



PRÁCTICA 1: DESPLIEGUE BÁSICO

RAÚL MATEUS SÁNCHEZ

COMPUTACIÓN EN LA NUBE
Escuela de Ingeniería Informática



Índice

Introducción	2
Objetivos	2
Desarrollo de la práctica	3
1. Despliega una instancia en EC2 que se pueda acceder por SSH desde el exterior, que llamaremos SSH_gate	3
2. Despliegue una instancia en EC2 que tenga un servidor web en la que muestre su nombre y su afición favorita. Esta máquina solo podrá ser accedida por SSH desde la máquina que desplegamos anteriormente SSH_gate.....	8
3. Actividad extra.....	14
Diagrama de arquitectura desplegada	21
Presupuesto y estimación de gasto de los recursos desplegados	22
Conclusiones	23

Introducción

En el marco actual de las infraestructuras tecnológicas, la computación en la nube se está convirtiendo ya en un estándar sobre el cual muchos servicios se asientan debido a su escalabilidad, facilidad de gestión y mantenimiento, así como el ahorro de costes que supone y la seguridad, rendimiento y productividad que proporciona. Además, se le proporciona al usuario un servicio con plena disponibilidad y accesibilidad, así como se le abstrae de la complejidad de este.

La computación en la nube ha permitido una evolución en cuanto a infraestructura y arquitectura sin precedentes tanto para empresas como usuarios, en donde se ha abierto una oportunidad de negocio que empresas como Amazon con AWS y Google con Google Cloud han sabido aprovechar y les ha permitido marcar la pauta en cuanto al *cloud computing*, manteniéndose a la vanguardia.

Objetivos

El objetivo de esta práctica es tener una primera toma de contacto con los servicios de AWS y aplicar los conocimientos obtenidos en la clase teórica. Para ello se utilizará el servicio EC2 para preparar y desplegar instancias en la nube.

Desarrollo de la práctica

1. Despliega una instancia en EC2 que se pueda acceder por SSH desde el exterior, que llamaremos SSH_gate

Para poder desplegar una instancia en EC2, necesitamos acceder a AWS Management Console. Para ello, desde el *Learner Lab*, una vez iniciado, descargaremos el fichero con formato *txt* que nos dará el enlace para poder entrar.

Una vez en *AWS Management Console*, debemos buscar el servicio EC2, el cual aparecerá como en la **Figura 1**: EC2 Dashboard.

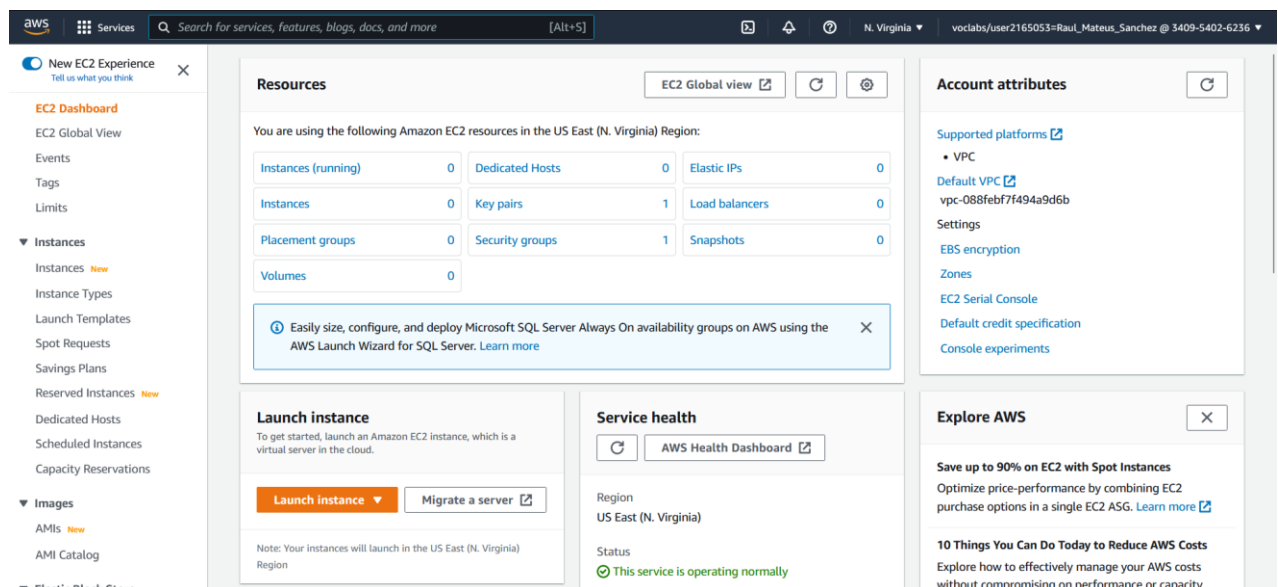


Figura 1: EC2 Dashboard

Launch an instance Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags Info

Name
[Add additional tags](#)

▼ Application and OS Images (Amazon Machine Image) Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Quick Start

Amazon Linux

macOS

Ubuntu

Windows

Red Hat

[Browse more AMIs](#)

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type

ami-026b57f3c383c2eec (64-bit (x86)) / ami-0636eac5d73e0e5d7 (64-bit (Arm))

Virtualization: hvm ENA enabled: true Root device type: ebs

Free tier eligible

Description

Amazon Linux 2 Kernel 5.10 AMI 2.0.20220912.1 x86_64 HVM gp2

Architecture
64-bit (x86)

AMI ID
ami-026b57f3c383c2eec

Verified provider

▼ Summary

Number of instances Info

Software Image (AMI)

Amazon Linux 2 Kernel 5.10 AMI...[read more](#)

ami-026b57f3c383c2eec

Virtual server type (instance type)

t2.micro

Firewall (security group)

New security group

Storage (volumes)

1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Cancel

Launch instance

Figura 2: Launching Instance

Para desplegar una instancia en EC2, seleccionaremos la opción *Launch Instance* donde podremos configurar la misma, como el nombre y la imagen que usaremos. En este caso, el enunciado de esta actividad nos indica que el nombre de la instancia deberá ser *SSH_gate*. La imagen que seleccionaremos será *Amazon Machine Image* o *Amazon Linux*, en su tipo de instancia más barata, ya que entendiendo que esta máquina solo nos servirá para realizar conexiones SSH tanto desde el exterior hacia esta como desde el interior hacia otras, no necesitaremos más recursos de los que nos da esta opción. De esta forma se refleja en la **Figura 2: Launching Instance**.

