

S3 y Load Balancers



Tabla de contenidos

- Introducción al entorno de prácticas
- Creación y Configuración del Bucket (Consola S3)
- Cargar objetos en el Bucket (Consola S3)
- Creación de las Instancias y vinculación con el Bucket (Consola EC2)
- Creación y Configuración del Balanceador (Consola EC2)
- Modificar el Security Group de las instancias (Consola EC2)
- Bibliografía

Introducción al entorno de prácticas

Entramos en la consola de desarrolladores AWS perteneciente 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 y Configuración del Bucket (Consola S3)

Entraremos en los servicios **S3** de AWS y le daremos a **Crear Bucket**.

Le pondremos un nombre y dejaremos como región **EU (Ireland)**.

Crear bucket

1

Nombre y región

2

Configurar opciones

3

Establecer permisos

4

Revisión

Nombre y región

Nombre del bucket ⓘ

dflorenzo17

Región

UE (Irlanda) ▾

Copiar configuración de un bucket existente

Seleccionar bucket (opcional) 26 buckets ▾

Crear

Cancelar

Siguiente

Cuestión 1: Un bucket se crea en una determinada región, como el resto de servicios AWS, pero se pueden gestionar accesos públicos, los cuales son globales.

Las opciones del bucket las dejaremos por defecto salvo los permisos, los cuales *setearemos* como públicos dejándolos de la siguiente manera:

Crear bucket

✓ 1

Nombre y región

✓ 2

Configurar opciones

3

Establecer permisos

4

Revisión

Administrar las listas de control de acceso (ACL) públicas para este bucket ⓘ

☐ Bloquear nuevas ACL públicas y la carga de objetos públicos (Recomendado) ⓘ

☐ Quitar el acceso público concedido mediante ACL públicas (Recomendado) ⓘ

Administrar políticas de bucket públicas para este bucket ⓘ

☐ Bloquear nuevas políticas de bucket públicas (Recomendado) ⓘ

☐ Bloquear el acceso público y entre cuentas si el bucket tiene políticas públicas (Recomendado) ⓘ

Administrar permisos del sistema

No conceder al grupo Envío de registros de Amazon S3 acceso de ... ▾

Anterior

Siguiente

Cargar objetos en el Bucket (Consola S3)

Una vez creado el bucket , cargaremos diversas imágenes con el formato `imgX.png` .

Para ello seleccionamos **Cargar** y subiremos los archivos de la siguiente manera:



Como permisos pondremos **Public/Read**:

Cargar

1

Seleccionar archivos

2

Establecer permisos

3

Establecer propiedades

4

Revisión

Administrar usuarios

ID de usuario	Objetos	Permisos del objeto
franrm(Propietario)	<input checked="" type="checkbox"/> Lectura <input checked="" type="checkbox"/> Escritura	<input checked="" type="checkbox"/> Lectura <input checked="" type="checkbox"/> Escritura <input type="checkbox"/>

Acceso para otras cuentas de AWS

+ Añadir cuenta

Cuenta	Objetos	Permisos del objeto
--------	---------	---------------------

Administrar permisos públicos

Conceder acceso de lectura público a estos objetos

⚠

Estos objetos tienen acceso de lectura público

Cargar

Anterior

Siguiente

Creación de las Instancias y vinculación con el Bucket (Consola EC2)

Ahora crearemos una **instancia EC2** igual que en la práctica anterior (la de introducción a AWS):

Creamos una instancia Ubuntu Server 14.04

Edit inbound rules

Type

Protocol

Port Range

Source

Description

HTTP	TCP	80	Custom	xxxxxxxxxx	HTTP DFlorenzo17
SSH	TCP	22	Custom	xxxxxxxxxx	SSH DFlorenzo17

Add Rule

NOTE: Any edits made on existing rules will result in the edited rule being deleted and a new rule created with the new details. This will cause traffic that depends on that rule to be dropped for a very brief period of time until the new rule can be created.

Cancel

Save

Nos conectamos a ella mediante SSH e instalamos **Apache2 con PHP** usando el siguiente comando:

```
sudo apt-get install -y apache2 php5
```

A continuación, creamos el archivo `/var/www/html/index.php` con el siguiente contenido:

```
<html>

</html>
```

De esta forma, este servidor mostrará dos de las cuatro imágenes subidas al bucket .

Realizamos una copia de la instancia usando alguna de las metodologías de la práctica anterior y modificamos el fichero `/var/www/html/index.php` para que ahora muestre las otras 2 imágenes subidas:

```
<html>

</html>
```

Cuestión 2: En este caso, la segunda instancia ha sido creada usada el método de `Launch More Like This` y configurándola manualmente igual que la primera.

Además, modificaremos la **Instancia** para que se despliegue en una zona de disponibilidad diferente a la anterior:

☐ Request Spot instances

Network

vpc-b6beb3d1

Create new VPC

Subnet

subnet-816a51c8 | eu-west-1b

Create new subnet

242 IP Addresses available

Auto-assign Public IP

Enable

Cuestión 3: En este caso, tanto el par de claves como el Security Group son los mismos que en la primera instancia, dado que lo que queremos son copias del mismo servidor, por lo que si hay que modificarlos, es mucho más rápido y sencillo.

Por último, comprobamos que ambas instancias funcionan correctamente.

La cada instancia debería mostrar aleatoriamente dos imágenes, las cuales serán distintas dependiendo de la instancia.

