Instances/VMs

M. F. L. Author
A01748161@itesm.mx
ITESM CSF

***Abstract*— For this report, a connection to a virtual machine and an instance via SSH was done. A file was also created and copied via scp.**

***Keywords*— Instance, EC2, macOS, terminal, SSH, scp**

I. Introduction

Virtualization, according to T-Systems Iberia, is able to create a virtual version of storage, network resources or a whole operating system in a computer, and one of their most common applications is that of executing an OS within another via a virtual machine.

A virtual machine works as an emulator of components and resources of a physical computer system, as it can run an operating system and execute it as an existing element of the physical plane.

A. Materials

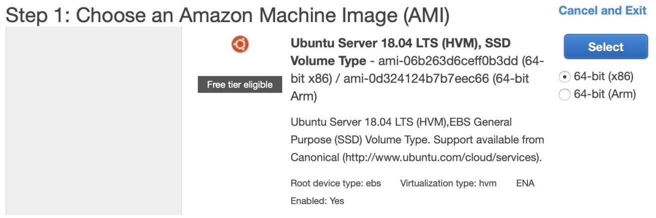
- MacBook Pro, with a 2.2 GHz 6-Core Intel Core i7 processor and macOS Catalina 10.15.6
- *Amazon Web Services (AWS) Educate* membership
  - EC2 Instance

II. Development

A. Create an Instance in AWS

The first step taken to create the instance was to select the specific type; in this case, it was an EC2 VM.

After accessing the EC2 panel and clicking on the *Launch Instance* button, the following configurations were created:



Step 1: Choose an Amazon Machine Image (AMI)

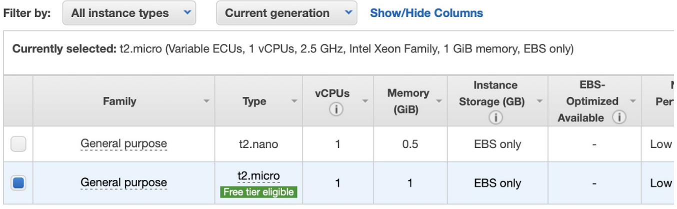
Ubuntu Server 18.04 LTS (HVM), SSD Volume Type - ami-06b263d6ceff0b3dd (64-bit x86) / ami-0d324124b7b7eec66 (64-bit Arm)

Ubuntu Server 18.04 LTS (HVM),EBS General Purpose (SSD) Volume Type. Support available from Canonical (http://www.ubuntu.com/cloud/services).

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

● 64-bit (x86)
○ 64-bit (Arm)

Figure 1. Choosing Ubuntu Server 18.04.

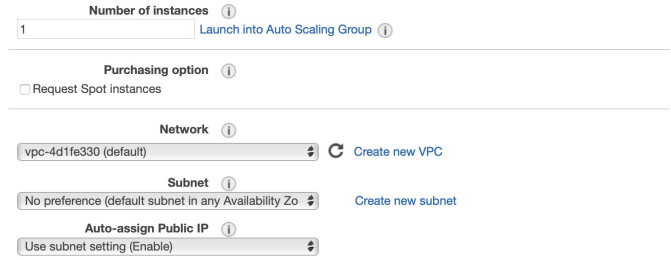


Filter by: All instance types | Current generation

Currently selected: t2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Per
□	General purpose	t2.nano	1	0.5	EBS only	-	Low
☑	General purpose	t2.micro Free tier eligible	1	1	EBS only	-	Low

Figure 2. Choosing an instance type.



Number of instances: 1

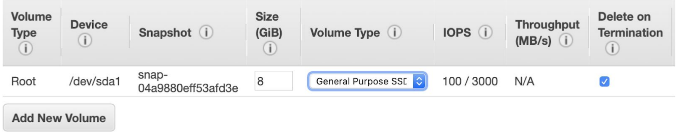
Purchasing option: □ Request Spot instances

Network: vpc-4d1fe330 (default)

Subnet: No preference (default subnet in any Availability Zo

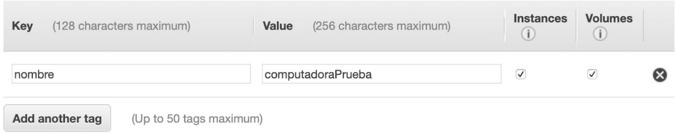
Auto-assign Public IP: Use subnet setting (Enable)

Figure 3. Configuring instance details (default settings).