Create key pair

Key pairs allow you to connect to your instance securely.

Enter the name of the key pair below. When prompted, store the private key in a secure and accessible location on your computer. **You will need it later to connect to your instance.** [Learn more](#)

Key pair name

SSH_gate_KEY

The name can include upto 255 ASCII characters. It can't include leading or trailing spaces.

Key pair type

☒ RSA
RSA encrypted private and public key pair

☐ ED25519
ED25519 encrypted private and public key pair (Not supported for Windows instances)

Private key file format

☒ .pem
For use with OpenSSH

☐ .ppk
For use with PuTTY

Cancel Create key pair

Figura 3: Key Pairs creation

Para poder acceder a la instancia necesitaremos generar un par de claves pública/privada para este fin. Para ello, como se aprecia en la **Figura 3: Key Pairs creation**, el nombre de esta será SSH_gate_KEY, cuyo tipo será RSA.

A continuación, estableceremos la configuración de red. En este caso, al no necesitar nada específico, dejaremos la configuración por defecto simplemente estableciendo un nuevo nombre al grupo de seguridad. En futuras ocasiones se podrá ajustar las IP que puede acceder a la instancia, pero por el momento permitiremos a cualquier IP conectarse, tal y como vemos en la **Figura 4: Network Settings**.

▼ Network settings [Info](#)

VPC - required [Info](#)

vpc-088feb7f494a9d6b (default) [↻](#)

Subnet [Info](#)

No preference [↻](#) [Create new subnet](#)

Auto-assign public IP [Info](#)

Enable

Firewall (security groups) [Info](#)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☒ Create security group ☐ Select existing security group

Security group name - required

ssh_gate-instance-security-group

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and _-:/()#,@!+=8;()!\$*

Description - required [Info](#)

ssh_gate_instance_security_group created 2022-10-06T12:15:00.000Z

Inbound security groups rules

▼ Security group rule 1 (TCP, 22, 0.0.0.0/0) [Remove](#)

Type Info	Protocol Info	Port range Info
ssh	TCP	22

Source type Info	Source Info	Description - optional Info
Anywhere	<input type="text" value="Add CIDR, prefix list or security group"/> 0.0.0.0/0 ✕	e.g. SSH for admin desktop

Figura 4: Network Settings

Una vez configurada, solo restará desplegarla. Resultando exitoso su lanzamiento, visible en la figura 5, podremos mediante *PuTTY* conectarnos a nuestra instancia, siguiendo los pasos indicados en la parte de *Windows Users: Using SSH to Connect* en el README del *Learner Lab*.

✓ Success
Successfully initiated launch of instance (i-02b70a13521c15a9f)

▼ Launch log

Initializing requests	Succeeded
Creating security groups	Succeeded
Creating security group rules	Succeeded
Launch initiation	Succeeded

Figura 5: Instance Launched Successfully

Siguiendo las instrucciones, se configura *PuTTY* para que no se agote la sesión, permitiendo tener la sesión iniciada por un periodo de tiempo más largo, estableciendo en la sección *Connection* el campo **Seconds between keepalives** a 30. Simplemente restará introducir la IP pública de la instancia especificada en la interfaz de instancias de EC2 e introducir el fichero .ppk que se ha descargado el cual contiene la clave pertinente para poder realizar la conexión con la máquina.

