

INSTALACIÓN WORDPRESS EN AWS



Sumario

1. Introducción.....	3
2. Creacion de instancias.....	4
3. Descargar Apache y PHP.....	6
4. Crear Base de Datos.....	8
5. Elastic File System.....	13
6. Descarga de Wordpress.....	15

1. Introducción

Vamos a los servicios de VPC y creamos una VPC para las subredes.

Create VPC [Info](#)

A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances. Mouse over a resource to highlight the related resources.

VPC settings

Resources to create [Info](#)
Create only the VPC resource or the VPC and other networking resources.

☐ VPC only ☒ VPC and more

Name tag auto-generation [Info](#)
Enter a value for the Name tag. This value will be used to auto-generate Name tags for all resources in the VPC.

☒ Auto-generate
vpc-wordpress

IPv4 CIDR block [Info](#)
Determine the starting IP and the size of your VPC using CIDR notation.

10.0.0.0/16 65,536 IPs
CIDR block size must be between /16 and /28.

IPv6 CIDR block [Info](#)
☒ No IPv6 CIDR block
☐ Amazon-provided IPv6 CIDR block

Tenancy [Info](#)
Default

► **Encryption settings - optional**

Preview

VPC [Show details](#)
Your AWS virtual network

vpc-wordpress-vpc

Subnets (4)
Subnets within this VPC

us-east-1a

- A vpc-wordpress-sl
- A vpc-wordpress-sl

us-east-1b

- B vpc-wordpress-sl
- B vpc-wordpress-sl

Le pondremos 2 subredes publicas y 2 privadas. Ademas es recomendable ponerle un NAT gateway.

Number of Availability Zones (AZs) [Info](#)
Choose the number of AZs in which to provision subnets. We recommend at least two AZs for high availability.

1 | 2 | 3

► **Customize AZs**

Number of public subnets [Info](#)
The number of public subnets to add to your VPC. Use public subnets for web applications that need to be publicly accessible over the internet.

0 | 2

Number of private subnets [Info](#)
The number of private subnets to add to your VPC. Use private subnets to secure backend resources that don't need public access.

0 | 2 | 4

► **Customize subnets CIDR blocks**

NAT gateways (\$) - updated [Info](#)
NAT gateway allows private resources to access the internet from any availability zone within a VPC, providing a single managed internet exit point for the entire region. Additional charges apply.

None | **Regional - new** | Zonal

Introducing regional NAT gateway [×](#)
AWS now offers a multi-AZ NAT Gateway, eliminating the need for separate NAT Gateways across availability zones.

