AWS.md

Introducción a Amazon AWS

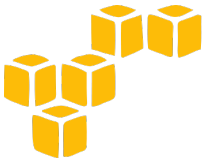


Tabla de contenidos

Introducción al entorno de prácticas

Entramos en la consola de desarrolladores AWS pertenenciente a CDA:

```
https://cdaesei.signin.aws.amazon.com/console
```

Una vez dentro, nos logueamos con los siguientes datos:

> **Cuenta** cdaesei

> **Usuario** `censurado`

> **Contraseña** `censurado`

Trabajaremos siempre sobre el servidor **EU (Ireland)**. Por otra parte, todas las instancias que levantemos serán **"Free tier eligible"** y **"Shutdown behaviour: Terminate"**.

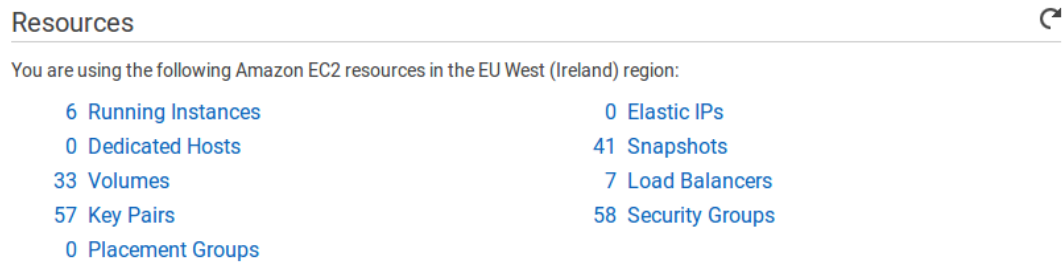
Por último, incluirán obligatoriamente el prefijo **CDA2018_**.

Seguir cualquier indicación mostrada en esta página en caso de duda o error.

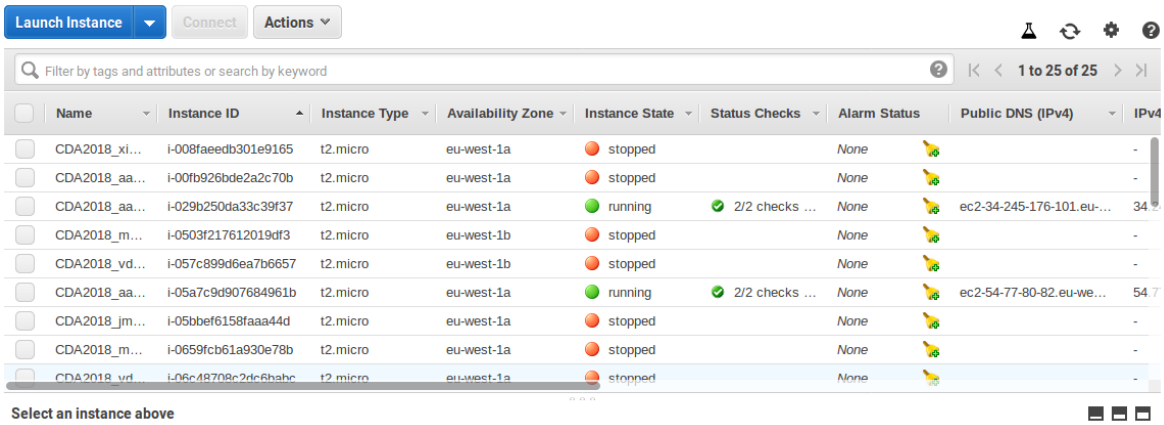
Creación de una instancia UBUNTU 14.04 LTS(HVM)

Lo primero entrar en los servicios **EC2** de AWS.

Una vez dentro, nos aparecerá el panel de control con todos los recursos disponibles:



Seleccionaremos **Running Instances**. De esta forma podemos ver todas las instancias anteriormente creadas y podremos configurar nuevas:

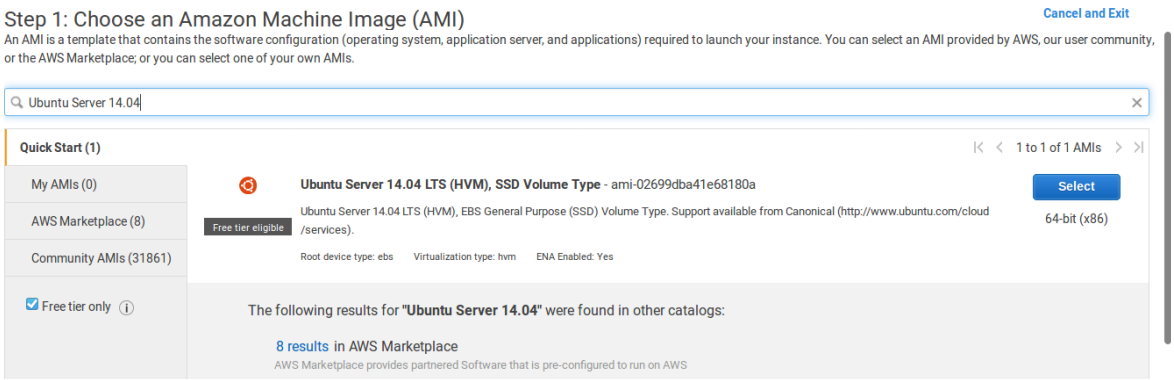


Pulsaremos en **Launch Instance** y procederemos a crear una nueva instancia.