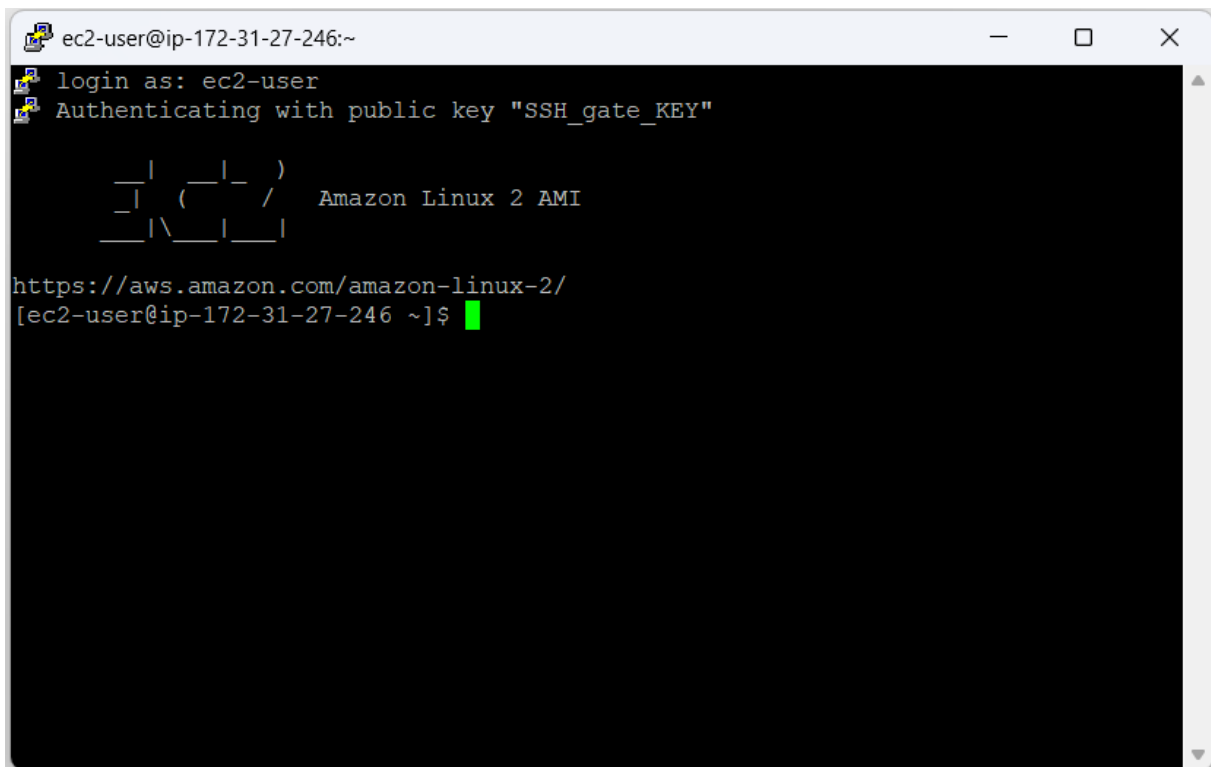


Figura 6: Conexión a la máquina "gate"

Una vez configurado *PuTTY*, se realiza la conexión con la instancia y se entrará con el usuario *ec2-user*, tal y como vemos en la **Figura 6: Conexión a la máquina "gate"**. A continuación, se procederá a desplegar otra instancia que actúe como servidor web accesible única y exclusivamente por la instancia ya creada.

2. Despliegue una instancia en EC2 que tenga un servidor web en la que muestre su nombre y su afición favorita. Esta máquina solo podrá ser accedida por SSH desde la máquina que desplegamos anteriormente SSH_gate.

Para desplegar esta nueva instancia se procederá del mismo modo que anteriormente. Desde el apartado de “*Launch an Instance*”, se configura esta nueva máquina de la misma manera que antes, aunque realizando ciertos ajustes en el grupo de seguridad para que solo pueda ser accedida por SSH_gate.

Name and tags [Info](#)

Name

web_server [Add additional tags](#)

▼ **Application and OS Images (Amazon Machine Image)** [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Search our full catalog including 1000s of application and OS images

Recents Quick Start

Amazon Linux macOS Ubuntu Windows Red Hat S

aws Mac ubuntu Microsoft Red Hat

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type Free tier eligible

ami-026b57f3c383c2eec (64-bit (x86)) / ami-0636eac5d73e0e5d7 (64-bit (Arm))

Virtualization: hvm ENA enabled: true Root device type: ebs

Description

Amazon Linux 2 Kernel 5.10 AMI 2.0.20220912.1 x86_64 HVM gp2

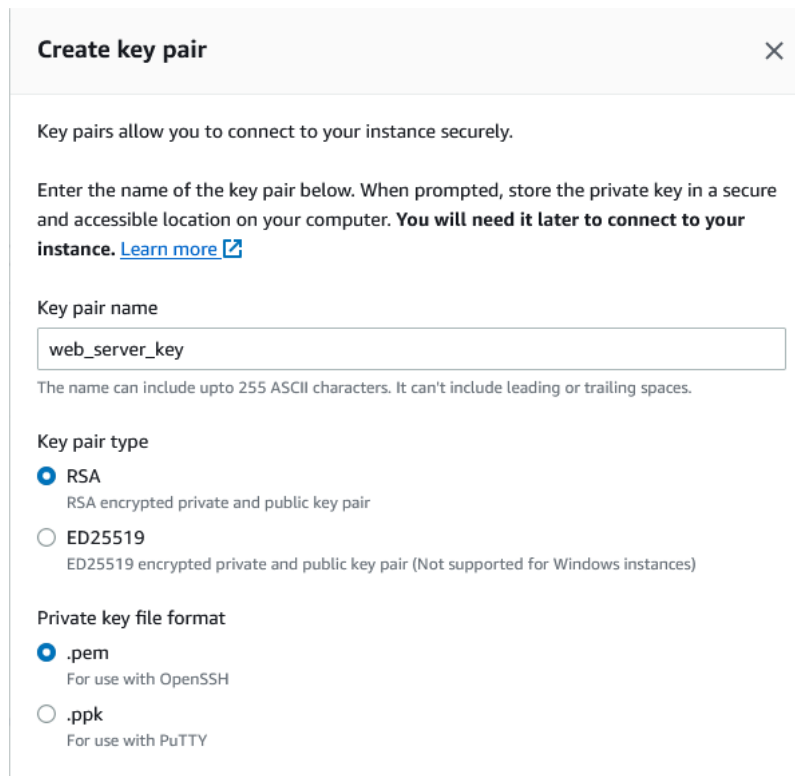
Architecture AMI ID

64-bit (x86) ami-026b57f3c383c2eec [Verified provider](#)

Figura 7: Creación del servidor web

Visible en la **Figura 7: Creación del servidor web** los primeros pasos de la creación de esta máquina, el nombre de esta máquina será “web_server”, y será construida bajo Amazon

Linux debido a la utilidad que se le va a dar. Al igual que en el apartado 1, se escoge el tipo de instancia más barata.



The screenshot shows a 'Create key pair' dialog box with a close button (X) in the top right corner. The dialog contains the following text and form elements:

- Header: **Create key pair**
- Introductory text: "Key pairs allow you to connect to your instance securely."
- Instructions: "Enter the name of the key pair below. When prompted, store the private key in a secure and accessible location on your computer. **You will need it later to connect to your instance.** [Learn more](#) [external link icon]"
- Form field: "Key pair name" with the text "web_server_key" entered.
- Validation text: "The name can include upto 255 ASCII characters. It can't include leading or trailing spaces."
- Section: "Key pair type"
- Options for Key pair type:
 - ☒ **RSA**
RSA encrypted private and public key pair
 - ☐ **ED25519**
ED25519 encrypted private and public key pair (Not supported for Windows instances)
- Section: "Private key file format"
- Options for Private key file format:
 - ☒ **.pem**
For use with OpenSSH
 - ☐ **.ppk**
For use with PuTTY

Figura 8: Creación de claves para la máquina "web_server"

Para poder acceder desde el "gate" a esta nueva máquina, se necesita generar las claves correspondientes. Como se aprecia en la **Figura 8: Creación de claves para la máquina "web_server"**, la creación de estas es prácticamente igual que para la máquina creada anteriormente, pero en este caso el formato de la clave privada será .pem debido a que accederemos a través de la máquina "SSH_gate", máquina Linux, a esta nueva máquina y no desde Windows.

