

Universidad Modelo



Escuela de Ingeniería.

Carrera: Ingeniería en Desarrollo de Tecnología y Software

Asignatura: Fundamentos de La nube

Nombre de la Actividad: Practica AWS

Fecha de Entrega: marzo 11 del 2021

Nombre del Alumno: Sinuhe Arturo Evia Campos



Image 1-
Instance being created for the usage of the test.

```
ubuntu@ip-125-26-10-122:~$ sudo -s
root@ip-125-26-10-122:/home/ubuntu# cd ..
root@ip-125-26-10-122:/home# git clone https://github.com/
sinuheArturo/FundamentosNube.git
Cloning into 'FundamentosNube'...
Username for 'https://github.com': sinuheArturo
Password for 'https://sinuheArturo@github.com':
remote: Enumerating objects: 148, done.
remote: Counting objects: 100% (148/148), done.
remote: Compressing objects: 100% (65/65), done.
Receiving objects: 15% (23/148), 100.01 KiB | 185.00 KiB/
Receiving objects: 16% (24/148), 100.01 KiB | 185.00 KiB/
Receiving objects: 17% (26/148), 100.01 KiB | 185.00 KiB/
Receiving objects: 18% (27/148), 100.01 KiB | 185.00 KiB/
Receiving objects: 19% (29/148), 100.01 KiB | 185.00 KiB/
Receiving objects: 20% (30/148), 100.01 KiB | 185.00 KiB/
```

Image 2-
In this instance, we are currently creating the super user (or root)
And we perform a “gitclone” action

```
Receiving objects: 100% (148/148), 1.99 MiB | 2.18 MiB/s,
done.
Resolving deltas: 100% (81/81), done.
root@ip-125-26-10-122:/home# ls
FundamentosNube  ubuntu
root@ip-125-26-10-122:/home#
```

Image 3- checking the files that are indeed in the ubuntu server.

```
root@ip-125-26-10-122:/home# vi installdocker.sh
```

Image 4- creating the file install docker.

```
curl \
    gnupg-agent \
    software-properties-common

curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt
sudo apt-key fingerprint 0EBFCD88

sudo add-apt-repository \
    "deb [arch=amd64] https://download.docker.com/linux/ubuntu
    $(lsb_release -cs) \
    stable"

sudo apt-get update

sudo apt-get install docker-ce docker-ce-cli containerd.io

#install docker compose

sudo curl -L
"https://github.com/docker/compose/releases/download/1.28.5/docker
name -s)-$(uname -m)" -o /usr/local/bin/docker-compose
sudo chmod +x /usr/local/bin/docker-compose
sudo ln -s /usr/local/bin/docker-compose /usr/bin/docker-compose
```

Image 5-

This file is used to install the necessary libraries to install docker and usage of docker.

```
<.sh" [New] 97L, 929C written
root@ip-125-26-10-122:/home# chmod +755 installdocker.sh
root@ip-125-26-10-122:/home# ls
FundamentosNube  installdocker.sh  ubuntu
root@ip-125-26-10-122:/home#
```

Image 6- We give the necessary permissions in order to use the execution command in order to move forward.

```
root@ip-125-26-10-122 /home/ubuntu# docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS
TS            NAMES
```

Image 7- shows the verification that let us see if the docker compose is there, giving us a green light that we successfully installed the required items.

```
ose up -d
Creating network "web_default" with the default driver
Creating web_measurementApp_1 ... done
root@ip-125-26-10-122: /home/FundamentosNube/web#
```

Image 8-

Here we clone the address by using the action “docker compose”

Manual snapshots ?

You can create a snapshot to back up your instance, its system disk, and attached disks.

+ Create snapshot

Give your snapshot a name.

Cancel  Create 


Image 9-

We created the snapshot that was solicited in the instructions.

Manual snapshots ?

You can create a snapshot to back up your instance, its system disk, and attached disks.

+ Create snapshot

>  **March 10, 2021 - 9:54 PM**

"ServerWeb1-copia1"

Showing 1 of 1 snapshots

Image 10-

The result of the action that was made for the snapshot.