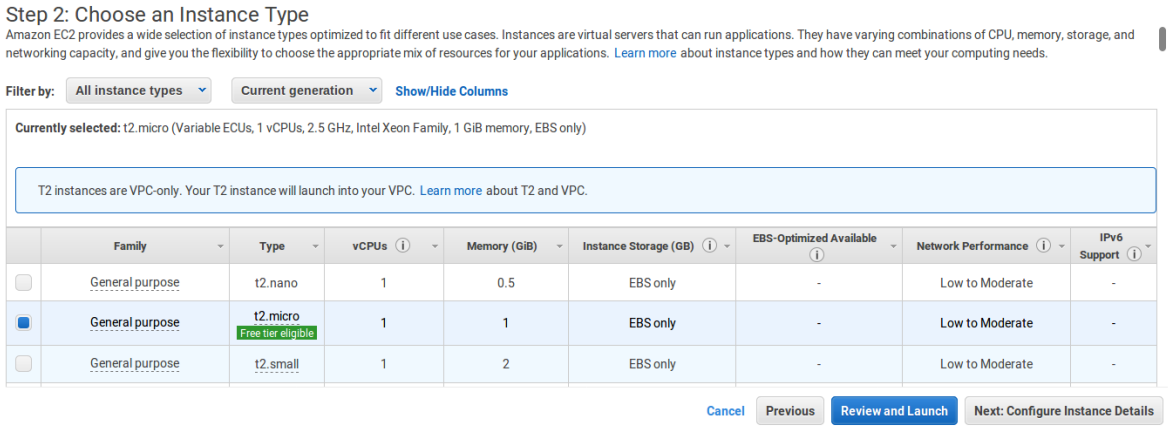
En nuestro caso deseamos que sea un `Ubuntu Server 14.04 LTS` , así que buscaremos la AMI correspondiente.

> **Cuestión1** Una *Amazon Machine Image* o *AMI* es un máquina virtual preconfigurada que será usada por *Amazon EC2* para crear instancias, es decir, máquinas virtuales clon de las *AMI* usadas por el usuario.
>
> De esta forma, una misma *AMI* sirve para crear múltiples instancias.

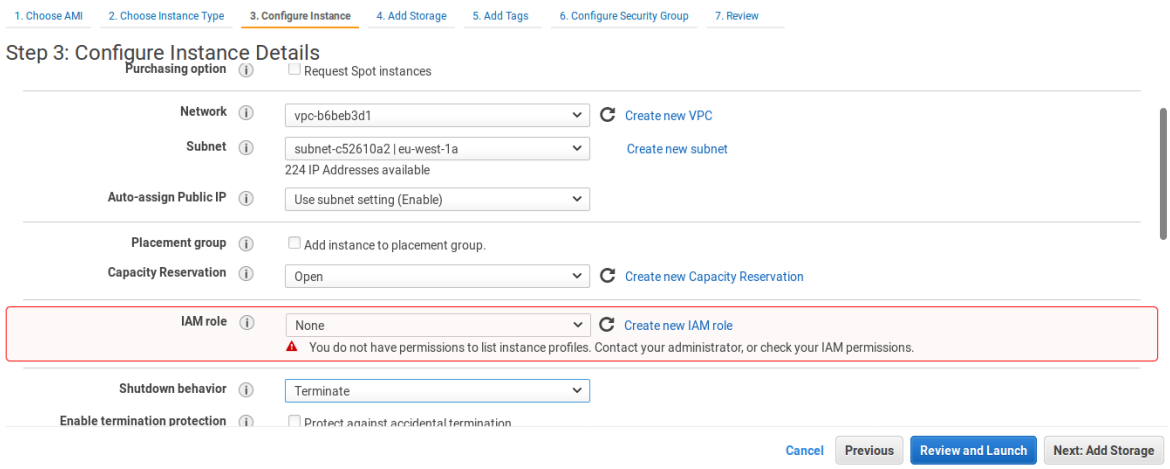
Como se puede observar, es necesario marcar la casilla **Free tier only** para evitar costes imprevistos.



De esta misma forma, escogeremos el tipo de instancia **t2.micro**, dado que es aquella pertenenciente al **Free tier**, y pulsamos **Next: Configure Instance Details**.