Otra diferencia en la creación de esta nueva máquina será en los ajustes de red. Como esta máquina solo podrá ser accedida por la máquina "SSH_gate", se debe ajustar el grupo de seguridad para cumplir este requisito

▼ Network settings [Info](#)

VPC - *required* [Info](#)

vpc-088febf7f494a9d6b
172.31.0.0/16

Subnet [Info](#)

No preference

Auto-assign public IP [Info](#)

Enable

Firewall (security groups) [Info](#)

A security group is a set of firewall rules that control traffic to and from your EC2 instance.

☒ Create security group

Security group name - *required*

web_server_security_group

This security group will be added to all network interfaces created for the instance. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, ., -, /, =, @, _.

Description - *required* [Info](#)

web_server_security_group created 20...

Inbound security groups rules

▼ Security group rule 1 (TCP, 22)

Type [Info](#)

ssh

Source type [Info](#)

Custom

Port range [Info](#)

22

Description - *optional* [Info](#)

ssh_gate-instance-security-group

Prefix lists

- com.amazonaws.us-east-1.dynamodb.pl-02cd2c6b
- com.amazonaws.global.groundstation.pl-0e5696d987d033653
- com.amazonaws.global.cloudfront.origin-facing.pl-3b927c52
- com.amazonaws.us-east-1.s3.pl-63a5400a

Security groups

- default
- sg-06e5d4782f0d606c8
- ssh_gate-instance-security-group sg-0101e454a0c3e335d

Figura 9: Ajustes de red

En la **Figura 9:** Ajustes de red, se observa la creación del grupo de seguridad el cual solo permitirá conexiones desde la máquina “SSH_gate”. Para ello, en vez de permitir el acceso a cualquiera, personalizamos esto para que solo pueda ser accedida desde “SSH_gate” mediante el grupo de seguridad de esta última. Se ajustan los diferentes parámetros como nombre del grupo y descripción y se finaliza la configuración de red.

▼ Network settings

Info

VPC - required

Info

vpc-088feb7f494a9d6b

172.31.0.0/16

(default) ▼

↻

Subnet

Info

No preference ▼

↻

Create new subnet

🔗

Auto-assign public IP

Info

Enable ▼

Firewall (security groups)

Info

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☒ Create security group

☐ Select existing security group

Security group name - required

web_server_security_group

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and _-./()#,@!+=&:()!\$*

Description - required

Info

web_server_security_group created 2022-10-13T11:10:00.000Z

Inbound security groups rules

▼ Security group rule 1 (TCP, 22, sg-0101e454a0c3e335d, SSH access from Gate)

Remove

Type

Info

ssh ▼

Protocol

Info

TCP

Port range

Info

22

Source type

Info

Custom ▼

Source

Info

🔍 Add CIDR, prefix list or security group

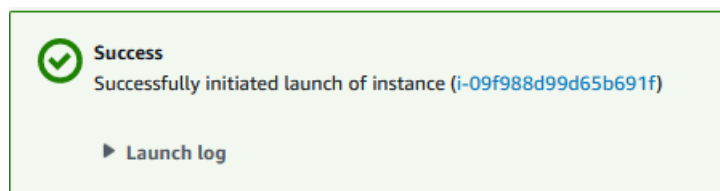
Description - optional

Info

SSH access from Gate

sg-0101e454a0c3e335d

✕



No se realizará ninguna configuración adicional a la máquina, por lo que ya es posible lanzarla, en este caso de forma exitosa como observamos en la **Figura 11: Instancia lanzada correctamente**. El siguiente paso no será otro que configurar el servidor web. Para poder hacer esto, será necesario configurar el servicio SSH para poder conectarnos desde “SSH_gate”

```
[ec2-user@ip-172-31-27-246 ~]$ pwd
/home/ec2-user
[ec2-user@ip-172-31-27-246 ~]$ cd ~/.ssh
[ec2-user@ip-172-31-27-246 .ssh]$ ls
authorized_keys
[ec2-user@ip-172-31-27-246 .ssh]$ touch web_server_priv_key.pem
[ec2-user@ip-172-31-27-246 .ssh]$ ls
authorized_keys  web_server_priv_key.pem
[ec2-user@ip-172-31-27-246 .ssh]$ vi web_server_priv_key.pem
[ec2-user@ip-172-31-27-246 .ssh]$ vi web_server_priv_key.pem
[ec2-user@ip-172-31-27-246 .ssh]$ ls web_server_priv_key.pem
web_server_priv_key.pem
[ec2-user@ip-172-31-27-246 .ssh]$ vi web_server_priv_key.pem
[ec2-user@ip-172-31-27-246 .ssh]$ sudo chmod 700 web_server_priv_key.pem
[ec2-user@ip-172-31-27-246 .ssh]$ ls
authorized_keys  web_server_priv_key.pem
[ec2-user@ip-172-31-27-246 .ssh]$
```

Figura 12: Traslado de clave privada a la máquina

El primer paso será introducir la clave privada que hemos descargado en la máquina “SSH_gate”. Para ello, se crea un archivo con extensión pem donde copiamos el contenido de la clave privada descargada, y se le cambia los permisos al fichero para poder establecer conexión.

```
[ec2-user@ip-172-31-27-246 .ssh]$ ssh -i web_server_priv_key.pem ec2-user@172.31.83.79
The authenticity of host '172.31.83.79 (172.31.83.79)' can't be established.
ECDSA key fingerprint is SHA256:0neMzYzB4A8p+tYWaCVPcNB4xrTiDBaxH/e5ovqN8yQ.
ECDSA key fingerprint is MD5:79:bf:14:92:87:31:d8:c4:9d:96:9d:ab:84:1b:50:2d.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '172.31.83.79' (ECDSA) to the list of known hosts.

  _ | _ | _ )
  _ | ( _ | _ /   Amazon Linux 2 AMI
  _ | \ _ | _ |

https://aws.amazon.com/amazon-linux-2/
4 package(s) needed for security, out of 7 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-83-79 ~]$
```

