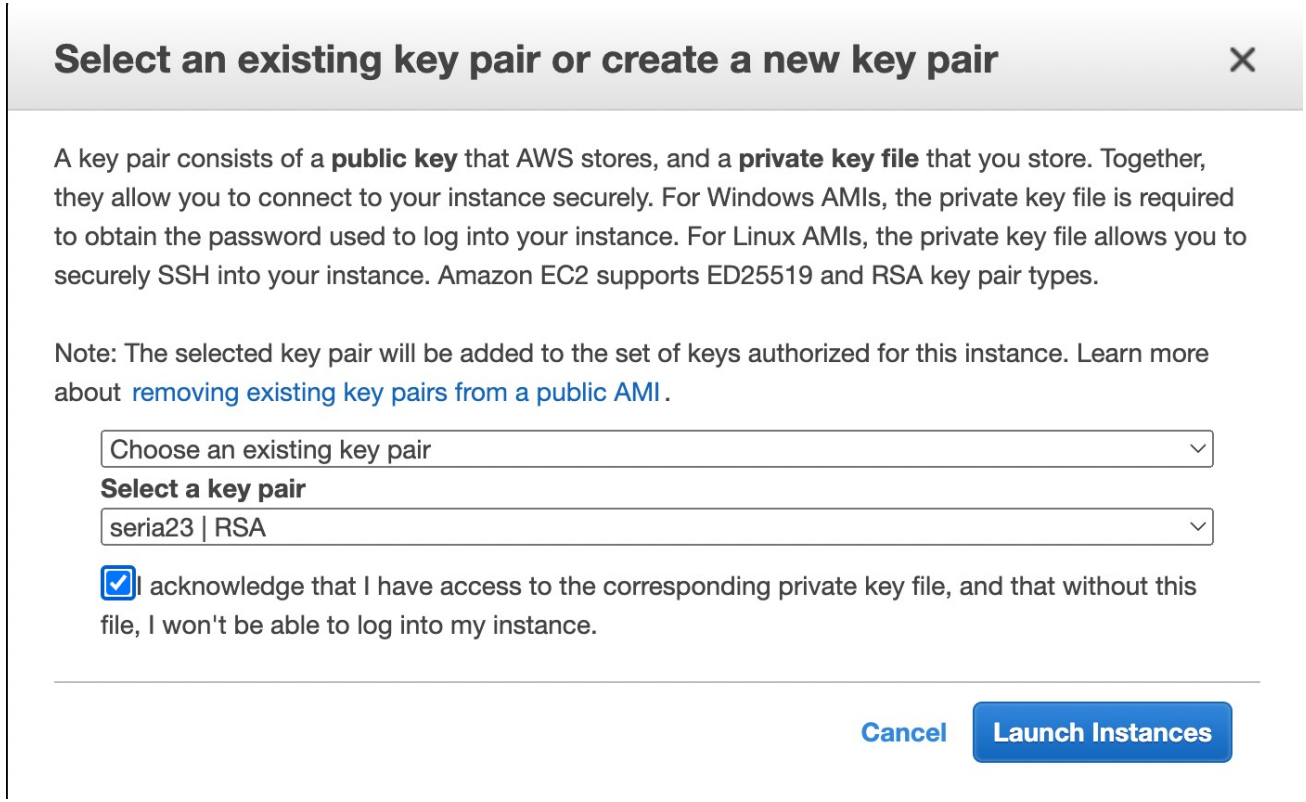


Modificam spatiul pe disk din 10G in **25G**.



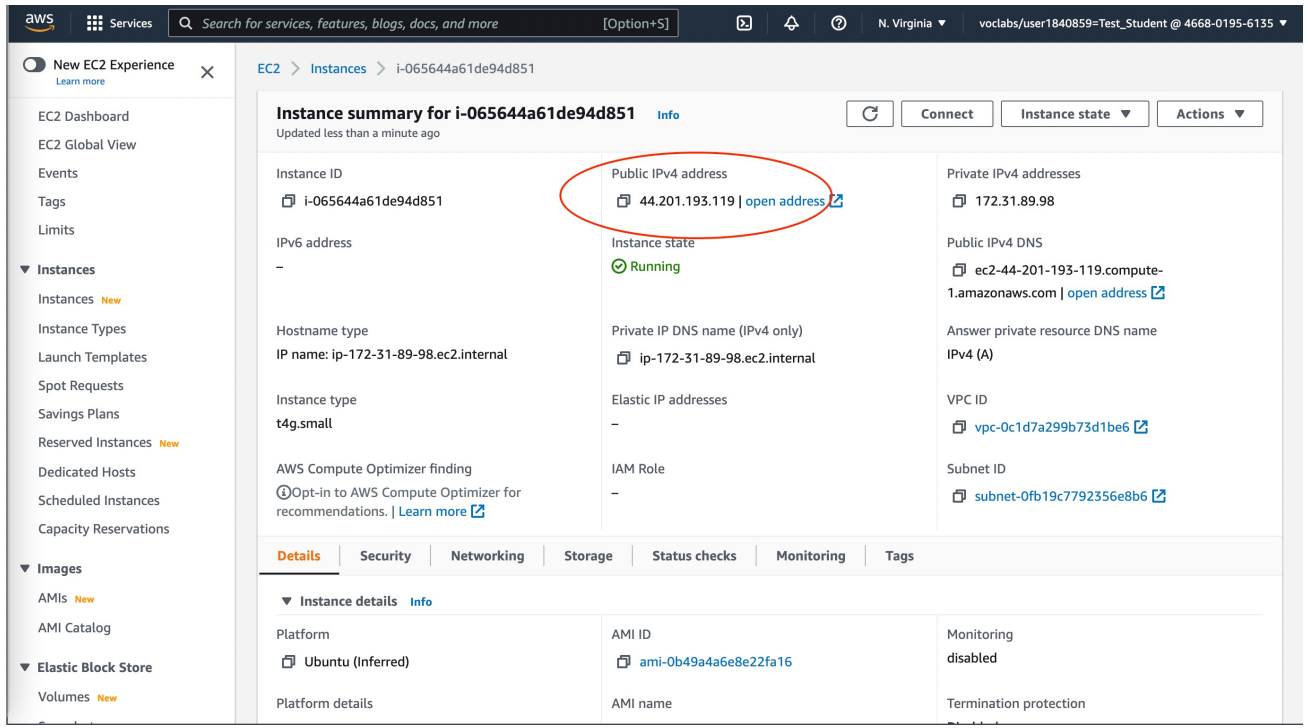
Dupa ce cream instanta, avem nevoie de o cheie privata pentru a accesa serverul via SSH.