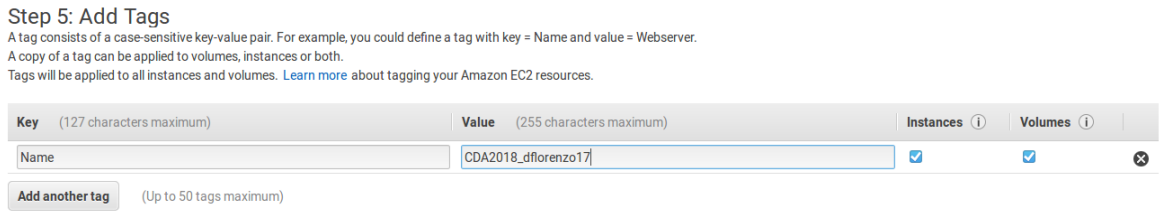


Entre todas las opciones disponibles, nos aseguramos de que **Shutdown behaviour** esté en **Terminate**.

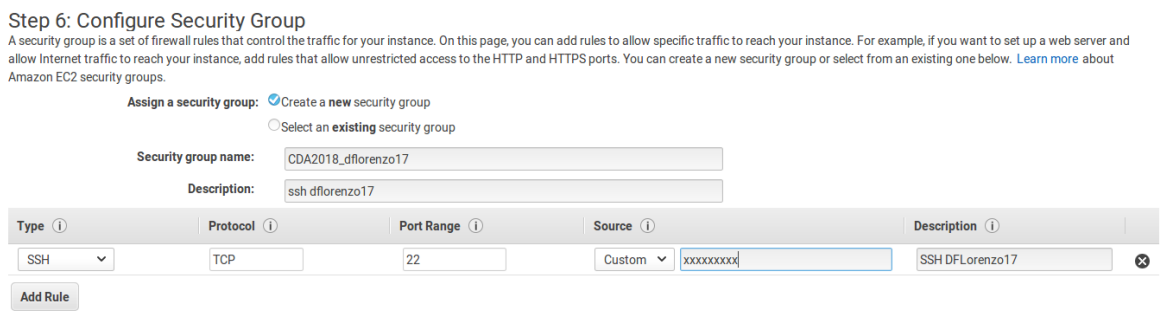


Pulsaremos en **Next: Add Storage** y seguido en **Next: Add tags**, dado que nos sirve la opción por defecto de almacenamiento.

En los tags añadiremos la etiqueta **Name** con el valor **CDA2018_usuario**:



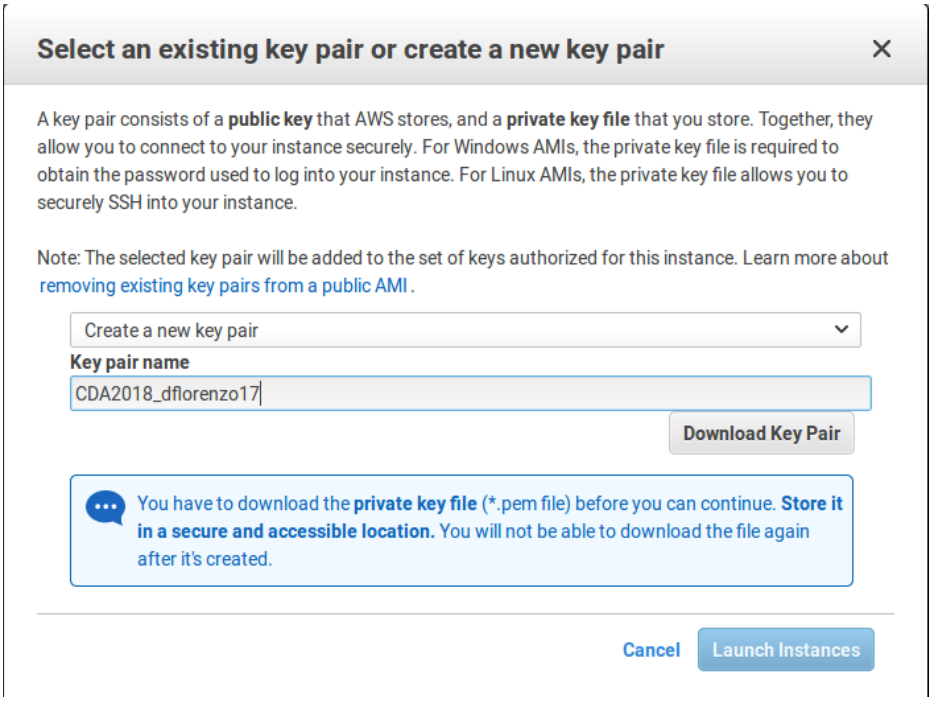
Por último configuraremos los **Security Groups** de tal forma que sólo nos podamos conectar desde nuestro equipo. Seleccionaremos un tipo de conexión SSH(22/TCP) desde nuestra IP:



> **Cuestión2** Un *Security Group* es un conjunto de reglas que forman un firewall virtual para poder controlar el tráfico de una o varias instancias.

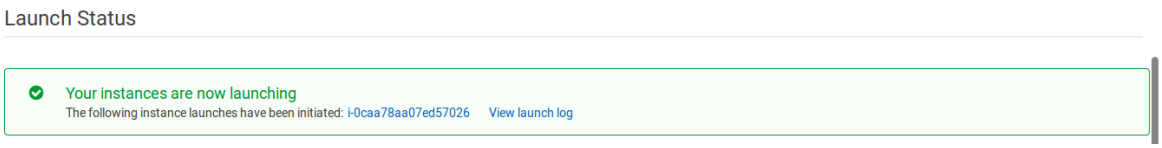
Aceptamos todo y lanzamos nuestra instancia.