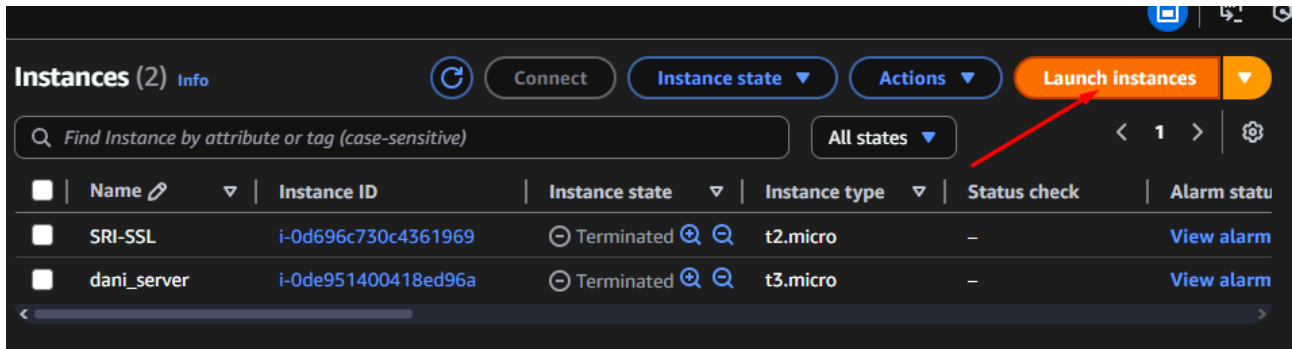
Preview

VPC [Show details](#)
Your AWS virtual network

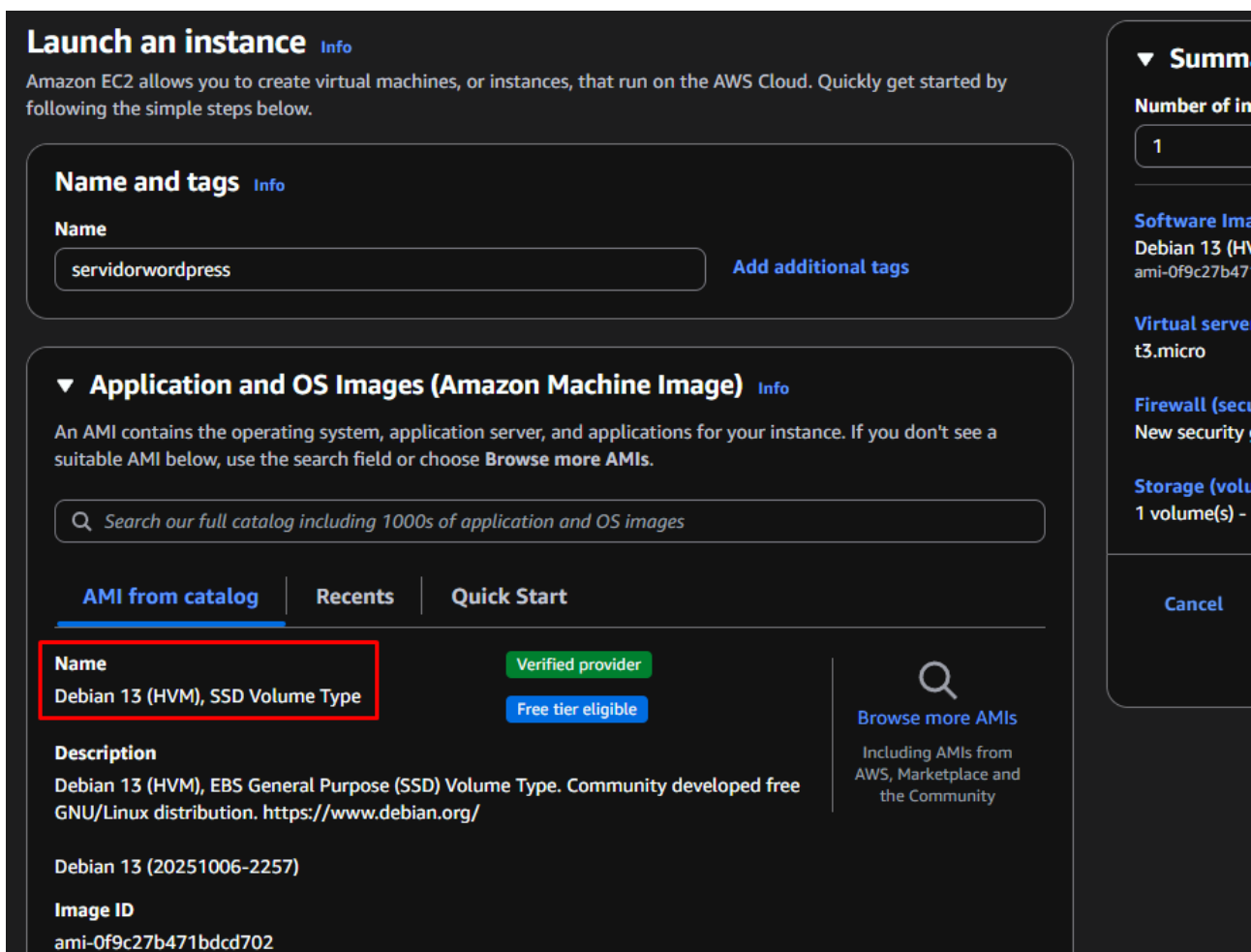
vpc-wordpress-vpc

2. Creacion de instancias

Vamos a ir a EC2 > Instancias. Aqui le daremos a Launch Instance



A la instancia le pondremos un nombre y elegiremos el sistema operativo Debian.



Aparte, necesitaremos crear un grupo de seguridad con la siguiente configuración:

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*

seguridadwordpress

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

launch-wizard-3 created 2025-12-18T13:05:44.824Z

Inbound Security Group Rules

▼ Security group rule 1 (TCP, 22, 212.104.181.12/32, http) Remove

Type Info	Protocol Info	Port range Info
ssh	TCP	22
Source type Info	Source Info	Description - <i>optional</i> Info
Custom	Q Add CIDR, prefix list or security group	http
	212.104.181.12/32 X	

▼ Security group rule 2 (TCP, 80, 0.0.0.0/0, Apertura puerto 80) Remove

Type Info	Protocol Info	Port range Info
HTTP	TCP	80
Source type Info	Source Info	Description - <i>optional</i> Info
Anywhere	Q Add CIDR, prefix list or security group	Apertura puerto 80
	0.0.0.0/0 X	

Summary

Number of instances | Info

1

Software Image (AMI)

Debian 13 (HVM), SSD Volume Type...read more
ami-0f9c27b471bdcd702

Virtual server type (instance type)

t2.micro

Firewall (security group)

New security group

Storage (volumes)

1 volume(s) - 8 GiB

Cancel Launch instance Preview code

Despues de eso, la instancia ya estará correctamente creada, faltaría desde el cmd desde el PC, conectarnos a la instancia con SSH.

```
C:\Users\Usuario>cd .ssh

C:\Users\Usuario\.ssh>ssh -i "wordpress.pem" admin@ec2-52-91-244-202.compute-1.amazonaws.com
The authenticity of host 'ec2-52-91-244-202.compute-1.amazonaws.com (52.91.244.202)' can't be established.
ED25519 key fingerprint is SHA256:DeHglcWNVXDTEfom3Khw8ngHDzkhoRallcxYYEq9ZE.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-52-91-244-202.compute-1.amazonaws.com' (ED25519) to the list of known hosts.
Linux ip-172-31-21-194 6.12.48+deb13-cloud-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.12.48-1 (2025-09-20) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
admin@ip-172-31-21-194:~$
```

3. Descargar Apache y PHP

Hacemos un update para recibir los paquetes que nos falten.

```
admin@ip-172-31-21-194: ~  
admin@ip-172-31-21-194:~$ sudo apt update
```