Figura 13: Conexión SSH al servidor web

Para realizar la conexión lo que se hace es, mediante la clave y la IP privadas de la máquina “web_server”, entablamos una conexión SSH de forma exitosa apreciable en la **Figura 13: Conexión SSH al servidor web**

Una vez la conexión está entablada, desde la máquina “web_server” se actualiza primero todo mediante `yum -y update` y se instala el servicio `httpd` mediante la orden `sudo yum`

`-y install httpd`. Una vez instalado, se debe activar y arrancar el servicio mediante las órdenes:

```
sudo systemctl enable httpd
sudo systemctl start httpd
```

Una vez instalado y configurado el servicio, se cambia los permisos de la carpeta `/var/www/html` para poder escribir un “index” mediante la orden:

```
sudo chmod 777 /var/www/html
```

A continuación, se crea un fichero `index.html` mediante la utilidad `touch`, y se edita poniendo el nombre y la afición favorita. Para probarlo, se crea una nueva regla dentro del grupo de seguridad para poder acceder desde cualquier máquina al puerto 80 del servidor web y encontrar lo solicitado, reflejadas en la **Figura 15: Grupo de seguridad de la máquina que aloja el servidor**.

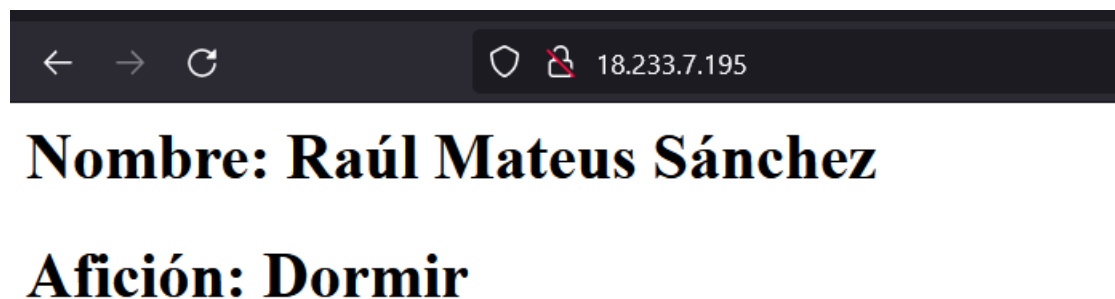


Figura 14: Sitio web accesible desde el exterior

▼ Inbound rules

Security group rule ID	Port range	Protocol	Source	Security groups
sgr-0d2234c3c54cd9806	80	TCP	0.0.0.0/0	web_server_security_group
sgr-0a40d95ea679eec12	22	TCP	sg-0101e454a0c3e335d	web_server_security_group

▼ Outbound rules

Security group rule ID	Port range	Protocol	Destination	Security groups
sgr-030aa2da29cbdd4e	All	All	0.0.0.0/0	web_server_security_group

Figura 15: Grupo de seguridad de la máquina que aloja el servidor

3. Actividad extra

Para la actividad extra, se propone crear dentro de la web zonas privadas para ciertos usuarios declarados, que bien estos usuarios podrían corresponder con distintas instancias, para gestionar su contenido privado como archivos de descarga.

En primer lugar, se modifica el fichero `/etc/httpd/conf/httpd.conf`, añadiendo varias sentencias que permitan la viabilidad de la propuesta

```
# Supplemental configuration
#
# Load config files in the "/etc/httpd/conf.d" directory, if any.
IncludeOptional conf.d/*.conf

KeepAlive On
KeepAliveTimeout 3600
ServerTokens prod
MaxKeepAliveRequests 500
TimeOut 300
LogLevel crit

# SSI(Server Side Includes)

<Directory "/var/www/html/www/">
    Options +Includes
    AddType text/html .html
    AddOutputFilter INCLUDES .html
</Directory>

<Directory "/home/*/">
    Options +Includes
    AddType text/html .html
    AddOutputFilter INCLUDES .html
</Directory>
~
~
-- INSERT --
```

Figura 16: Modificación del fichero `httpd.conf`

En resumen, se añaden varias cláusulas para mejorar el rendimiento del sitio web, así como para garantizar conexiones persistentes. Además, se permite la ejecución de SSI, las cuales son un conjunto de directivas que se escriben en los ficheros HTML y permite añadir contenido dinámico sin necesidad de otras tecnologías como JavaScript o PHP.

Para que los usuarios puedan autenticarse en la zona de descargas o zona privada a la que tengan acceso, se debe volver a modificar el fichero `httpd.conf`

```
<Directory "/var/www/html/private-zone/">
    AuthType Basic
    AuthName "Archivos protegidos. Introduzca su identificacion privada"
    # (Following line optional)
    AuthBasicProvider file
    AuthUserFile "/usr/local/apache/passwd/passwords"
    Require valid-user
</Directory>
```

Figura 17: Protección de la zona privada

En este caso, se implanta una zona de descargas para usuarios autenticados mediante las cláusulas reflejadas en la **Figura 17: Protección de la zona privada** y añadidas al fichero de configuración general `httpd.conf`.