Si no tenemos un par de claves creados, los generaremos de la siguiente forma:



> **Cuestión3** El par de llaves de AWS es un método criptográfico de clave pública que nos permite cifrar ciertos datos para realizar las pertinentes conexiones, como la contraseña. Por otro lado, el destinatario utiliza la clave privada para descrifrar dichos datos.

¡Y ya estaría! Nuestra instancia de un Ubuntu Server 14.04 LTS estaría siendo ejecutada ahora mismo:



Uso de la instancia creada

Para conectarnos a nuestra nueva instancia, usaremos el siguiente comando:

```
ssh -i <archivo .pem> <usuario>@<EC2 DNS (IPv4)>
# En mi caso:
ssh -i Downloads/CDA2018_dflorenzo17.pem ubuntu@ec2-34-240-246-60.eu-west-1.compute.amazonaws.com
```

Al entrar deberíamos ver la siguiente salida:

```
Welcome to Ubuntu 14.04.5 LTS (GNU/Linux 3.13.0-161-generic x86_64)

 * Documentation:  https://help.ubuntu.com/
```

```
  System information as of Sun Dec 30 15:49:12 UTC 2018

  System load:  0.0               Processes:           98
  Usage of /:   10.3% of 7.74GB   Users logged in:     0
  Memory usage: 5%                IP address for eth0: 172.30.0.171
  Swap usage:   0%

  Graph this data and manage this system at:
    https://landscape.canonical.com/

  Get cloud support with Ubuntu Advantage Cloud Guest:
    http://www.ubuntu.com/business/services/cloud

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

New release '16.04.5 LTS' available.
Run 'do-release-upgrade' to upgrade to it.



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.

ubuntu@ip-172-30-0-171:~$
```

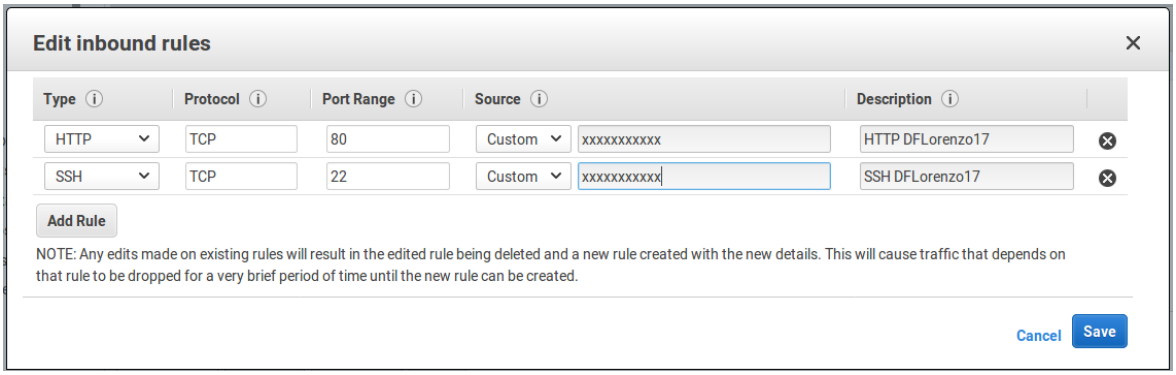
Instalación de un servidor web con PHP

Ejecutaremos los siguientes comandos para instalar el servidor web:

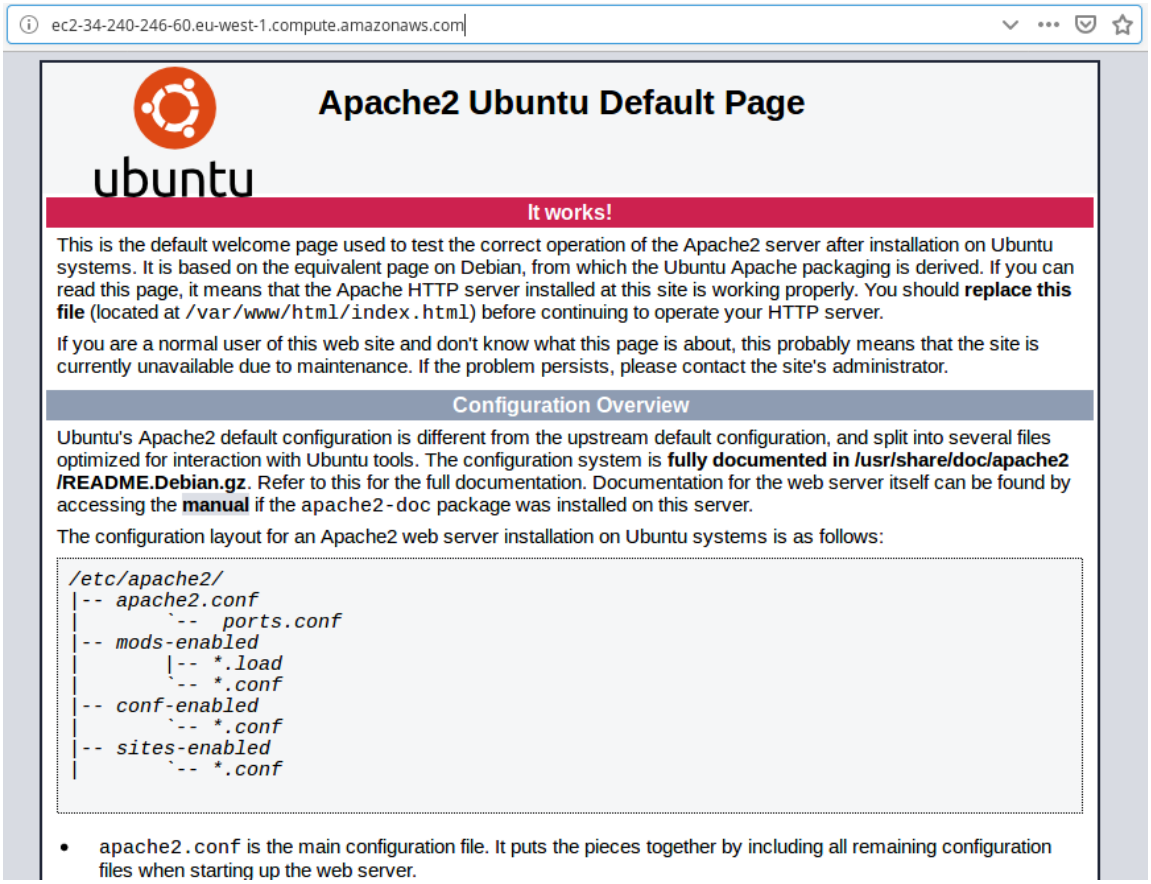
```
sudo apt-get update -y
sudo apt-get install -y apache2
sudo service apache2 restart
```

Si intentamos acceder a la dirección DNS(IPv4) desde el navegador, no obtendremos respuesta.

Para ello deberemos añadir la regla HTTP al Security Group:



De esta forma ya podemos acceder al servidor web a través de nuestro navegador:



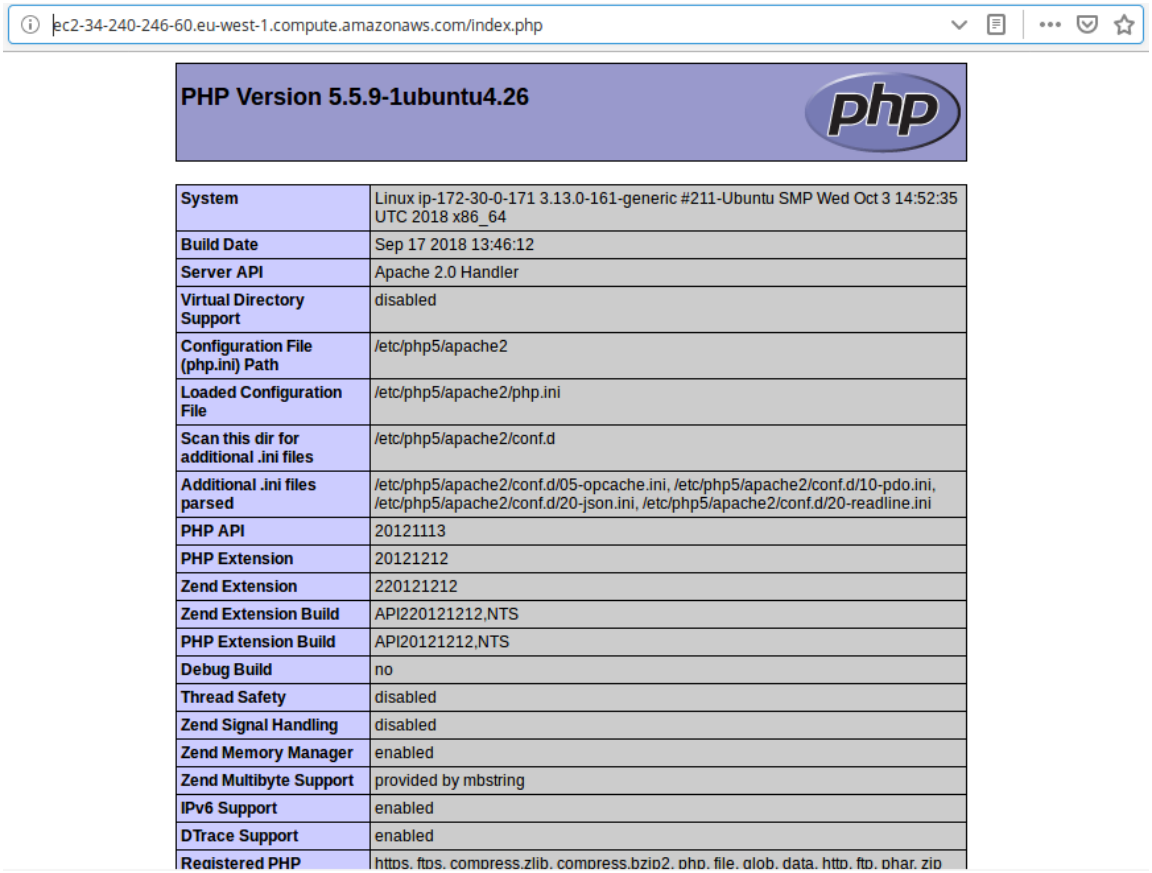
Ahora sólo falta instalar PHP ejecutando el comando:

```
sudo apt-get install -y php5
```