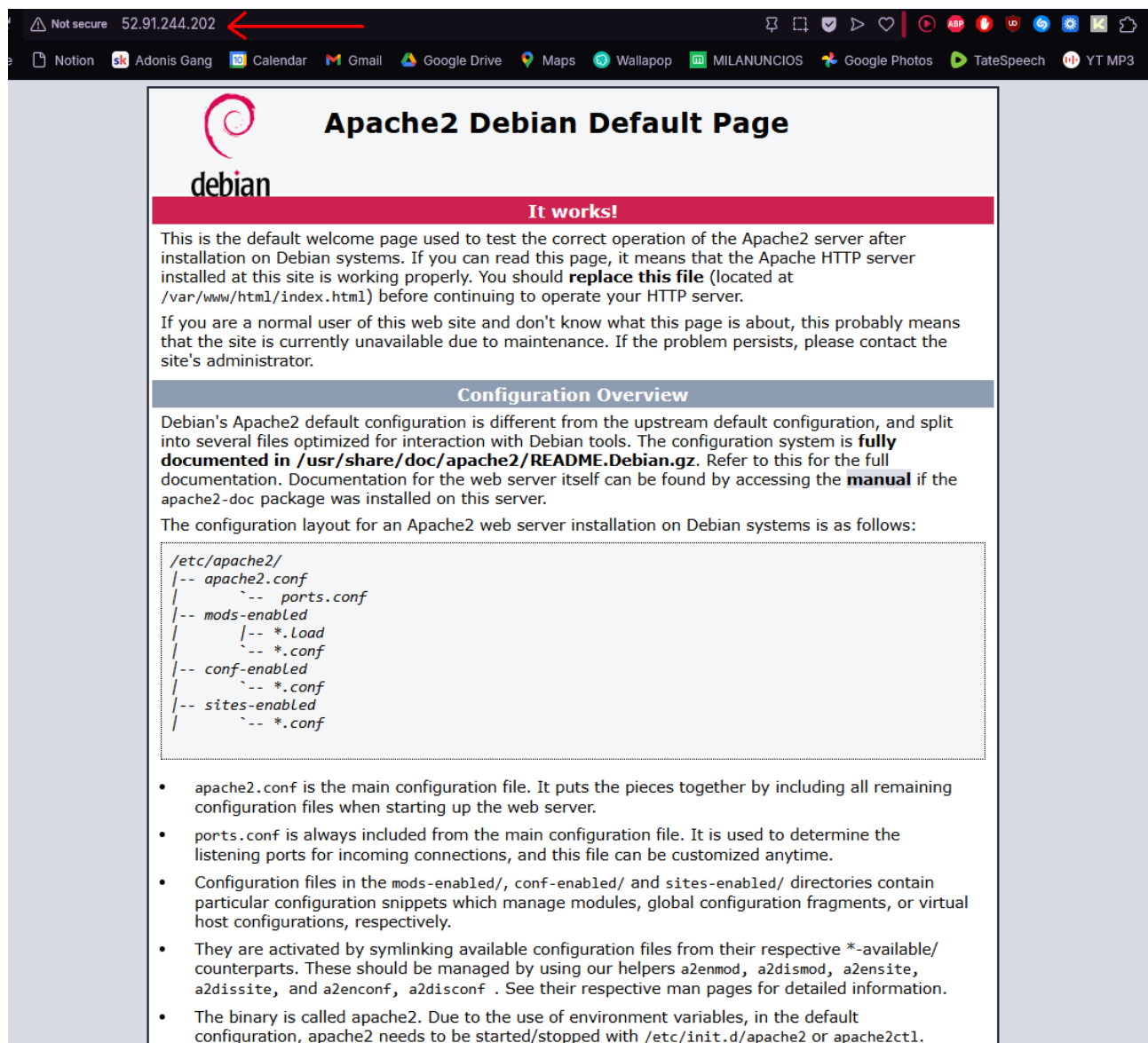
Despues instalamos el servidor apache con este comando:

```
admin@ip-172-31-21-194: ~  
admin@ip-172-31-21-194:~$ sudo apt install apache2
```

Despues lo iniciamos y comprobamos que esta “running”.

```
admin@ip-172-31-21-194: ~  
admin@ip-172-31-21-194:~$ sudo systemctl start apache2  
admin@ip-172-31-21-194:~$ sudo systemctl enable apache2  
Synchronizing state of apache2.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.  
Executing: /usr/lib/systemd/systemd-sysv-install enable apache2  
admin@ip-172-31-21-194:~$ sudo systemctl status apache2  
● apache2.service - The Apache HTTP Server  
   Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)  
   Active: active (running) since Thu 2025-12-18 13:52:25 UTC; 39s ago  
 Invocation: abded8e743824132a-13b8af6b370053  
    Docs: https://httpd.apache.org/docs/2.4/  
  Main PID: 1392 (apache2)  
    Tasks: 55 (limit: 1127)  
  Memory: 5.4M (peak: 5.6M)  
     CPU: 33ms  
   CGroup: /system.slice/apache2.service  
           └─1392 /usr/sbin/apache2 -k start  
             └─1394 /usr/sbin/apache2 -k start  
               └─1395 /usr/sbin/apache2 -k start  
  
Dec 18 13:52:25 ip-172-31-21-194 systemd[1]: Starting apache2.service - The Apache HTTP Server...  
Dec 18 13:52:25 ip-172-31-21-194 apachectl[1391]: AH00558: apache2: Could not reliably determine the server's f  
Dec 18 13:52:25 ip-172-31-21-194 systemd[1]: Started apache2.service - The Apache HTTP Server.  
lines 1-17/17 (END)
```

Si vamos al navegador y ponemos la direccion IP de la instancia, veremos que nuestro servidor apache esta activo.