Se poate folosi o cheie existenta daca ati mai creat alte instante anterior. In acest exemplu, se foloseste cheia privata **seria23.pem**



Dupa ce am creat serverul, ne conectam via SSH pe adresa lui IP (in acest exemplu **44.201.193.119**) folosind cheia privata definita anterior (in acest caz **seria23.pem**) si userul **ubuntu** (creat automat).

ssh -i serie23.pem ubuntu@44.201.193.119



Dupa ce ne conectam pe server via ssh, instalam docker:

sudo su -

apt-get update

apt install -y docker docker-compose

```
1. ne conectam pe server via ssh
ssh -i .ssh/seria23.pem ubuntu@44.201.193.119
2. ne facem root
sudo su -
3. instalam docker
apt-get update
apt install -y docker docker-compose
```

Tragem imaginea docker cu topologia de laborator:

docker pull pliviu/ubmi:AMD64

docker image ls

docker run -it -d <image ID>

docker ps

docker exec -it <container ID> bash

cd /opt/ubmi/

./start.sh L1

```
1.  tragem imaginea docker
docker pull pliviu/ubmi:AMD64

2. aflam ID-ul imaginii docker
docker image ls
(IMAGE ID: 467e4bb96012)

3. cream un container folosind imaginea
docker run -it -d 467e4bb96012

4. aflam ID-ul containerului
docker ps
(CONTAINER ID: 6dc0feb59731)

5. ne atasam la container (repetam acest pas in mai multe console)

6. tragem automat topologia de laborator
cd /opt/ubmi
./start.sh L1
(asteptam cateva zeci de secunde pentru ca se downloadeaza ~ 150Mb)
```

Pornim pe rand fiecare device si ne conectam la el din alta consola. Conectarea se va face cu comanda ***telnet localhost <port>*** iar portul este cel de pe ultima coloana.

Consola principala, cea din figura alaturata, va ramane tot timpul deschisa.

⇒ start R1

⇒ start R2

⇒ start TIER1-ISP

⇒ ……

```
=> list
Name       Type       State      Server           Console
R1         7200       stopped    localhost:7200   5101
R2         7200       stopped    localhost:7200   5102
R3         7200       stopped    localhost:7200   5103
R4         3725       stopped    localhost:7200   5104
TIER1-ISP  3725       stopped    localhost:7200   5110
SW1        3725       stopped    localhost:7200   6101
SW2        3725       stopped    localhost:7200   6102
SW         ETHSW      always on  localhost:7200   n/a
=>
```

Pentru conectarea la PC-urile virtuale facem ***telnet localhost 7300***

Folosind 1,2,3 … 6 ne mutam de pe un PC pe altul.

Daca vrem sa parasim aceasta consola folosim comanda ***disconnect***. Comanda quit o sa opreasca toate cele 6 PC-uri si nu ne dorim asta.

```
root@327ba2aa39f0:/opt/ubmi# telnet localhost 7300
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.

Welcome to Virtual PC Simulator, version 0.5b2
Dedicated to Daling.
Build time: Jun 23 2018 13:14:38
Copyright (c) 2007-2014, Paul Meng (mirnshi@gmail.com)
All rights reserved.

VPCS is free software, distributed under the terms of the "BSD" licence.
Source code and license can be found at vpcs.sf.net.
For more information, please visit wiki.freecode.com.cn.

Press '?' to get help.

Executing the startup file




Checking for duplicate address...
PC6 : 1.1.1.1 255.255.255.252 gateway 1.1.1.2

PC1[1]> disconnect

PC1[1]>
Good-bye
Connection closed by foreign host.
```

Topologiile noi se vor incarca astfel:

cd /opt/ubmi

./start.sh Lx (unde x = numar)

```
root@327ba2aa39f0:/opt/ubmi#
root@327ba2aa39f0:/opt/ubmi# ./start.sh
Syntax: ./start <lab number>
Example: ./start.sh L1
root@327ba2aa39f0:/opt/ubmi#
```

Diagrama pentru L1 este cea folosita in GNS3 iar interfetele si legaturile dintre PC-uri, switch-uri si routere sunt identice.