Crearemos un archivo **phpinfo** con la siguiente instrucción:

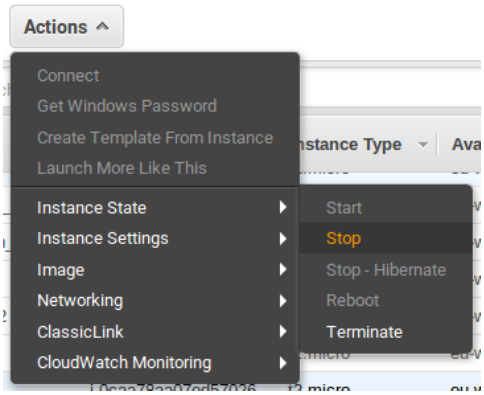
```
echo "<?php phpinfo(); ?>" | sudo tee /var/www/html/index.php
```

Si intentamos acceder desde el navegador a la página `index.php` , deberíamos ver algo similar a esto:



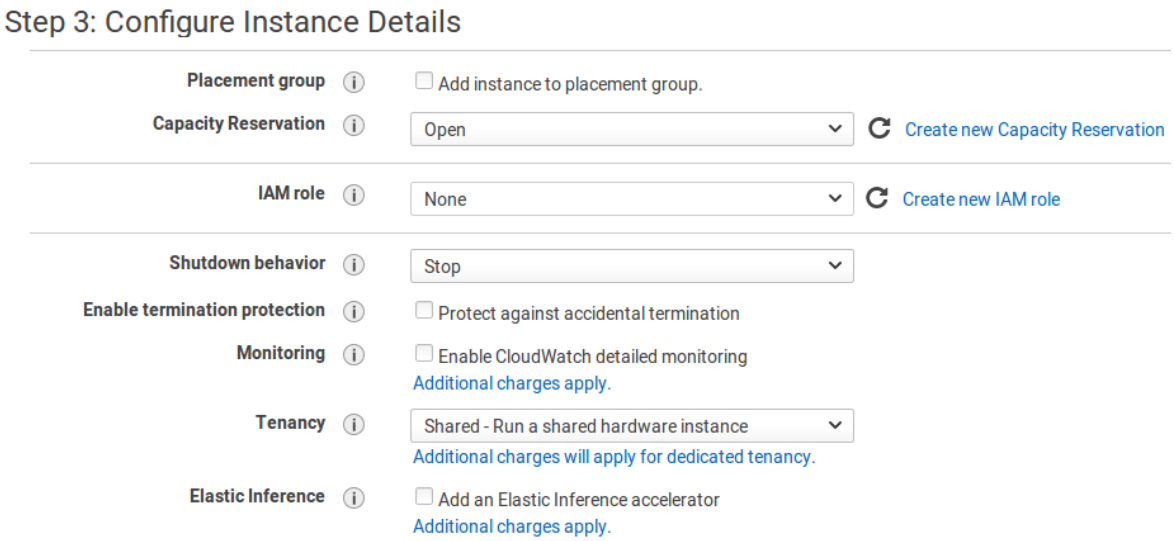
Detención de la instancia

Para detener la instancia, la seleccionaremos en el Panel de Control de Amazon EC2 y en **Actions** pulsaremos la opción **Stop**:

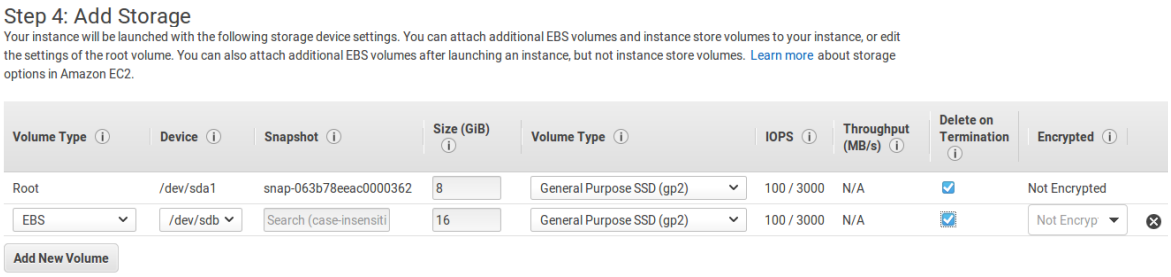


Creación de varias instancias

Esta vez crearemos la instancia con la opción **Shutdown behaviour** en **Stop**:



Además le agregaremos un disco **SSD de 16Gbi**:



Una vez creada, accedemos a ella a través del puerto 22 (SSH) y comprobamos la presencia de los dos discos duros con `cat /proc/partitions`:

```
ubuntu@ip-172-30-0-24:~$ cat /proc/partitions
major minor  #blocks  name

 202        0    8388608 xvda
 202        1    8377897 xvda1
 202       16   16777216 xvdb
```

Como podemos observar, tenemos un dispositivo `xvda` formateado como `xvda1`, el cual contiene el sistema operativo, y otro dispositivo `xvdb` de 16Gbi que es el añadido.