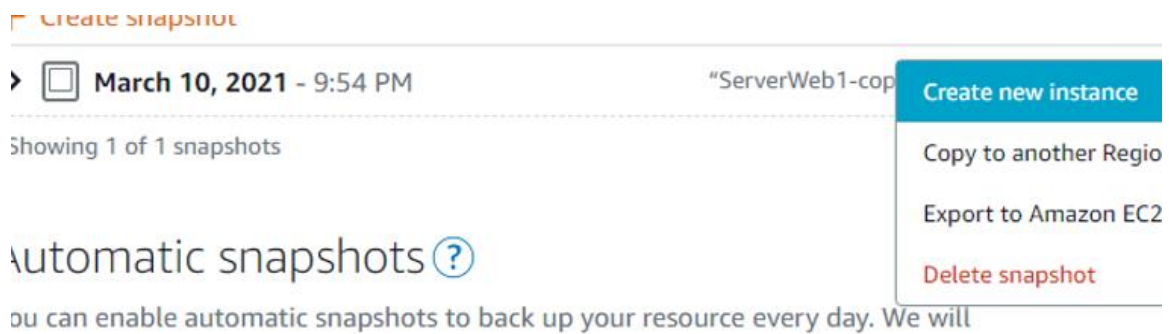


Image 11- We create a new instance from the snapshot.



Image 12-

Here we created successfully both instances, one is the original, the second is created from the snapshot.

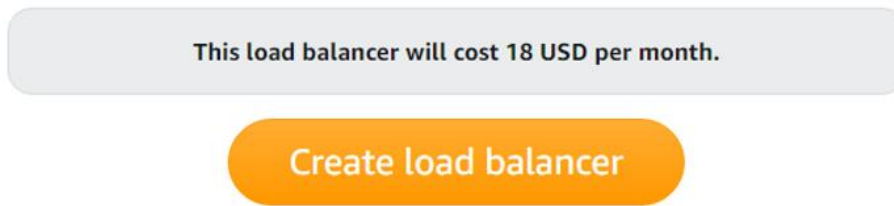


Image 13-

We are creating a load balancer.

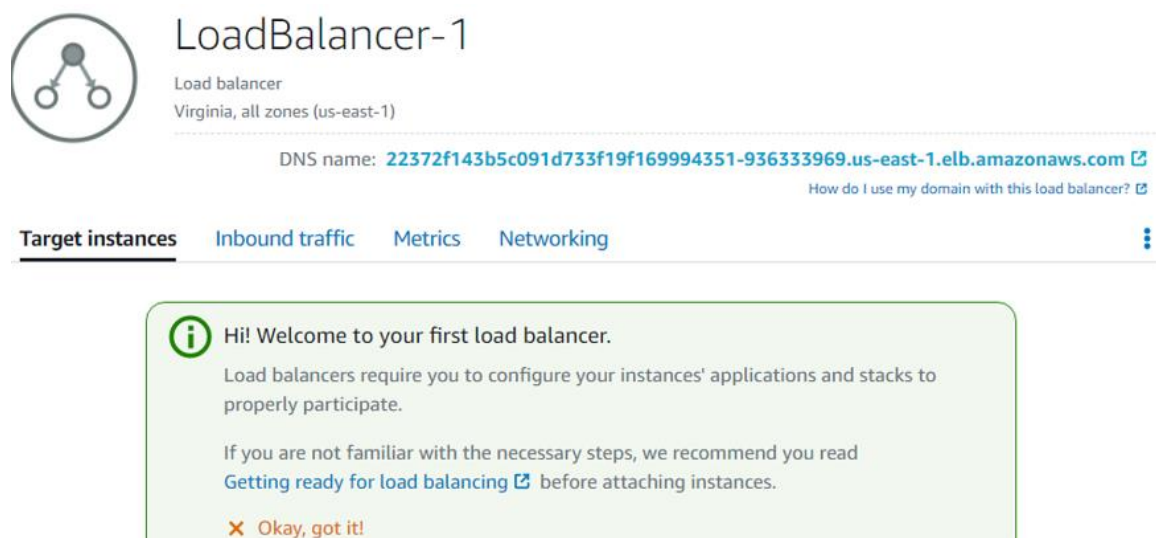



Image 14-

Created the balancer succesfully.