Not secure 52.91.244.202

Notion Adonis Gang Calendar Gmail Google Drive Maps Wallapop MILANUNCIOS Google Photos TateSpeech YT MP3

Apache2 Debian Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Debian systems. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Debian's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Debian tools. The configuration system is **fully documented in `/usr/share/doc/apache2/README.Debian.gz`**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Debian systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|   |-- ports.conf
|-- mods-enabled
|   |-- *.load
|   |-- *.conf
|-- conf-enabled
|   |-- *.conf
|-- sites-enabled
|   |-- *.conf
```

- `apache2.conf` is the main configuration file. It puts the pieces together by including all remaining configuration files when starting up the web server.
- `ports.conf` is always included from the main configuration file. It is used to determine the listening ports for incoming connections, and this file can be customized anytime.
- Configuration files in the `mods-enabled/`, `conf-enabled/` and `sites-enabled/` directories contain particular configuration snippets which manage modules, global configuration fragments, or virtual host configurations, respectively.
- They are activated by symlinking available configuration files from their respective `*-available/` counterparts. These should be managed by using our helpers `a2enmod`, `a2dismod`, `a2ensite`, `a2dissite`, and `a2enconf`, `a2disconf`. See their respective man pages for detailed information.
- The binary is called `apache2`. Due to the use of environment variables, in the default configuration, `apache2` needs to be started/stopped with `/etc/init.d/apache2` or `apache2ctl`.

Tambien es conveniente obtener PHP, para ello deberiamos hacer otro update y upgrade.

```
admin@ip-172-31-21-194: ~
admin@ip-172-31-21-194:~$ sudo apt -y update && sudo apt upgrade
```

Luego descargamos PHP con un modulo de apache.

```
admin@ip-172-31-21-194: ~  
admin@ip-172-31-21-194:~$ sudo apt install php libapache2-mod-php php-cli  
Installing:  
libapache2-mod-php php php-cli
```

Es necesario descargar mysql como modulo de apache:

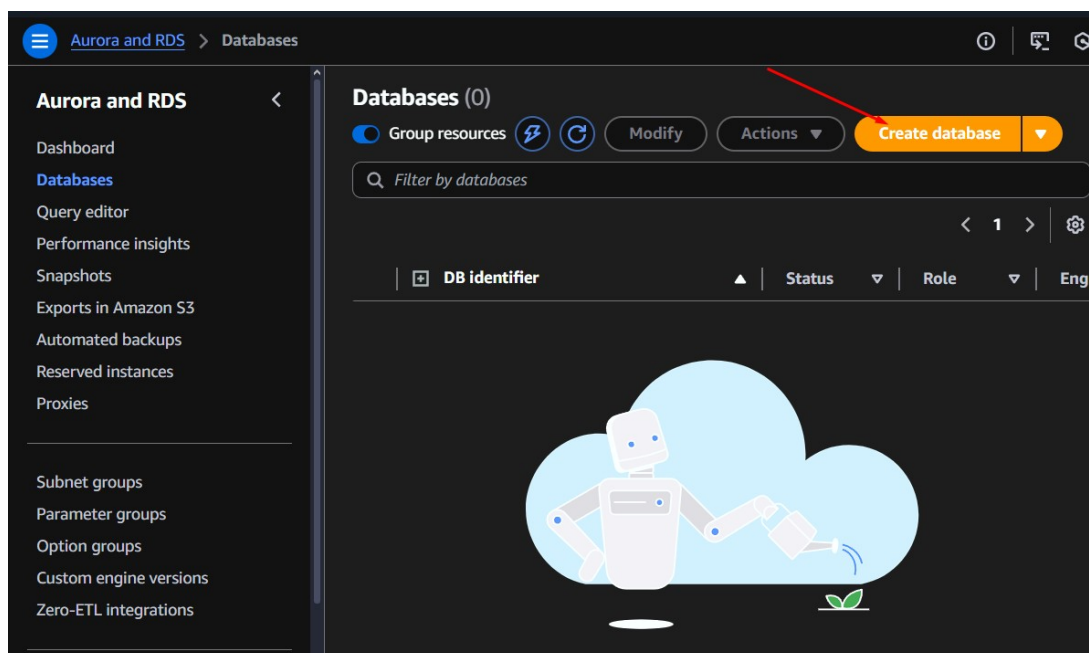
```
admin@ip-172-31-21-194: ~  
admin@ip-172-31-21-194:~$ sudo apt install php-mysql
```

Luego para aplicar los cambios, reiniciamos el server de apache y comprobamos si PHP ha sido instalado correctamente con “php -v”.