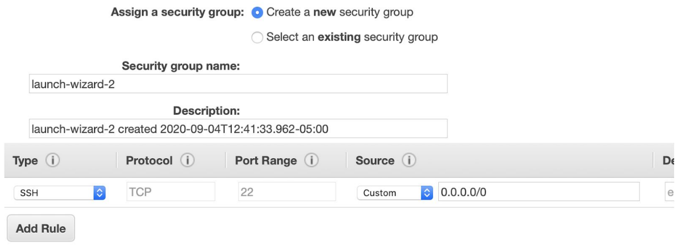
Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination
Root	/dev/sda1	snap-04a9880eff53afd3e	8	General Purpose SSD	100 / 3000	N/A	☑

Figure 4. Adding storage.



Key (128 characters maximum)	Value (256 characters maximum)	Instances	Volumes
nombre	computadoraPrueba	☑	☑

Figure 5. Adding tags.



Assign a security group: ● Create a **new** security group
○ Select an **existing** security group

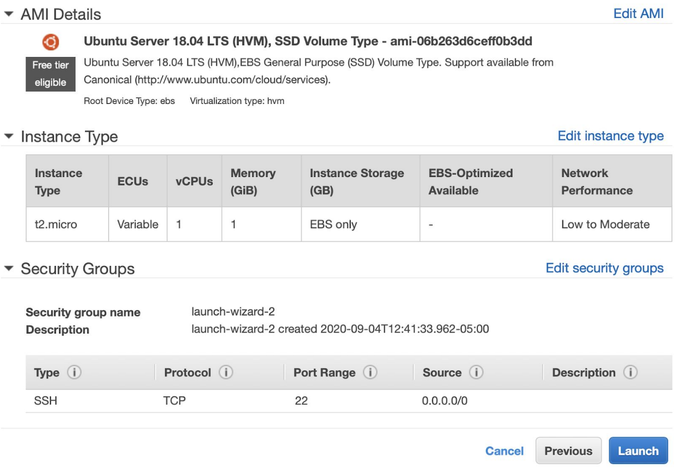
Security group name: launch-wizard-2

Description: launch-wizard-2 created 2020-09-04T12:41:33.962-05:00

Type	Protocol	Port Range	Source
SSH	TCP	22	Custom 0.0.0.0/0

Figure 6. Configuring security groups (default).

[1] This report was sent for revision on September 4th, 2020.
The author is with the Instituto de Estudios Superiores Monterrey, Santa Fe.



AMI Details

Ubuntu Server 18.04 LTS (HVM), SSD Volume Type - ami-06b263d6ceff0b3dd

Ubuntu Server 18.04 LTS (HVM),EBS General Purpose (SSD) Volume Type. Support available from Canonical (http://www.ubuntu.com/cloud/services).

Root Device Type: ebs Virtualization type: hvm

Instance Type

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	Variable	1	1	EBS only	-	Low to Moderate

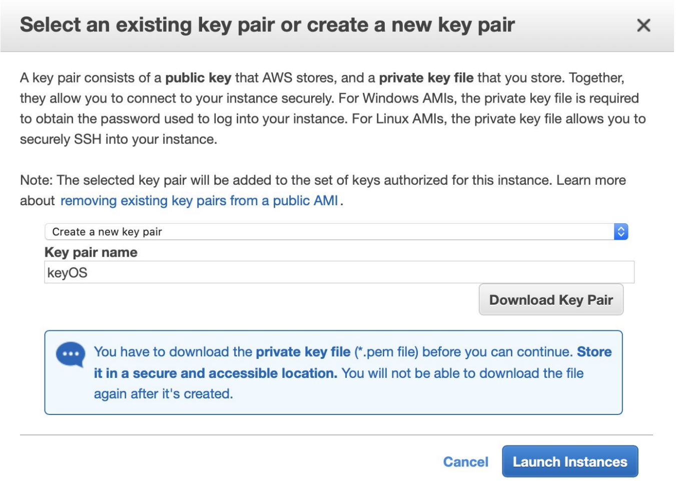
Security Groups

Security group name launch-wizard-2

Description launch-wizard-2 created 2020-09-04T12:41:33.962-05:00

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	0.0.0.0/0	

Figure 7. Reviewing instance launch.