Target instances

Traffic will be evenly distributed to the following instances:

 **Attach another**
All available instances in **Virginia** attached

**ServerWeb2**
512 MB RAM, 1 vCPU, 20 GB SSD
Ubuntu
⌘ Attaching...

**ServerWeb1**
512 MB RAM, 1 vCPU, 20 GB SSD
Ubuntu
⌘ Attaching...

Image 15-

Here we attached both instances from the server web in the load balancer.

lightsailaws.amazon.com/ls/remote/us-east-1/instances/ServerWeb2/terminal?protocol=ssh

```
top - 04:10:00 up 6 min, 1 user, load average: 0.04, 0.
Tasks: 104 total, 1 running, 103 sleeping, 0 stopped,
%Cpu(s): 0.0 us, 0.0 sy, 0.0 ni,100.0 id, 0.0 wa, 0.
MiB Mem : 475.4 total, 5.4 free, 239.9 used,
MiB Swap: 0.0 total, 0.0 free, 0.0 used.
```

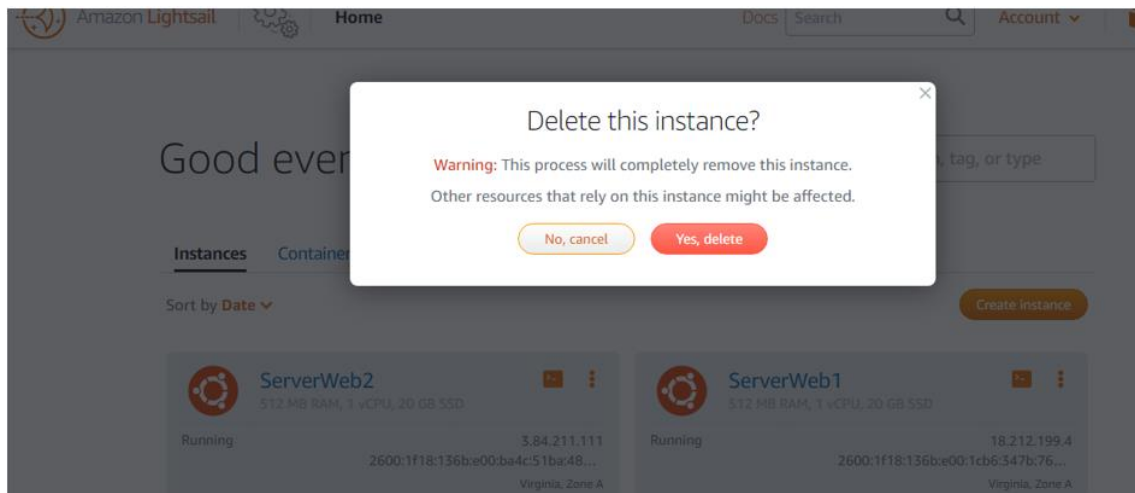
PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU
1	root	20	0	167360	11024	8076	S	0.0
2	root	20	0	0	0	0	S	0.0
3	root	0	-20	0	0	0	I	0.0
4	root	0	-20	0	0	0	I	0.0
6	root	0	-20	0	0	0	I	0.0
8	root	20	0	0	0	0	I	0.0
9	root	0	-20	0	0	0	I	0.0
10	root	20	0	0	0	0	S	0.0
11	root	20	0	0	0	0	I	0.0
12	root	rt	0	0	0	0	S	0.0
13	root	20	0	0	0	0	S	0.0
14	root	20	0	0	0	0	S	0.0
15	root	0	-20	0	0	0	I	0.0
16	root	20	0	0	0	0	S	0.0
17	root	20	0	0	0	0	S	0.0
18	root	20	0	0	0	0	S	0.0
19	root	20	0	0	0	0	S	0.0
20	root	20	0	0	0	0	S	0.0
21	root	20	0	0	0	0	S	0.0
22	root	0	-20	0	0	0	I	0.0