Ahora formatearemos el disco `xvdb` con el comando:

```
sudo mkfs.ext4 /dev/xvdb
```

Esta será la salida:

```
mke2fs 1.42.9 (4-Feb-2014)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=0 blocks, Stripe width=0 blocks
1048576 inodes, 4194304 blocks
209715 blocks (5.00%) reserved for the super user
```

```
First data block=0
Maximum filesystem blocks=4294967296
128 block groups
32768 blocks per group, 32768 fragments per group
8192 inodes per group
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
        4096000

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done
```

> Para comprobar el correcto funcionamiento del nuevo disco, vamos a montarlo

```
# Montamos el disco
sudo mkdir /cda2018
sudo mount /dev/xvdb /cda2018
# Comprobamos el punto de montaje
df -Th /cda2018
```

Veremos la siguiente salida:

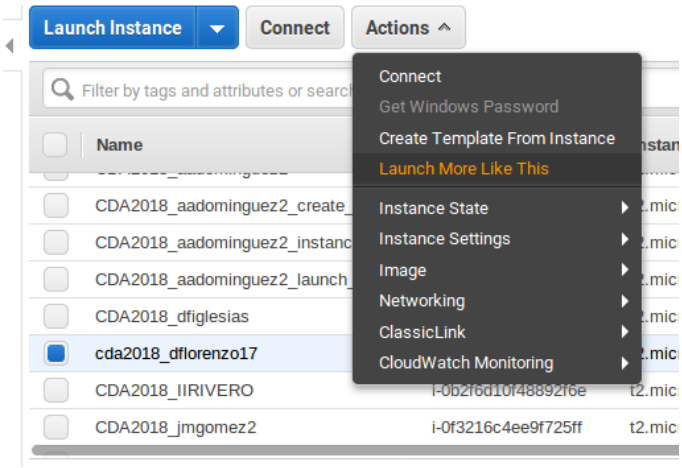
```
Filesystem     Type  Size  Used Avail Use% Mounted on
/dev/xvdb      ext4   16G   44M   15G   1% /cda2018
```

Por último añadimos la línea `/dev/xvdb /cda2018 ext4 noatime 0 0` al archivo `/etc/fstab`

y crearemos un archivo de texto en `/cda2018` con la frase *CDA2018_tarea3*.

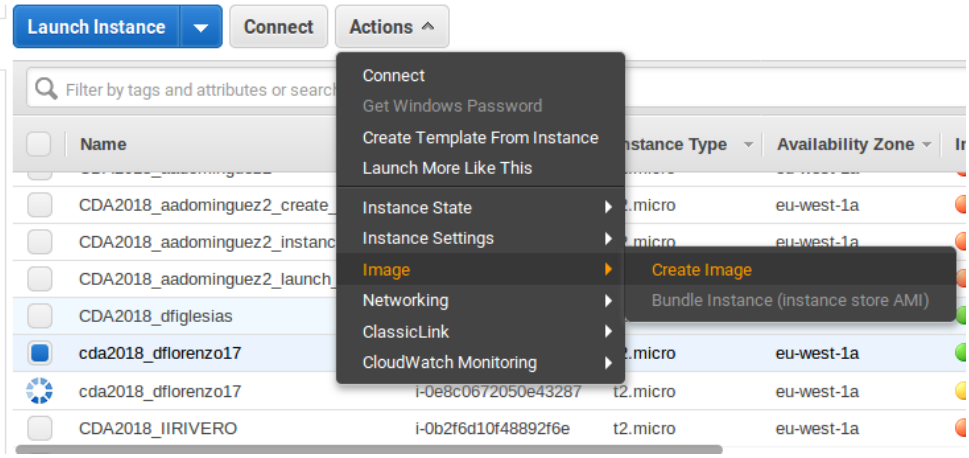
Duplicaremos la instancia de las siguientes maneras:

- Actions > Launch More Like This



> **Cuestión4** El método *Launch More Like This* nos permite duplicar una instancia, la cual será usada como plantilla, para poder desplegar muchas similares haciendo que AWS copie los datos de configuración de esta, pero no clonándola directamente.