Select an existing key pair or create a new key pair

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about removing existing key pairs from a public AMI.

Create a new key pair

Key pair name

keyOS

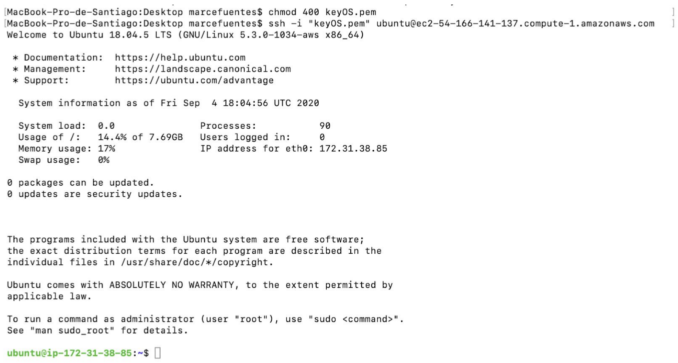
You have to download the **private key file** (*.pem file) before you can continue. **Store it in a secure and accessible location.** You will not be able to download the file again after it's created.

Figure 8. Creating a new key pair.

After performing said steps, the instance was up and running with the *IPv4 Public IP "54.166.141.137"* and the *Public DNS (IPv4) "ec2-54-166-141-137.compute-1.amazonaws.com"*.

B. Connect to the Instance via SSH

In order to connect to the instance using the MacBook's own terminal, the command *chmod 400 keyOS.pem* was first run to enable the needed permissions for the key pair to actually work. After giving the necessary authorization to the keys, the command *ssh -i "keyOS.pem" ubuntu@ec2-54-166-141-137.compute-1.amazonaws.com* was run and the connection was made.

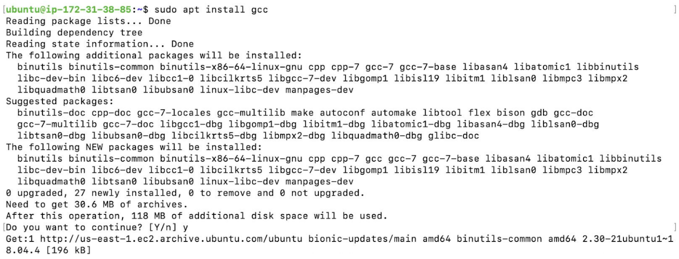


```
MacBook-Pro-de-Santiago:Desktop marcefuentes$ chmod 400 keyOS.pem
MacBook-Pro-de-Santiago:Desktop marcefuentes$ ssh -i "keyOS.pem" ubuntu@ec2-54-166-141-137.compute-1.amazonaws.com
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 5.3.0-1034-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Fri Sep  4 18:04:56 UTC 2020

  System load:  0.0               Processes:           90
  Usage of /:   14.4% of 7.69GB   Users logged in:     0
  Memory usage: 17%               IP address for eth0: 172.31.38.85
  Swap usage:   0%

0 packages can be updated.
0 updates are security updates.



The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-38-85:~$ 
```

Figure 9. Connection via SSH.