Llegados a este punto, la pregunta es: ¿Quiénes son los usuarios autorizados? Eso se puede definir mediante la utilidad `htpasswd`, la cual nos permitirá añadir usuarios autorizados con una contraseña en un fichero, y usar ese fichero para comprobar qué usuario está intentando entrar. Para ello, se ejecuta la orden:

```
htpasswd -c /usr/local/apache/passwd/passwords ul
```

Una vez tenemos los usuarios, creamos un directorio bajo `/var/www/html` llamado `private-zone`, y pondremos tres ficheros de prueba:

```
[ec2-user@ip-172-31-83-79 var]$ cd /var/www/html
[ec2-user@ip-172-31-83-79 html]$ ls
index.html
[ec2-user@ip-172-31-83-79 html]$ mkdir private-zone
[ec2-user@ip-172-31-83-79 html]$ ls
index.html private-zone
[ec2-user@ip-172-31-83-79 html]$ cd private-zone
[ec2-user@ip-172-31-83-79 private-zone]$ touch prueba1.txt
[ec2-user@ip-172-31-83-79 private-zone]$ touch prueba2.txt
[ec2-user@ip-172-31-83-79 private-zone]$ touch prueba3.txt
[ec2-user@ip-172-31-83-79 private-zone]$ kls
-bash: kls: command not found
[ec2-user@ip-172-31-83-79 private-zone]$ ls
prueba1.txt prueba2.txt prueba3.txt
[ec2-user@ip-172-31-83-79 private-zone]$
```

Figura 18: Creación de la zona privada

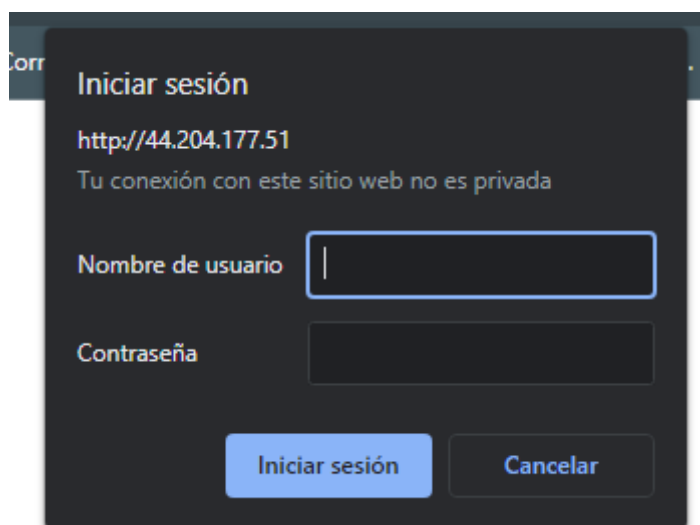
A continuación, desde el propio `index.html` que se muestra al acceder a la IP pública desde el navegador, añadiremos un enlace a esta zona privada donde solicitará al usuario identificarse.

Nombre: Raúl Mateus Sánchez

Afición: Dormir

[Acceder a la zona privada](#)

Figura 19: Nueva apariencia de la web

A dark-themed login dialog box with a title bar. The title is "Iniciar sesión". Below the title is the URL "http://44.204.177.51" and a warning message "Tu conexión con este sitio web no es privada". There are two input fields: "Nombre de usuario" and "Contraseña". At the bottom, there are two buttons: "Iniciar sesión" and "Cancelar".

Corr

Iniciar sesión

http://44.204.177.51

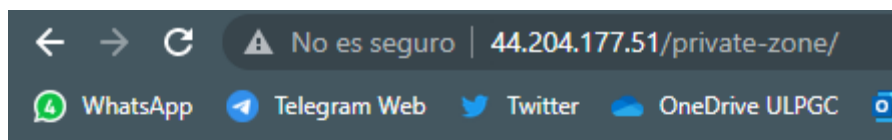
Tu conexión con este sitio web no es privada

Nombre de usuario

Contraseña

Iniciar sesión **Cancelar**

Figura 20: Solicitud de identificación



Index of /private-zone





<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
 Parent Directory		-	
 prueba1.txt	2022-10-13 17:23	0	
 prueba2.txt	2022-10-13 17:23	0	
 prueba3.txt	2022-10-13 17:23	0	

Figura 21: Zona de descarga para usuarios identificados

Por otra parte, teniendo en cuenta la dinamicidad de las IP públicas asignadas a las instancias, realizaremos una redirección de cualquier conexión HTTP por el puerto 80 a conexiones HTTPS por el puerto 443. Para ello, ejecutamos la siguiente orden:

```
yum install mod_ssl
```

Esta orden instala el soporte necesario para SSL v3 y TLS 1.x para el servidor Apache, aunque los navegadores no lo reconocerán como seguro al no tener un certificado verificado por un tercero. Por otra parte, para que el servidor acepte conexiones por el puerto 443, debemos configurar en el panel de control de la instancia, desde AWS, esta nueva regla dentro de su grupo de seguridad.

▼ Inbound rules

<input type="text" value="Filter rules"/>				
Security group rule ID	Port range	Protocol	Source	Security groups
sgr-0d2234c3c54cd9806	80	TCP	0.0.0.0/0	web_server_security_group
sgr-0d7173903f75bbfce	443	TCP	0.0.0.0/0	web_server_security_group
sgr-0a40d95ea679eec12	22	TCP	sg-0101e454a0c3e335d	web_server_security_group

▼ Outbound rules

<input type="text" value="Filter rules"/>				
Security group rule ID	Port range	Protocol	Destination	Security groups
sgr-030aa2da29cbdd4e	All	All	0.0.0.0/0	web_server_security_group