```
admin@ip-172-31-21-194: ~  
admin@ip-172-31-21-194:~$ sudo systemctl restart apache2  
admin@ip-172-31-21-194:~$ php -v  
PHP 8.4.11 (cli) (built: Aug 3 2025 07:32:21) (NTS)  
Copyright (c) The PHP Group  
Built by Debian  
Zend Engine v4.4.11, Copyright (c) Zend Technologies  
with Zend OPcache v8.4.11, Copyright (c), by Zend Technologies  
admin@ip-172-31-21-194:~$
```

4. Crear Base de Datos

En el servicio de RDS, le damos a Databases > Create Database.



Elegiremos una BD de MySQL.


Choose a database creation method


☒ **Full configuration**
You set all of the configuration options, including ones for availability, security, backups, and maintenance.


☐ **Easy create**
Use recommended best-practice configurations. Some configuration options can be changed after the database is created.


Engine options


Engine type [Info](#)


☐ Aurora (MySQL Compatible)



☐ Aurora (PostgreSQL Compatible)



☒ MySQL


☐ PostgreSQL


☐ MariaDB


☐ Oracle


☐ Microsoft SQL Server


☐ IBM Db2


Seleccionaremos el Template “Sandbox” y que solo cree una “DB instance deployment”.

Templates

Choose a sample template to meet your use case.

☐ **Production**
Use defaults for high availability and fast, consistent performance.

☐ **Dev/Test**
This instance is intended for development use outside of a production environment.

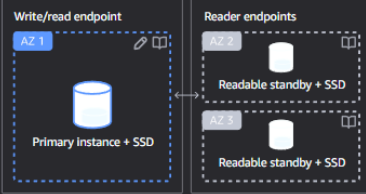
☒ **Sandbox**
To develop new applications, test existing applications, or gain hands-on experience with Amazon RDS.

Availability and durability

Deployment options [Info](#)
Choose the deployment option that provides the availability and durability needed for your use case. AWS is committed to a certain level of uptime depending on the deployment option you choose. Learn more in the [Amazon RDS service level agreement \(SLA\)](#).

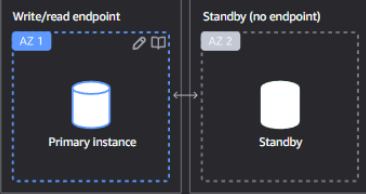
☐ **Multi-AZ DB cluster deployment (3 instances)**
Creates a primary DB instance with two readable standbys in separate Availability Zones. This setup provides:

- 99.95% uptime
- Redundancy across Availability Zones
- Increased read capacity
- Reduced write latency



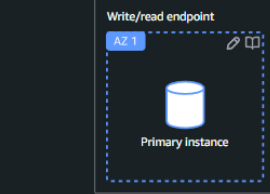
☐ **Multi-AZ DB instance deployment (2 instances)**
Creates a primary DB instance with a non-readable standby instance in a separate Availability Zone. This setup provides:

- 99.95% uptime
- Redundancy across Availability Zones



☒ **Single-AZ DB instance deployment (1 instance)**
Creates a single DB instance without standby instances. This setup provides:

- 99.5% uptime
- No data redundancy



Le escribiremos un nombre, el usuario admin y una contraseña que queramos.

Settings

DB instance identifier [Info](#)
Type a name for your DB instance. The name must be unique across all DB instances owned by your AWS account in the current AWS Region.

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 63 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

▼ **Credentials Settings**

Master username [Info](#)
Type a login ID for the master user of your DB instance.

1 to 16 alphanumeric characters. The first character must be a letter.

Credentials management
You can use AWS Secrets Manager or manage your master user credentials.

☒ **Managed in AWS Secrets Manager - most secure**
RDS generates a password for you and manages it throughout its lifecycle using AWS Secrets Manager.

☒ **Self managed**
Create your own password or have RDS create a password that you manage.