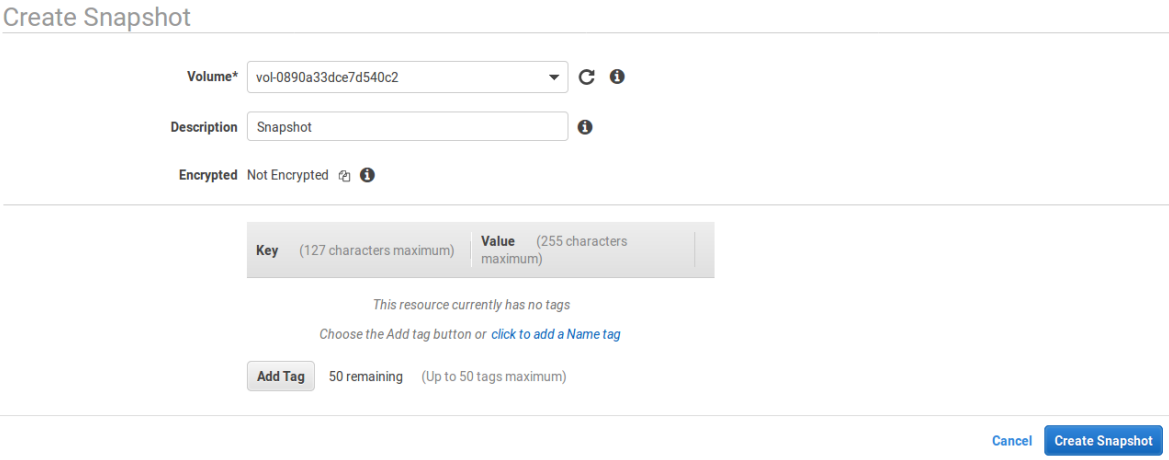
- Actions > Image > Create Image



> **Cuestión4** El método *Create Image* sí que clona la instancia, haciendo a partir de ella una imagen que será igual a la original.

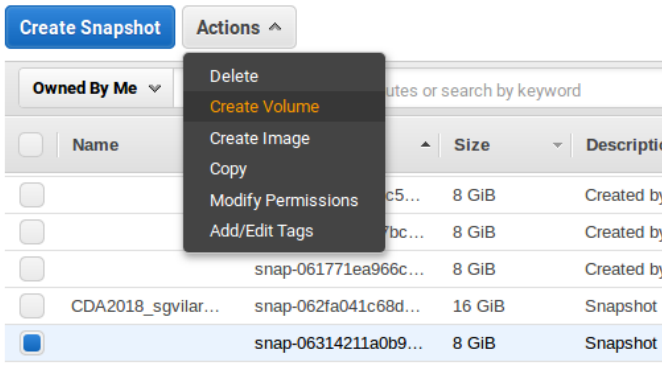
Vinculando volúmenes

Crearemos un *snapshot* del volumen de nuestra imagen usando el **Dashboard EC2**:

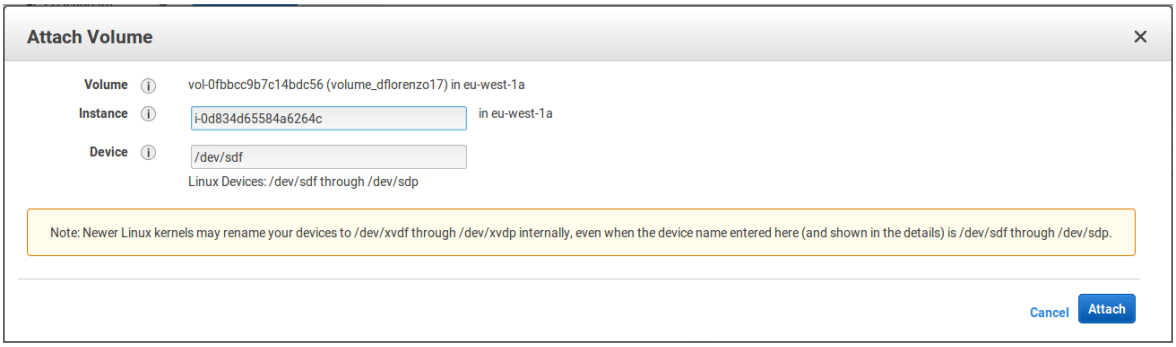


> **Cuestión5** Los *snapshots* son una forma de poder preservar el estado actual de los volúmenes de una instancia (o dispositivos). De esta forma, podremos compartir o restaurar el estado de una instancia.

Una vez creado el *snapshot* lo usaremos para crear otro volumen:



Y lo vinculamos a una de las instancias ya creadas (en este caso **launch_more**):



Ahora modificamos el texto de nuestro fichero alojado en `/cda2018` y vinculamos el volumen a la instancia `create_image`.

Bibliografía

- https://github.com/Student-Puma/HomeLab
- https://www.dev-metal.com/install-setup-php-5-6-ubuntu-14-04-lts/
- https://docs.aws.amazon.com/es_es/AWSEC2/latest/UserGuide/ec2-key-pairs.html
- https://www.digitalocean.com/community/tutorials/como-instalar-linux-apache-mysql-php-lamp-en-ubuntu-16-04-es
- https://cursos.faitic.uvigo.es/tema1819/claroline/document/goto/index.php/2018-2019/AWS-EC2_2018.pdf