Figura 22: Grupo de seguridad del servidor web actualizado

El siguiente paso será configurar el servidor para redirigir cualquier petición por el puerto 80 (HTTP) al puerto 443 (HTTPS). Para ello, modificamos el fichero `/etc/httpd/conf/httpd.conf` donde, mediante la cláusula *Redirect* permanente se podrá lograr esta redirección.

```
*
# Load config files in the "/etc/httpd/conf.d" directory, if any.
IncludeOptional conf.d/*.conf

KeepAlive On
KeepAliveTimeout 3600
ServerTokens prod
MaxKeepAliveRequests 500
Timeout 300
LogLevel crit

<VirtualHost *:80>
  Redirect permanent / https://54.89.61.134
</VirtualHost>

<VirtualHost *:443>
  DocumentRoot "/var/www/html"
# SSI(Server Side Includes)

<Directory "/var/www/html/www/">
  Options +Includes
  AddType text/html .html
  AddOutputFilter INCLUDES .html
</Directory>

<Directory "/home/*/">
  Options +Includes
  AddType text/html .html
  AddOutputFilter INCLUDES .html
</Directory>

<Directory "/var/www/html/private-zone/">
  AuthType Basic
  AuthName "Archivos protegidos. Introduzca su identificacion privada"
# (Following line optional)
  AuthBasicProvider file
  AuthUserFile "/usr/local/apache/passwd/passwords"
  Require valid-user
</Directory>

SSLEngine on
SSLCertificateFile "/etc/pki/tls/certs/dominio.tld.crt"
SSLCertificateKeyFile "/etc/pki/tls/private/dominio.tld.key"
</VirtualHost>
[
"httpd.conf" 405L, 12906B
```

Figura 23: Redirección del puerto 80 al 443

Al hacer la redirección es visible como se ha activado SSL, así como especificado certificados y llaves. Para generarlos a modo de ejemplo, se ejecutan las órdenes:

```
openssl req -sha256 -x509 -nodes -newkey rsa:4096 -days 1825 -out  
    /etc/pki/tls/certs/dominio.tld.crt -keyout  
    /etc/pki/tls/private/dominio.tld.key  
  
chmod 600 /etc/pki/tls/certs/dominio.tld.crt  
    /etc/pki/tls/private/dominio.tld.key
```

Por otra parte, para este ejemplo se ha dado por supuesto que la instancia que aloja el servidor web deberá estar dando servicio constantemente, por lo que mientras no se reinicia la máquina, no cambiará su IP pública. En cualquier caso, no se debe nunca confiar en estos supuestos y si quisiéramos desplegarlo a mayor nivel se debería ejecutar un script en cada reinicio del sistema que obtenga la IP pública de la instancia y la actualice en este fichero, mediante utilidades como `dig` o `aws`.

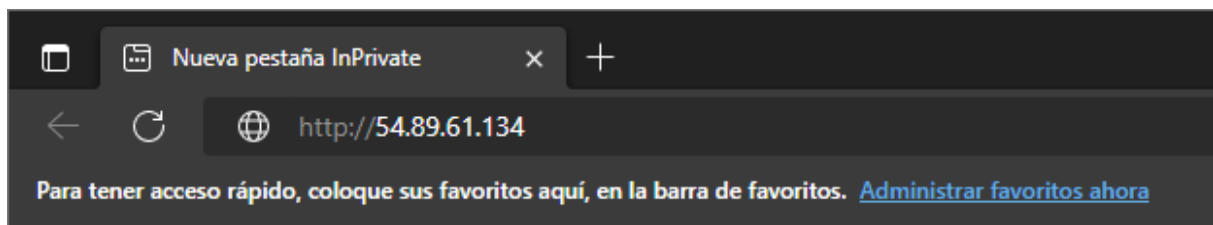


Figura 24: Conexión mediante HTTP



Figura 25: Redirección a HTTPS

A modo de prueba, se intenta conectar con el servidor web después de reiniciarlo con los nuevos cambios, visible en la **Figura 24:** Conexión mediante HTTP, pero automáticamente redirige a HTTPS mostrando la web, apreciable en la **Figura 25:** Redirección a HTTPS.