☐ **Auto generate password**
Amazon RDS can generate a password for you, or you can specify your own password.

Master password [Info](#)

Password strength **Strong**
Minimum constraints: At least 8 printable ASCII characters. Can't contain any of the following symbols: / ' * @
Confirm master password [Info](#)

En la configuración de la instancia le ponemos que tenga un “db.t3.micro” con un SSD de 200 GB.

Instance configuration

The DB instance configuration options below are limited to those supported by the engine that you selected above.

DB instance class [Info](#)
▼ **Hide filters**

☐ **Show instance classes that support Amazon RDS Optimized Writes** [Info](#)
Amazon RDS Optimized Writes improves write throughput by up to 2x at no additional cost.

☐ **Include previous generation classes**

☐ Standard classes (includes m classes)

☐ Memory optimized classes (includes r and x classes)

☒ Burstable classes (includes t classes)

db.t3.micro

2 vCPUs 1 GiB RAM EBS Bandwidth: Up to 2,085 Mbps Network: Up to 5 Gbps

Storage

Storage type [Info](#)
Provisioned IOPS SSD (io2) storage volumes are now available.

General Purpose SSD (gp2)
Baseline performance determined by volume size

Allocated storage [Info](#)

GiB

Allocated storage value must be 20 GiB to 6,144 GiB

► **Additional storage configuration**

No hace falta que se conecte a ningun recurso EC2.

Connectivity Info

Compute resource

Choose whether to set up a connection to a compute resource for this database. Setting up a connection will automatically change connectivity settings so that the compute resource can connect to this database.

☒ **Don't connect to an EC2 compute resource**
Don't set up a connection to a compute resource for this database. You can manually set up a connection to a compute resource later.

☐ **Connect to an EC2 compute resource**
Set up a connection to an EC2 compute resource for this database.

Virtual private cloud (VPC) Info

Choose the VPC. The VPC defines the virtual networking environment for this DB instance.

Default VPC (vpc-05a4f460ff5424bba)
6 Subnets, 6 Availability Zones

Only VPCs with a corresponding DB subnet group are listed.

After a database is created, you can't change its VPC.

DB subnet group Info

Choose the DB subnet group. The DB subnet group defines which subnets and IP ranges the DB instance can use in the VPC that you selected.

default

Pondremos que no tenga acceso publico y también elegiremos un nombre para ponerle al nuevo grupo de seguridad.

Public access Info

☐ **Yes**
RDS assigns a public IP address to the database. Amazon EC2 instances and other resources outside of the VPC can connect to your database. Resources inside the VPC can also connect to the database. Choose one or more VPC security groups that specify which resources can connect to the database.

☒ **No**
RDS doesn't assign a public IP address to the database. Only Amazon EC2 instances and other resources inside the VPC can connect to your database. Choose one or more VPC security groups that specify which resources can connect to the database.

VPC security group (firewall) Info

Choose one or more VPC security groups to allow access to your database. Make sure that the security group rules allow the appropriate incoming traffic.

☐ **Choose existing**
Choose existing VPC security groups

☒ **Create new**
Create new VPC security group

