

TAREA #1: AMAZON LIGHTSAIL INSTANCE

RODRIGO YERENA HERRERA

UNIVERSIDAD MODELO | ESCUELA DE INGENIERÍA | DTS | 5TO SEMESTRE | ADMINISTRACIÓN DE
BASE DE DATOS | ING. ALFREDO JOSÉ BOLIO DOMÍNGUEZ

Objetivo de la práctica

Crear una instancia de Amazon Lightsail para poder montar un contenedor a través de Docker, poder administrar una base de datos y bloquear a ciertas direcciones IP para evitar el acceso al mundo exterior.

Desarrollo de la práctica:

Primero se debe acceder a la página de “Amazon Web Services” y se accede a la consola de Lightsail a través de una cuenta existente de Amazon.


Good afternoon!

Filter by name, location, tag, or type

Instances Containers Databases Networking Storage Snapshots

Sort by Date ▾

Create instance

**dbClase**
512 MB RAM, 1 vCPU, 20 GB SSD


Running


18.218.124.166
Ohio, Zone A

Una vez accedido a la consola, se procede a crear una instancia nueva: Seleccionaremos la opción de Linux/Unix con la opción de blueprint “OS Only” con Ubuntu.

Pick your instance image ?


Select a platform


 **Linux/Unix**
26 blueprints


 **Microsoft Windows**
4 blueprints


Select a blueprint


Apps + OS **OS Only**


 **Amazon Linux 2**
2.0.20201126.0


 **Amazon Linux**
2018.03.0.202...


 **Ubuntu**
20.04 LTS


 **Ubuntu**
18.04 LTS


 **Ubuntu**
16.04 LTS


 **Debian**
10.5

 **Debian**
9.5

 **Debian**
8.7

 **FreeBSD**
12.1

 **openSUSE**
15.1

 **CentOS**
7 1901-01

Ubuntu 20.04 LTS

Ubuntu 20.04 LTS - Focal. Lean, fast and powerful, Ubuntu Server delivers services reliably, predictably and economically. It is the perfect base on which to build your instances. Ubuntu is free and will always be, and you have the option to get support and Landscape from Canonical.

[Learn more about Ubuntu on the AWS Marketplace](#)

Identify your instance

Your Lightsail resources must have unique names.

dbClaseRodrigoYH × 1

TAGGING OPTIONS

Use tags to filter and organize your resources in the Lightsail console. Key-value tags can also be used to organize your billing, and to control access to your resources.
[Learn more about tagging.](#)

Key-only tags ?

+ Add key-only tags

Key-value tags ?

+ Add key-value tag

Create instance

Your use of AWS services is subject to the [AWS Customer Agreement](#).

Le damos un nombre a nuestra instancia, en mi caso le he puesto “dbClaseRodrigoYH” y finalmente le damos click a crear instancia para que esta se construya.

Good afternoon!

Filter by name, location, tag, or type

Instances Containers Databases Networking Storage Snapshots

Sort by Date

Create Instance

dbClaseRodrigoYH	dbClase
512 MB RAM, 1 vCPU, 20 GB SSD	512 MB RAM, 1 vCPU, 20 GB SSD
Pending	Running
18.189.193.255 Ohio, Zone A	18.218.124.166 Ohio, Zone A

Esperamos a que se construya nuestra instancia.

Una vez que haya finalizado el proceso de construcción, accedemos a la instancia. Le vamos a dar click en “Connect using SSH” para poder abrir la consola de comandos.

dbClaseRodrigoYH

512 MB RAM, 1 vCPU, 20 GB SSD
Ubuntu
Ohio, Zone A (us-east-2a)

Stop Reboot

Manage tags

Private IP: 172.26.8.147 Public IP: 18.189.193.255 Status: Running

Connect Storage Metrics Networking Snapshots Tags History Delete

Connect securely using your browser

You can still use your own compatible ssh client with your device or software to connect to your instance. Learn how to connect using your own SSH client

Connect using SSH

Connect using your own SSH client

You can connect to your instance using the following address and user name:

Public IP (P): 18.189.193.255
User name (U): ubuntu

When you connect with your client, you will also need the private key.
You configured this instance to use default (us-east-2) key pair.
You can download your default private key from the Account page.