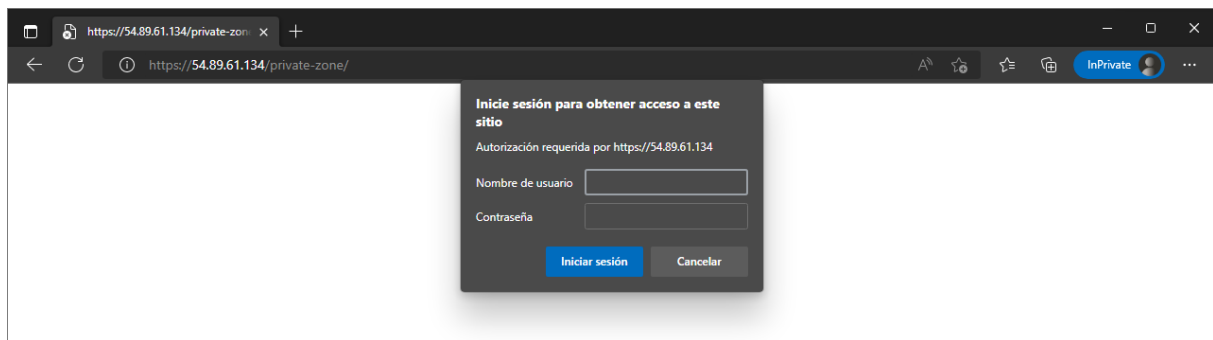


Figura 26: Autenticación de usuarios

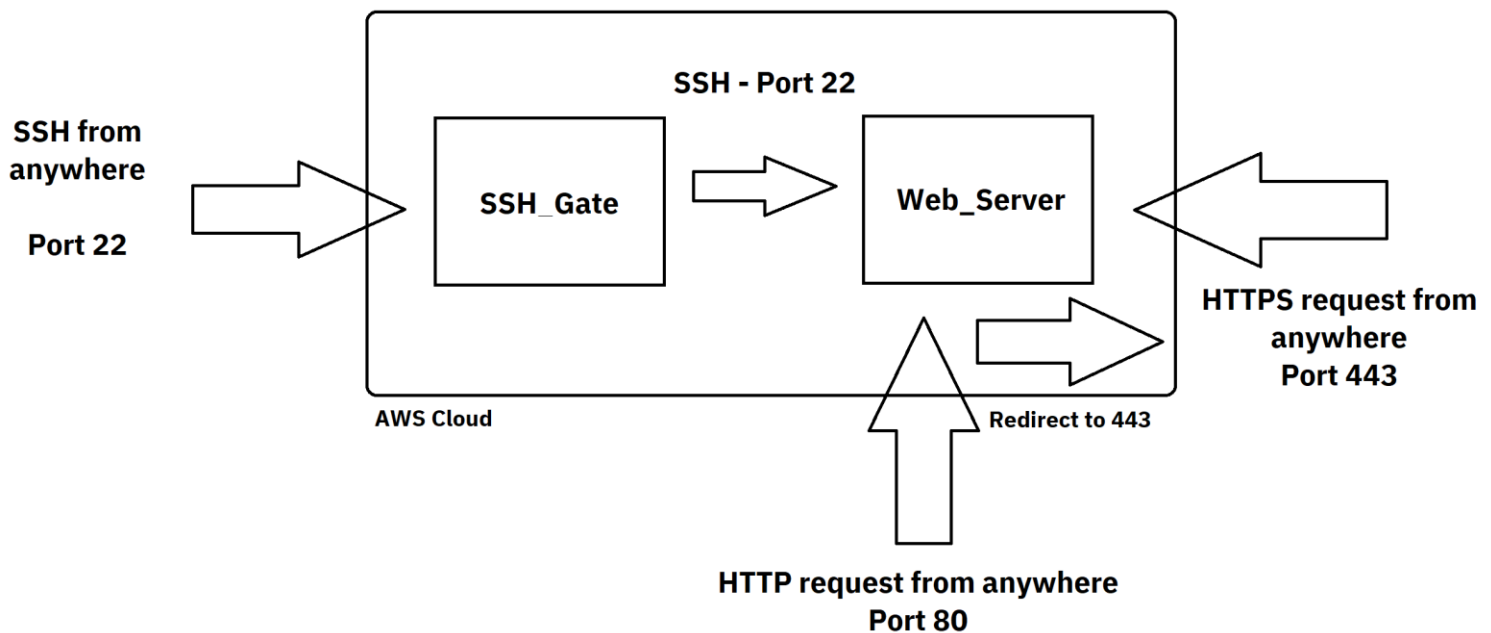


Figura 27: Zona privada segura

Por último, se prueba el acceso a la zona de descarga, apreciando que sigue requiriendo una identificación y además ahora está bajo peticiones por el puerto 443 o HTTPS.

Diagrama de arquitectura desplegada

Se muestra el diagrama de la arquitectura desplegada:



La arquitectura desplegada corresponde a dos instancias creadas bajo el paraguas de AWS, concretamente instancias EC2. Una de ellas, **SSH_Gate**, servirá de pasarela para acceder a la máquina que aloje el servidor web, siendo esta la única que pueda acceder mediante SSH. Por otro lado, a **SSH_Gate** puede tratar de acceder todo el mundo, pero solo lo logrará quien tenga la llave correspondiente.

Por otro lado, al servidor web se le podrá hacer solicitudes HTTP desde cualquier parte, aunque si son por el puerto 80 serán redirigidos al 443.

Presupuesto y estimación de gasto de los recursos desplegados

Después de analizar la arquitectura desplegada, se debe tener en cuenta, aparte del gasto por hora que supone tener en marcha cada una de las instancias y el presupuesto, la finalidad de cada una de estas y si estarán siempre dando servicio o no. Es obvio que el servidor web, si se tratase de un caso real, debe estar dando servicio de forma continua, por lo que no variará su precio por hora.

Por otro lado, contamos con otra instancia que sirve de pasarela hacia el servidor web, la cual no tiene por qué estar en funcionamiento siempre, ya que una vez configurado el servidor web, accederemos a esta misma a través de la pasarela para cuestiones puntuales de configuración, mantenimiento o revisión.

En cuanto al presupuesto y coste de las instancias por horas, se ha desplegado dos instancias bajo Amazon Linux en su tipo más barato, 0,0116\$ por hora. Por otra parte, el presupuesto inicial con el que contamos es de 100\$, pero para acercarnos más a la realidad, se debe tener en cuenta que a lo largo de la asignatura se realizarán 8 prácticas las cuales, probablemente, reúnan cada vez más recursos y gastos. No se maneja un porcentaje concreto de gasto que nos supondrá cada práctica, pero en líneas generales, es posible fijar un tope en el presupuesto de 12,50\$ (dividiendo 100/8), reduciéndolo a 10\$ teniendo en cuenta que una vez concluyan las pruebas y su defensa, no se deberían volver a arrancar.

Teniendo todo esto en cuenta, y fijando un plazo de realización y defensa de la práctica en dos semanas desde su comienzo contabilizando el uso de las instancias en un rango de 2 a 3 horas por día (42 horas en total), podemos determinar una estimación de gasto máximo de $(0.0116 \times 2) \times 42 = 0.97\$$. Este probablemente sea la estimación real dadas las circunstancias de la práctica, su dificultad y su posterior defensa.

Se podría tener en cuenta ya el propio desarrollo de la práctica, pero al ser una estimación y no un análisis del coste generado, se debe trabajar anticipándonos al gasto que pueda ocasionar el uso de las instancias.

En un caso real, donde se diese un servicio al exterior de forma constante, y un presupuesto de 100\$ en total, se concretaría que la instancia que sirve de pasarela se usará la mitad que el propio servidor web, por lo que nos daría para dar un servicio un total de, aproximadamente, 5747 horas, o lo que es lo mismo, 7 meses llegando a casi 8 de servicio, teniendo en cuenta estas circunstancias.

Conclusiones

En esta primera toma de contacto con los servicios de AWS, se ha aprendido a crear y manejar instancias bajo el servicio EC2, así como se ha aprendido sus características, coste y limitaciones. Una de estas limitaciones viene dada por la dinamicidad de la IP pública, puesto que dificulta la puesta en marcha de muchos servicios hacia el exterior si no queremos pagar un poco más.

En definitiva, esta ha sido una guía para adentrarse en la nube y su uso comercial y empresarial, y nos permitirá desarrollar futuras prácticas con mucha mayor soltura.