C. Install gcc

To perform the installation of the C compiler, the command used was *sudo apt install gcc.*

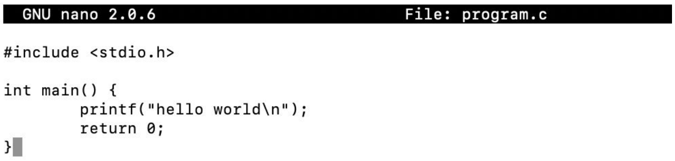


```
ubuntu@ip-172-31-38-85:~$ sudo apt install gcc
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  binutils binutils-common binutils-x86-64-linux-gnu cpp cpp-7 gcc-7 gcc-7-base libasan4 libatomic1 libbinutils
  libc-dev-bin libc6-dev libcc1-0 libcilkrts5 libgcc-7-dev libgomp1 libisl19 libitm1 liblsan0 libmpc3 libmpx2
  libquadmath0 libtsan0 libubsan0 linux-libc-dev manpages-dev
Suggested packages:
  binutils-doc cpp-doc gcc-7-locales gcc-multilib make autoconf automake libtool flex bison gdb gcc-doc
  gcc-7-multilib gcc-7-doc libgcc1-dbg libgomp1-dbg libitm1-dbg libatomic1-dbg libasan4-dbg liblsan0-dbg
  libtsan0-dbg libubsan0-dbg libcilkrts5-dbg libmpx2-dbg libquadmath0-dbg glibc-doc
The following NEW packages will be installed:
  binutils binutils-common binutils-x86-64-linux-gnu cpp cpp-7 gcc gcc-7 gcc-7-base libasan4 libatomic1 libbinutils
  libc-dev-bin libc6-dev libcc1-0 libcilkrts5 libgcc-7-dev libgomp1 libisl19 libitm1 liblsan0 libmpc3 libmpx2
  libquadmath0 libtsan0 libubsan0 linux-libc-dev manpages-dev
0 upgraded, 27 newly installed, 0 to remove and 0 not upgraded.
Need to get 30.6 MB of archives.
After this operation, 118 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-updates/main amd64 binutils-common amd64 2.30-21ubuntu1~1
8.04.4 [196 kB]
```

Figure 10. Installing the compiler.

D. Upload a C Source Code using scp

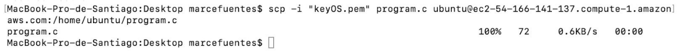
Before actually uploading any file to the AWS instance, a C source code file was created using the MacBook's terminal. This was done by running the line *nano -w program.c*, accessing the editor mode, and directly writing and essentially creating the file.



```
  GNU nano 2.0.6                         File: program.c

#include <stdio.h>

int main() {
        printf("hello world\n");
        return 0;
}
```

Figure 11. Writing program.c

To upload the file, a similar command to the one used to connect to the was used: *scp -i "keyOS.pem" program.c ubuntu@ec2-54-166-141-137.compute-1.amazonaws.com:/home/ubuntu/program.c*

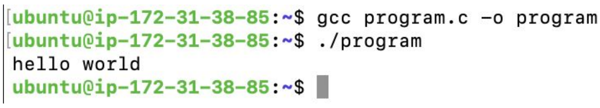


```
MacBook-Pro-de-Santiago:Desktop marcefuentes$ scp -i "keyOS.pem" program.c ubuntu@ec2-54-166-141-137.compute-1.amazon
aws.com:/home/ubuntu/program.c
program.c                                                                     100%   72     0.6KB/s   00:00
MacBook-Pro-de-Santiago:Desktop marcefuentes$ 
```

Figure 12. Uploading the file.

E. Compile the Program and Run it in the Cloud

The very last steps were to compile and execute the code. This was done with the commands *gcc program.c -o program* and *./program*, respectively.



```
[ubuntu@ip-172-31-38-85:~$ gcc program.c -o program
[ubuntu@ip-172-31-38-85:~$ ./program
hello world
ubuntu@ip-172-31-38-85:~$ █
```

Figure 13. Correct compilation and execution.

III. Conclusion

The whole process of launching an instance, configuring it correctly, connecting to it via a physical computer, and finally uploading files to it was actually easier than I expected. Thanks to the previous practice where I got to know how to move around the terminal with much more confidence, I can say that for this second report I felt like I had a better understanding of the commands that were used. For future references, I would like to try uploading and executing much more complex files rather than a simple *hello world* program, as I believe that it can be useful to use a virtual machine as a sort of “test-zone.”

References

T-Systems Iberia. (March 26th, 2018). *¿Para qué sirve una máquina virtual?* Recovered on July 18th, 2020 from https://www.t-systemsblog.es/para-que-sirve-una-maquina-virtual/