©2020-2020, Amazon Web Services, Inc. or its affiliates

```
https://lightsail.aws.amazon.com/ls/remote/us-east-2/instances/dbClaseRodrigoYH/terminal?protocol=ssh
Welcome to Ubuntu 20.04 LTS (GNU/Linux 5.4.0-1018-aws x86_64)

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

System information as of Sat Nov 28 23:27:33 UTC 2020

System load:  0.34          Processes:    101
Usage of /:   6.7% of 19.32GB Users logged in: 0
Memory usage: 42%          IPv4 address for eth0: 172.26.8.147
Swap usage:   0%

0 updates can be installed immediately.
0 of these updates are security updates.

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

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-172-26-8-147:~$
```

Se abrirá una consola de comandos como se muestra en la imagen, primero procederemos a instalar Docker en nuestra instancia, para ello, necesitamos primero acceder en modo superusuario: “sudo -s”, posteriormente procedemos ya con la instalación de Docker, para ello ejecutaremos los siguientes comandos:

- apt-get update
- apt-get install Docker

```
https://lightsail.aws.amazon.com/ls/remote/us-east-2/instances/dbClaseRodrigoYH/terminal?protocol=ssh
Selecting previously unselected package dnsmasq-base.
Preparing to unpack .../7-dnsmasq-base_2.80-1.1ubuntu1_amd64.deb ...
Unpacking dnsmasq-base (2.80-1.1ubuntu1) ...
Selecting previously unselected package docker.io.
Preparing to unpack .../8-docker.io_19.03.8-0ubuntu1.20.04.1_amd64.deb ...
Unpacking docker.io (19.03.8-0ubuntu1.20.04.1) ...
Selecting previously unselected package ubuntu-fan.
Preparing to unpack .../9-ubuntu-fan_0.12.13_all.deb ...
Unpacking ubuntu-fan (0.12.13) ...
Setting up runc (1.0.0-rc10-0ubuntu1) ...
Setting up dns-root-data (2019052802) ...
Setting up libidn11:amd64 (1.33-2.2ubuntu2) ...
Setting up bridge-utils (1.6-2ubuntu1) ...
Setting up pigz (2.4-1) ...
Setting up cgroupfs-mount (1.4) ...
Setting up containerd (1.3.3-0ubuntu2) ...
Created symlink /etc/systemd/system/multi-user.target.wants/containerd.service - /lib/systemd/system/containerd.service
Setting up docker.io (19.03.8-0ubuntu1.20.04.1) ...
Adding group 'docker' (GID 119) ...
Done.
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket - /lib/systemd/system/docker.socket.
docker.service is a disabled or a static unit, not starting it.
Setting up dnsmasq-base (2.80-1.1ubuntu1) ...
Setting up ubuntu-fan (0.12.13) ...
Created symlink /etc/systemd/system/multi-user.target.wants/ubuntu-fan.service - /lib/systemd/system/ubuntu-fan.service
Processing triggers for systemd (245.4-4ubuntu3.1) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for dbus (1.12.16-2ubuntu2.1) ...
Processing triggers for libc-bin (2.31-0ubuntu9) ...
root@ip-172-26-8-147:/home/ubuntu# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS
root@ip-172-26-8-147:/home/ubuntu#
```

Esperamos a que el proceso de instalación haya finalizado. De igual manera ejecutamos los siguientes comandos en el siguiente orden después de la instalación anterior:

- snap install Docker
- snap install Docker.io

Esperamos a que se carguen todos los archivos. Una vez finalizado todo el proceso de instalación, ejecutamos el siguiente comando: “docker ps” esto es para verificar que se haya instalado Docker en nuestra instancia.

```
https://lightsail.aws.amazon.com/ls/remote/us-east-2/instances/dbClaseRodrigoYH/terminal?protocol=ssh
Preparing to unpack .../9-ubuntu-fan_0.12.13_all.deb ...
Unpacking ubuntu-fan (0.12.13) ...
Setting up runc (1.0.0-rc10-0ubuntu1) ...
Setting up dns-root-data (2019052802) ...
Setting up libidn11:amd64 (1.33-2.2ubuntu2) ...
Setting up bridge-utils (1.6-2ubuntu1) ...
Setting up pigz (2.4-1) ...
Setting up cgroupfs-mount (1.4) ...
Setting up containerd (1.3.3-0ubuntu2) ...
Created symlink /etc/systemd/system/multi-user.target.wants/containerd.service → /lib/systemd/system/containerd.service.
Setting up docker.io (19.03.8-0ubuntu1.20.04.1) ...
Adding group 'docker' (GID 119) ...
Done.
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /lib/systemd/system/docker.socket.
docker.service is a disabled or a static unit, not starting it.
Setting up dnsmasq-base (2.80-1.1ubuntu1) ...
Setting up ubuntu-fan (0.12.13) ...
Created symlink /etc/systemd/system/multi-user.target.wants/ubuntu-fan.service → /lib/systemd/system/ubuntu-fan.service.
Processing triggers for systemd (245.4-4ubuntu3.1) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for dbus (1.12.16-2ubuntu2.1) ...
Processing triggers for libc-bin (2.31-0ubuntu9) ...
root@ip-172-26-8-147:/home/ubuntu# docker ps
CONTAINER ID        IMAGE               NAMES               COMMAND              CREATED             STATUS
root@ip-172-26-8-147:/home/ubuntu# git clone https://github.com/RodrigoYH/AdminBD.git
Cloning into 'AdminBD'...
remote: Enumerating objects: 17, done.
remote: Counting objects: 100% (17/17), done.
remote: Compressing objects: 100% (16/16), done.
remote: Total 17 (delta 2), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (17/17), 1001.52 KiB | 3.13 MiB/s, done.
root@ip-172-26-8-147:/home/ubuntu#
```

El siguiente paso consiste en instalar nuestro contenedor de mariaDB en nuestra instancia, para ello vamos a clonar nuestro archivo “docker-compose.yml” de GitHub a nuestra instancia con el siguiente comando:

```
git clone <la ruta de nuestro repositorio de GitHub>
```

Esperamos a que se termine de clonar los datos de nuestro repositorio.

Escribimos “ls” para verificar que si tenemos la carpeta del repositorio y posteriormente “cd <El nombre del repositorio>” en mi caso es “cd AdminBD”.

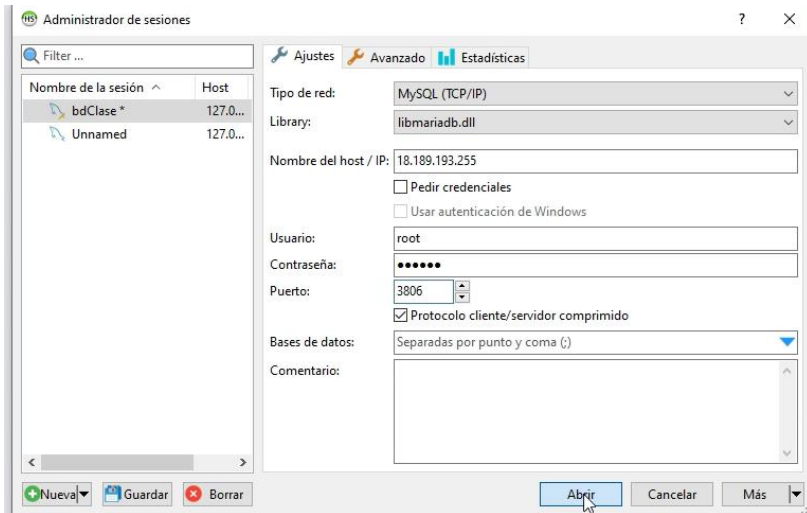
Una vez dentro de la carpeta, procedemos a levantar el contenedor, para ello ejecutamos el siguiente comando: “docker-compose up -d”.

```
https://lightsail.aws.amazon.com/ls/remote/us-east-2/instances/dbClaseRodrigoYH/terminal?protocol=ssh
Processing triggers for dbus (1.12.16-2ubuntu2.1) ...
Processing triggers for libc-bin (2.31-0ubuntu9) ...
root@ip-172-26-8-147:/home/ubuntu# docker ps
CONTAINER ID        IMAGE               NAMES               COMMAND              CREATED             STATUS
root@ip-172-26-8-147:/home/ubuntu# git clone https://github.com/RodrigoYH/AdminBD.git
Cloning into 'AdminBD'...
remote: Enumerating objects: 17, done.
remote: Counting objects: 100% (17/17), done.
remote: Compressing objects: 100% (16/16), done.
remote: Total 17 (delta 2), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (17/17), 1001.52 KiB | 3.13 MiB/s, done.
root@ip-172-26-8-147:/home/ubuntu# ls
AdminBD
root@ip-172-26-8-147:/home/ubuntu# cd AdminBD
bash: cd: AdminBD: No such file or directory
root@ip-172-26-8-147:/home/ubuntu# cd AdminBD
root@ip-172-26-8-147:/home/ubuntu/AdminBD# ls
Parcial1.pdf  clase.rar  docker-compose.yml
root@ip-172-26-8-147:/home/ubuntu/AdminBD# docker-compose up -d
Creating network "adminbd_default" with the default driver
Pulling basedatos (mariadb)...
latest: Pulling from library/mariadb
da7391352a9b: Pull complete
14428a6d4bcd: Pull complete
2c2d948710f2: Pull complete
22776aa82430: Pull complete
90e64230d63d: Pull complete
f30861f14a10: Pull complete
e8e9e6a3da24: Pull complete
420a23f08c41: Extracting [=====] 786.4kB/1.
267MBB3de482: Download complete
a8690a3260b7: Download complete
4202ba90333a: Download complete
8.93MB0b4aa6: Download complete
919kB
```