Creación y Configuración del Balanceador (Consola EC2)

En la consola **EC2** de AWS seleccionamos **Load Balancers** y creamos uno nuevo que sea `internet-facing` en el puerto 80:

1. Configure Load Balancer 2. Configure Security Settings 3. Configure Security Groups 4. Configure Routing 5. Register Targets

Step 1: Configure Load Balancer

To configure your load balancer, provide a name, select a scheme, specify one or more listeners, and select a network. The default listener that receives HTTP traffic on port 80.

Name	<input type="text" value="CDA2018_dflorenzo17"/>
Scheme	<input checked="" type="radio"/> internet-facing <input type="radio"/> internal
IP address type	<input type="text" value="ipv4"/>

Listeners

A listener is a process that checks for connection requests, using the protocol and port that you configured.

Load Balancer Protocol	Load Balancer Port
<input type="text" value="HTTP"/>	<input type="text" value="80"/>
<input type="button" value="Add listener"/>	

Seleccionamos las zonas de disponibilidad de las instancias creadas anteriormente:

Availability Zones

Specify the Availability Zones to enable for your load balancer. The load balancer routes traffic to the targets in 1 subnets from at least two Availability Zones to increase the availability of your load balancer.

VPC	<input type="text" value="vpc-b6beb3d1 (172.30.0.0/16)"/>	
Availability Zone	Subnet ID	Subnet IPv4 CIDR
<input checked="" type="checkbox"/> eu-west-1a	subnet-c52610a2	172.30.0.0/24
<input checked="" type="checkbox"/> eu-west-1b	subnet-816a51c8	172.30.1.0/24
<input type="checkbox"/> eu-west-1c	subnet-8781f2dc	172.30.2.0/24

Creamos, además, un nuevo **Security Group** para el puerto 80:

Step 3: Configure Security Groups

A security group is a set of firewall rules that control the traffic to your load balancer. On this page, you can add rules to allow specific traffic to reach your load balancer. First, decide whether to create a new security group or select an existing one.

Assign a security group:	<input checked="" type="radio"/> Create a new security group <input type="radio"/> Select an existing security group
Security group name:	<input type="text" value="CDA2018_dflorenzo17-loadbalancer-security"/>
Description:	<input type="text" value="load-balancer-wizard-1 created on 2019-01-08T18:20:53.248+01:00"/>

Type	Protocol	Port Range	Source
<input type="text" value="Custom TCP"/>	<input type="text" value="TCP"/>	<input type="text" value="80"/>	<input type="text" value="Custom"/> <input type="text" value="xxx.xxx.xxx.xxx"/>

Cuestión 4: En este caso, como se trata de un servidor HTTP, el `Security Group` del **Load Balancer** permite el acceso a través del puerto 80, pero no del puerto 22(SSH), dado que si no se *randomizarían* las conexiones por este protocolo y en ocasiones será necesario conectarnos a una instancia en concreto.

Configuramos el enrutamiento hacia las instancias:

Step 4: Configure Routing

Target group

Target group ⓘ

New target group ▼

Name ⓘ

Target type
☒ Instance
☐ IP
☐ Lambda function

Protocol ⓘ

HTTP ▼

Port ⓘ

Y registramos como **targets** dichas instancias:

Step 5: Register Targets

Register targets with your target group. If you register a target in an enabled Availability Zone, the load balancer starts routing requests to the targets as soon as the registration process completes and 1 target passes the initial health checks.

Registered targets

To deregister instances, select one or more registered instances and then click Remove.

Remove

<input type="checkbox"/>	Instance ▼	Name ▼	Port ▼	State ▼	Security groups ▼	Zone
<input type="checkbox"/>	i-071133fb...	dflorenzo...	80	● running	dflorenzo17_security	eu-west-1a
<input type="checkbox"/>	i-0c321105...	dflorenzo...	80	● running	dflorenzo17_security	eu-west-1b

Modificar el Security Group de las instancias (Consola EC2)

Sólamente nos queda modificar las **inbound conexions** de las instancias creadas.

Para ello modificamos el **Security Group** que comparten de la siguiente manera:

Edit inbound rules ✕

Type ⓘ	Protocol ⓘ	Port Range ⓘ	Source ⓘ	Description ⓘ	
All traffic ▼	All	0 - 65535	Custom ▼ sg-080b48caa2adee2f5	LoadBalancer	✕
SSH ▼	TCP	22	Custom ▼ 77.27.190.113/32	SSH	✕

Add Rule

NOTE: Any edits made on existing rules will result in the edited rule being deleted and a new rule created with the new details. This will cause traffic that depends on that rule to be dropped for a very brief period of time until the new rule can be created.

Cancel Save

Cuestión 5: Ahora las instancias tendrás una regla hija que adopte las características del **Security Group** padre (el del Load Balancer) y además, cada una tendrá un regla SSH por el motivo comentando en la cuestión 4 .

Obtenemos la **dirección DNS** del **Load Balancer** y ¡comprobamos que funciona!

Ahora deberían mostrarse aleatoriamente una de las cuatro imágenes que contiene el bucket, dado que las peticiones se distribuyen entre ambas instancias.

Bibliografía

- ✓ <https://github.com/Student-Puma/HomeLab>
- ✓ https://docs.aws.amazon.com/es_es/AmazonS3/latest/dev/UsingBucket.html
- ✓ https://docs.aws.amazon.com/es_es/AWSEC2/latest/UserGuide/using-regions-availability-zones.html
- ✓ https://cursos.faitic.uvigo.es/tema1819/claroline/document/goto/index.php/2018-2019/AWS-EC2_2018_-_S3_-_Load_Balancers.pdf