lightsailaws.amazon.com/ls/remote/us-east-1/instances/ServerWeb1/terminal?protocol=ssh

```
top - 04:09:59 up 37 min, 2 users, load average: 0.01,
Tasks: 113 total, 1 running, 112 sleeping, 0 stopped,
%Cpu(s): 0.0 us, 0.0 sy, 0.0 ni,100.0 id, 0.0 wa, 0.
MiB Mem : 475.4 total, 30.0 free, 265.4 used,
MiB Swap: 0.0 total, 0.0 free, 0.0 used.
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU
12120	root	20	0	64160	18548	5144	S	0.3
1	root	20	0	168472	9712	5408	S	0.0
2	root	20	0	0	0	0	S	0.0
3	root	0	-20	0	0	0	I	0.0
4	root	0	-20	0	0	0	I	0.0
6	root	0	-20	0	0	0	I	0.0
9	root	0	-20	0	0	0	I	0.0
10	root	20	0	0	0	0	S	0.0
11	root	20	0	0	0	0	I	0.0
12	root	rt	0	0	0	0	S	0.0
13	root	20	0	0	0	0	S	0.0
14	root	20	0	0	0	0	S	0.0
15	root	0	-20	0	0	0	I	0.0
16	root	20	0	0	0	0	S	0.0
17	root	20	0	0	0	0	S	0.0
18	root	20	0	0	0	0	S	0.0
19	root	20	0	0	0	0	S	0.0
20	root	20	0	0	0	0	S	0.0
21	root	20	0	0	0	0	S	0.0
22	root	0	-20	0	0	0	I	0.0

Image 16- here we used a top command in Linux, let us see the work load of both servers, here we separated the loads on two servers.



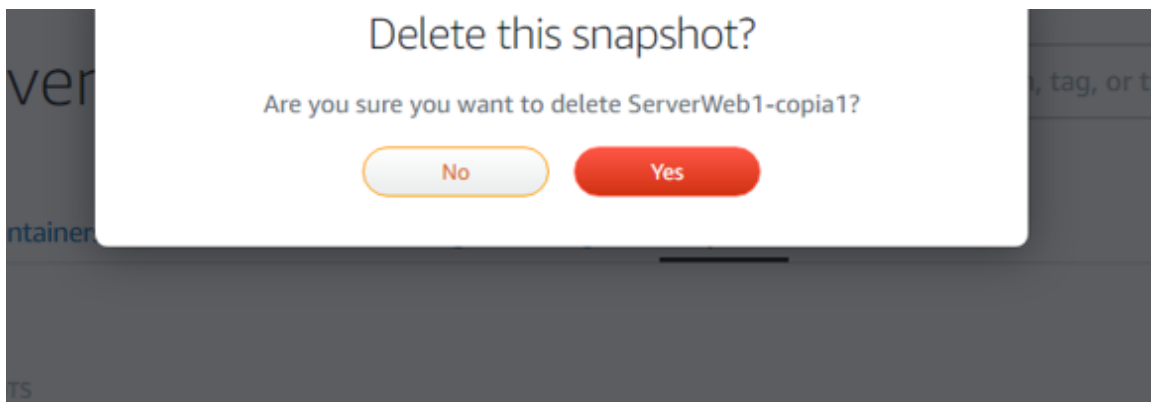
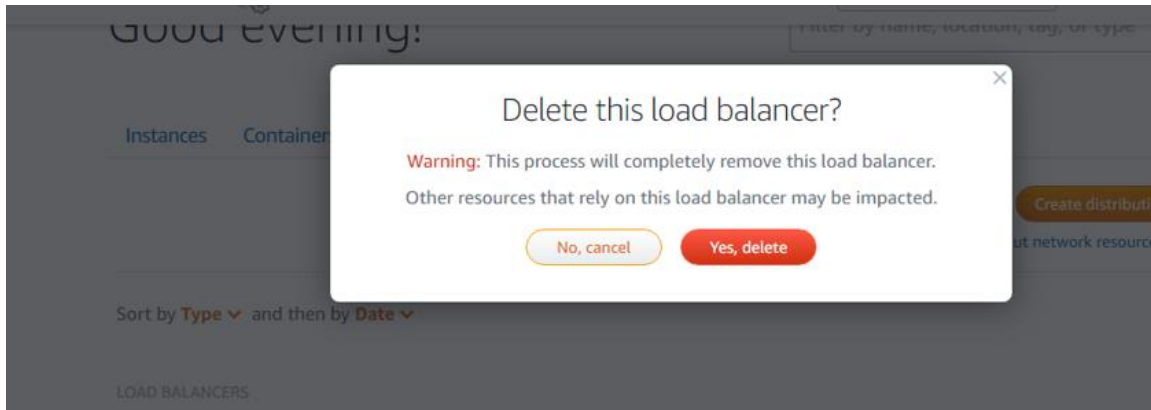


Image 17-

Here we simply delete every instance, secure the termination of load balancer and snapshots to secure that we are charged more money in the end of the month.