Esperamos a que se terminen de ejecutar todos los procesos. de instalación del archivo.ym.

Podemos escribir “docker ps” para verificar que nuestro contenedor se halla instalado correctamente en nuestra instancia.

El siguiente paso es crear las reglas de firewall, para ello vamos a abrir nuestro administrador de base de datos, en mi caso yo estoy utilizando HeidiSQL:



Escribimos los datos de acceso a nuestra instancia, colocamos la dirección IP pública que nos proporciona nuestra instancia, seguido de nuestro usuario root, nuestra contraseña y el puerto que definimos en nuestro archivo "docker-compose.yml".

Si intentamos acceder, nos va a marcar un error y esto es porque todavía no hemos definido las reglas de firewall. Esto es lo que vamos a hacer a continuación.

Nos dirigimos a nuestra instancia, y en la sección de "Networking", primero en la sección de firewall vamos a eliminar la regla HTTP del puerto 80, posteriormente procedemos a crear una nueva regla de firewall:

Firewall

Create rules to open ports to the internet, or to a specific IP address or range.

[Learn more about firewall rules](#)

+ Add rule

Specify a port and protocol to open. Specify a port range using a dash, such as 0 - 65535.

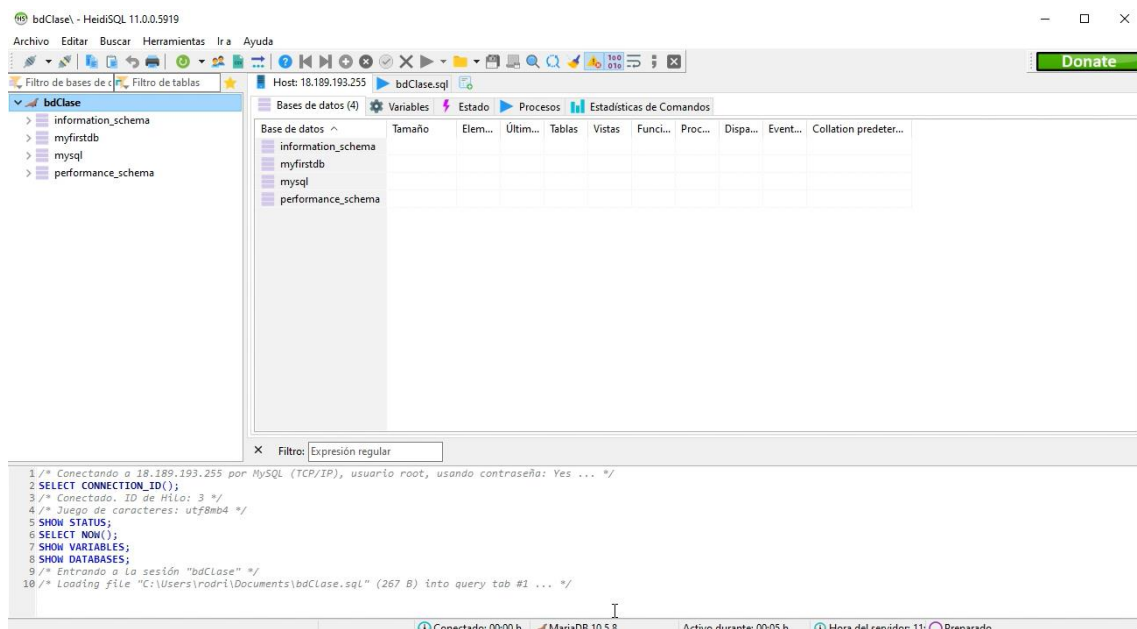
Application	Protocol	Port or range	<input type="checkbox"/> Restrict to IP address
Custom	TCP	3806	