New VPC security group name

seguridadbwordpress

Availability Zone Info

No preference

RDS Proxy

RDS Proxy is a fully managed, highly available database proxy that improves application scalability, resiliency, and security.

☐ **Create an RDS Proxy** Info
RDS automatically creates an IAM role and a Secrets Manager secret for the proxy. RDS Proxy has additional costs. For more information, see [Amazon RDS Proxy pricing](#).

Certificate authority - optional Info

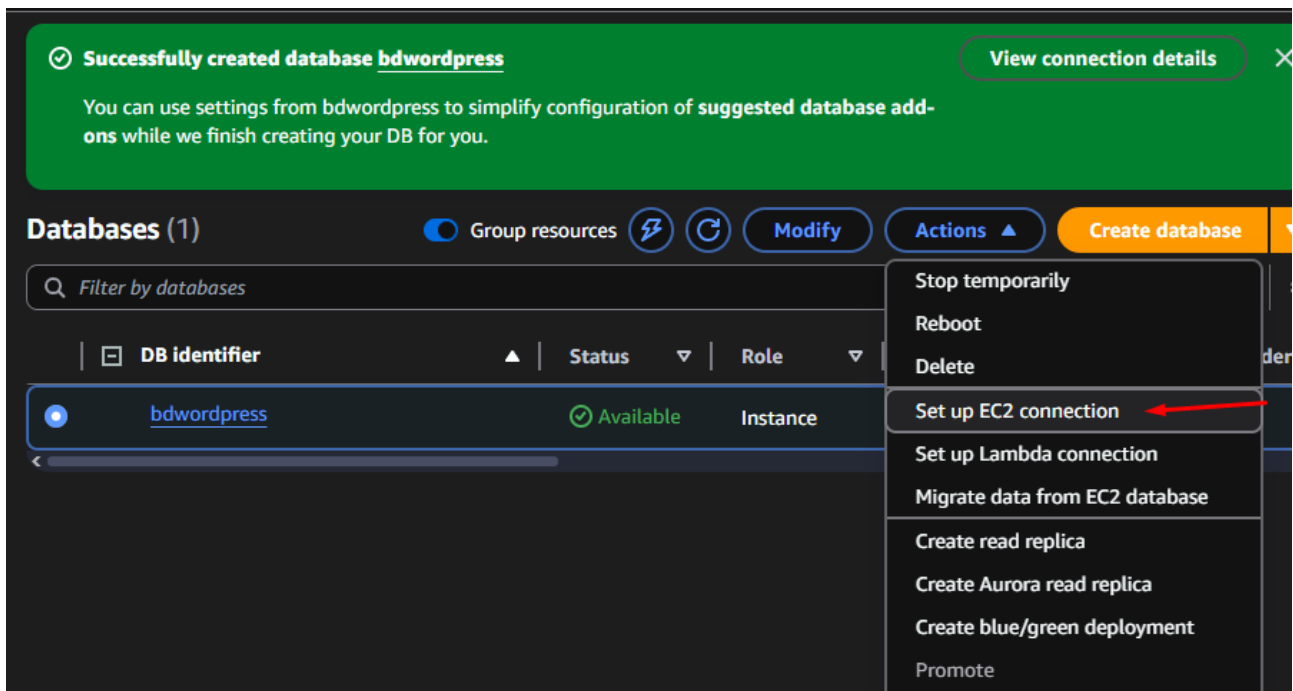
Using a server certificate provides an extra layer of security by validating that the connection is being made to an Amazon database. It does so by checking the server certificate that is automatically installed on all databases that you provision.

rds-ca-rsa2048-g1 (default)
Expiry: May 26, 2061

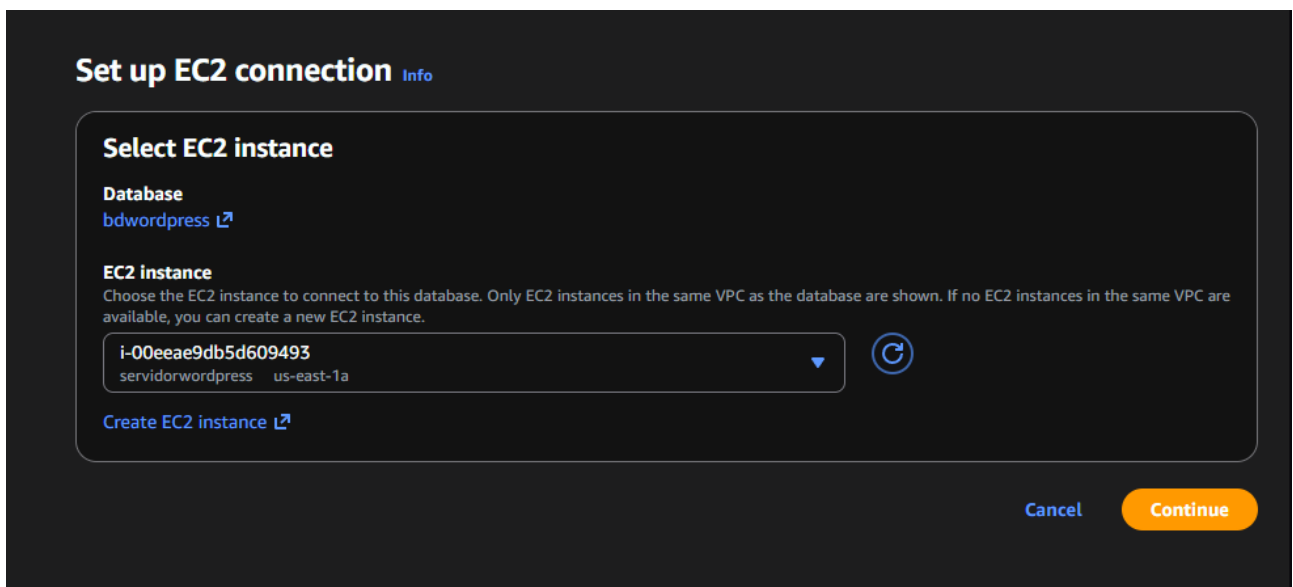
If you don't select a certificate authority, RDS chooses one for you.

► **Additional configuration**

Como podemos comprobar, la base de datos ya esta creada, falta ponerle una conexión EC2, entonces le damos clic derecho > Set up EC2 connection.

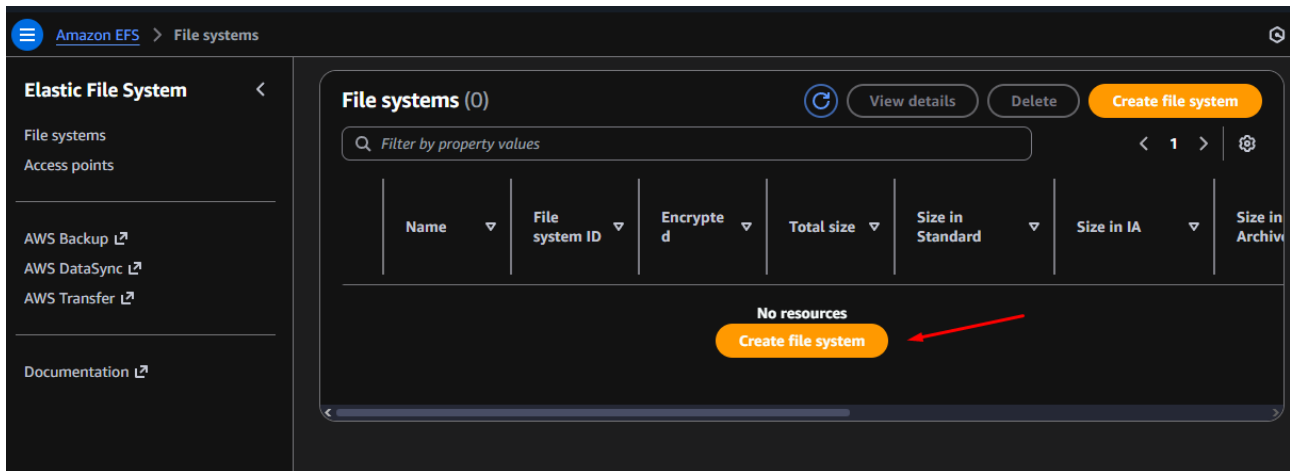


De instancia le pondremos la que habiamos creado al principio.



5. Elastic File System

Vamos a los servicios de EFS y creamos un nuevo sistema de archivos.



Le ponemos un nombre personalizado y elegiremos la VPC que habíamos creado antes.

Create file system

Create a file system with the recommended settings shown below by choosing Create file system. To view all settings or to customize your file system, choose Customize. [Learn more](#)

Name - optional
Name your file system.

almacenwordpress

Name can include letters, numbers, and +-=._:/ symbols, up to 256 characters.

Virtual Private Cloud (VPC)
Choose the VPC where you want EC2 instances to connect to your file system.

vpc-033381c64faebf811
vpc-wordpress-vpc

Recommended settings
Your file system is created with the following recommended settings unless you choose to customize the file system. You will be charged for storage and throughput. We recommend reviewing pricing for these features using the [AWS Pricing Calculator](#).

Setting	Value	Editable after creation
Throughput mode Learn more	Elastic	Yes
Transition into Infrequent Access (IA)	30 day(s) since last access	Yes
Transition into Archive	90 day(s) since last access	Yes
Transition into Standard	None	Yes
Automatic backups	Enabled	Yes
Encryption	Enabled	No

[Cancel](#) [Customize](#) [Create file system](#)

Despues iremos a las Ibound Rules de la instancia y crearemos una regla nueva que permita a la instancia del servidor entrar.

Edit inbound rules [Info](#)

Inbound rules control the incoming traffic that's allowed to reach the instance.

Security group rule ID	Type Info	Protocol Info	Port range Info	Source Info	Description - optional Info	
sgr-0a954e1bb37f19571	HTTP	TCP	80	Cu... <input type="text" value="0.0.0.0/0"/>	Apertura puerto 80	Delete
sgr-0a0770194f60f3bb2	SSH	TCP	22	Cu... <input type="text" value="212.104.181.12/32"/>	http	Delete
-	NFS	TCP	2049	Cu... <input type="text" value="52.91.244.202/32"/>	regla instancia	Delete

[Add rule](#)

Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

[Cancel](#) [Preview changes](#) [Save rules](#)

Luego le daremos a que se conecte por IP.

Attach ✕

Mount your Amazon EFS file system on a Linux instance. [Learn more](#)

☒ Mount via DNS ☐ Mount via IP

Using the EFS mount helper:

```
sudo mount -t efs -o tls fs-0a1f5e749519a9206:/ efs
```

Using the NFS client:

```
sudo mount -t nfs4 -o nfsvers=4.1,rsize=1048576,wsiz=1048576,hard,timeo=600,retrans=2,noresvport fs-0a1f5e749519a9206.efs.us-east-1.amazonaws.com:/ efs
```

See our user guide for more information. [Learn more](#)

[Close](#)

Luego ponemos el siguiente comando para montar el recurso.

```
admin@ip-172-31-21-194:~$ sudo mount -t nfs4 -o nfsvers=4.1,rsize=1048576,wsiz=1048576,hard,timeo=600,retrans=2,noresvport 10.0.135.34:/ efs
```

6. Descarga de Wordpress

Luego nos vamos al directorio de html y usamos el comando wget para instalar wordpress.

```
Select admin@ip-172-31-21-194: /var/www/html
admin@ip-172-31-21-194:~$ cd /var/www/html
admin@ip-172-31-21-194:/var/www/html$ wget http://wordpress.org/latest.tar.gz
```

Luego descomprimos lo descargado.

```
admin@ip-172-31-21-194:~$ tar -xf latest.tar.gz
admin@ip-172-31-21-194:~$
```

Luego instalamos el mysql client.

```
admin@ip-172-31-21-194:~$ sudo apt install default-mysql-client
Installing:
  default-mysql-client

Installing dependencies:
  libconfig-inifiles-perl  libdbi-perl  libncurses6  libterm-readkey-perl  mariadb-client-compat  mariadb-common
  libdbd-mariadb-perl      libmariadb3  libpcre2-posix3  mariadb-client  mariadb-client-core  mysql-common

Suggested packages:
  libclone-perl  libmldbm-perl  libnet-daemon-perl  libsql-statement-perl

Summary:
  Upgrading: 0, Installing: 13, Removing: 0, Not Upgrading: 0
  Download size: 5527 kB
  Space needed: 86.0 MB / 6458 MB available

Continue? [Y/n] Y
```

Despues tenemos que conectarnos la base de datos para configurarla de manera que podamos usar wordpress.

```
admin@ip-172-31-21-194:/var/www/html$ sudo mysql -u admin -h bdwordpress.ca2v85hycviw.us-east-1.rds.amazonaws.com -p
```

Finalmente, en el navegador podremos entrar en wordpress y configurar desde la interfaz web.



A continuación debes introducir los detalles de conexión de tu base de datos. Si no estás seguro de esta información contacta con tu proveedor de alojamiento web.

Nombre de la base de datos	<input type="text" value="wordpress"/>	El nombre de la base de datos que quieres usar con WordPress.
Nombre de usuario	<input type="text" value="danibenitez"/>	El nombre de usuario de tu base de datos.
Contraseña	<input type="password" value="wordpressASIR"/>	La contraseña de tu base de datos.
Servidor de la base de datos	<input type="text" value="localhost"/>	Deberías recibir esta información de tu proveedor de alojamiento web, si localhost no funciona.
Prefijo de tabla	<input type="text" value="wp_"/>	Si quieres ejecutar varias instalaciones de WordPress en una sola base de datos cambia esto.