Cancel

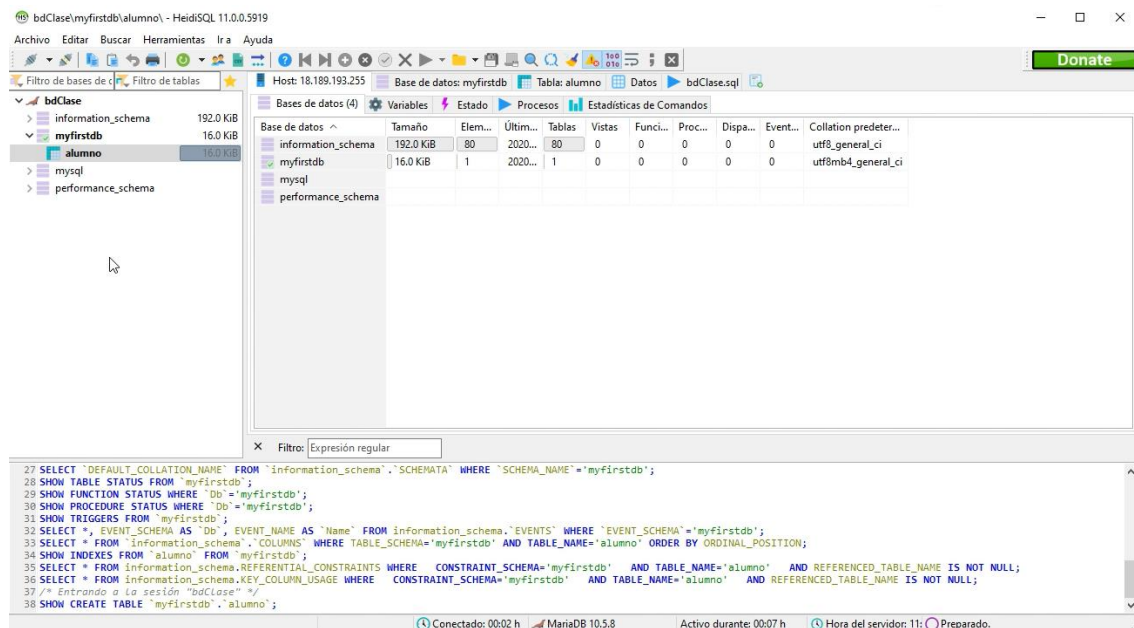
Application	Protocol	Port or range / Code	Restricted to
SSH	TCP	22	Any IP address Lightsail browser SSH/RDP

En la nueva regla definimos como protocolo el TCP y colocamos el puerto que definimos en nuestro archivo "docker-compose.yml" y le damos crear.

Volvemos a hacer una segunda prueba de acceso desde HeidiSQL.



Ahora si tenemos el acceso ya que hemos definido la regla de firewall, procedemos a crear una tabla simple en la base de datos, yo en mi caso, creé una tabla “Alumno” con los atributos: idAlumno, nombre, carrera y semestre.



Finalmente queda hacer el último paso que es bloquear el paso al exterior, para ello vamos a modificar nuestra regla de firewall que hicimos anteriormente:

Firewall ?

Create rules to open ports to the internet, or to a specific IP address or range.

[Learn more about firewall rules](#)

+ Add rule

Application	Protocol	Port or range / Code	Restricted to
SSH	TCP	22	Any IP address Lightsail browser SSH/RDP

Specify a port and protocol to open. Specify a port range using a dash, such as 0 - 65535.

Application:
 Protocol:
 Port or range:
☒ Restrict to IP address

Specify an IP address allowed to connect to your instance. You can specify a range of IP addresses using a dash, or using CIDR notation. [Learn more](#)

Source IP address (192.0.2.0) or range (192.0.2.0-192.0.2.255 or 192.0.2.0/24)

+ Add another

Cancel ☒ Save ☒

Vamos a seleccionar la opción “Restrict to IP address” y en el recuadro colocamos la dirección IP de nuestra computadora. Esto es para que nadie pueda entrar a nuestra instancia y solo con esta dirección IP se pueda acceder. En mi caso si podré acceder porque estoy en mi computadora, pero si intento acceder desde otra computadora no me va a dejar acceder ya que no es una dirección IP que se encuentra registrada para darle acceso a la instancia.