Migración de Base de Datos desde EC2 Privada a Amazon RDS con Respaldo en S3

Objetivo

El objetivo principal de esta actividad es migrar una base de datos local (alojada en una instancia. EC2 privada) a Amazon RDS, creando copias de seguridad en S3 durante el proceso.

Migrar la BBDD local (MySQL en EC2 privada) a RDS:

- Dejar de depender de una base de datos local (poco escalable y sin backups automáticos).
- Usar RDS para obtener alta disponibilidad, escalabilidad y backups gestionados por AWS.

Crear copias de seguridad en S3:

- Guardar un respaldo de la BBDD local antes de migrar a RDS.
- Asegurar que los datos no se pierdan durante la migración.

Configurar WordPress para usar RDS:

- Modificar el archivo wp-config.php para apuntar a la nueva BBDD en RDS.

Realizar la migración de una base de datos local (MySQL en EC2) a Amazon RDS, incluyendo copias de seguridad en S3 usando un rol IAM existente (como LabRole). Haremos pruebas con una base de datos de prueba antes de migrar la real, simulando un entorno profesional.

Pre-requisitos

Recursos en AWS:

- Instancia EC2 pública: Con WordPress instalado (frontal).
- Instancia EC2 privada: Con MySQL local (base de datos actual).
- RDS MySQL: Instancia creada y accesible desde la instancia pública.
- Bucket S3: Para backups.
- Rol IAM: Asignado a las instancias EC2 con permisos para S3.

Credenciales:

- Usuario MySQL local: root / Passw0rdRootAcc0unt.
- Usuario RDS: admin / Passw0rdRootAcc0unt.

Recursos:

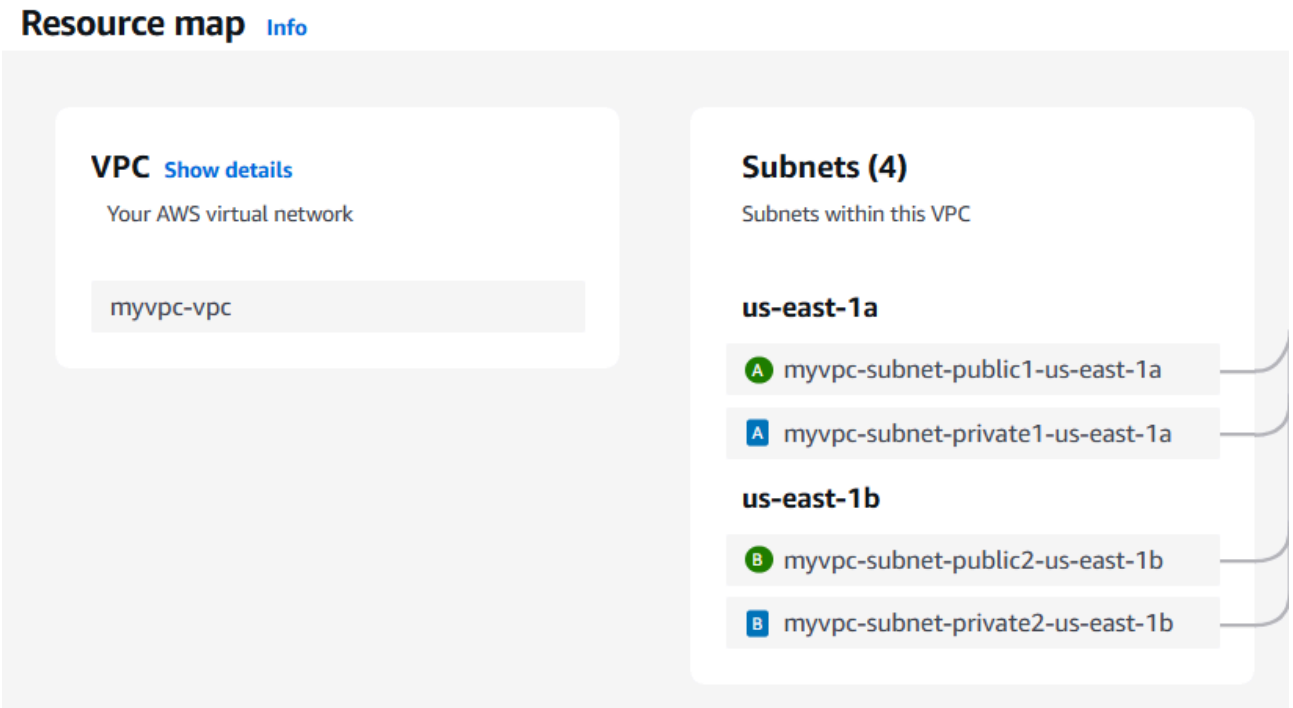
https://hackmd.io/@joaniznardo/asixcloud20242025-s03

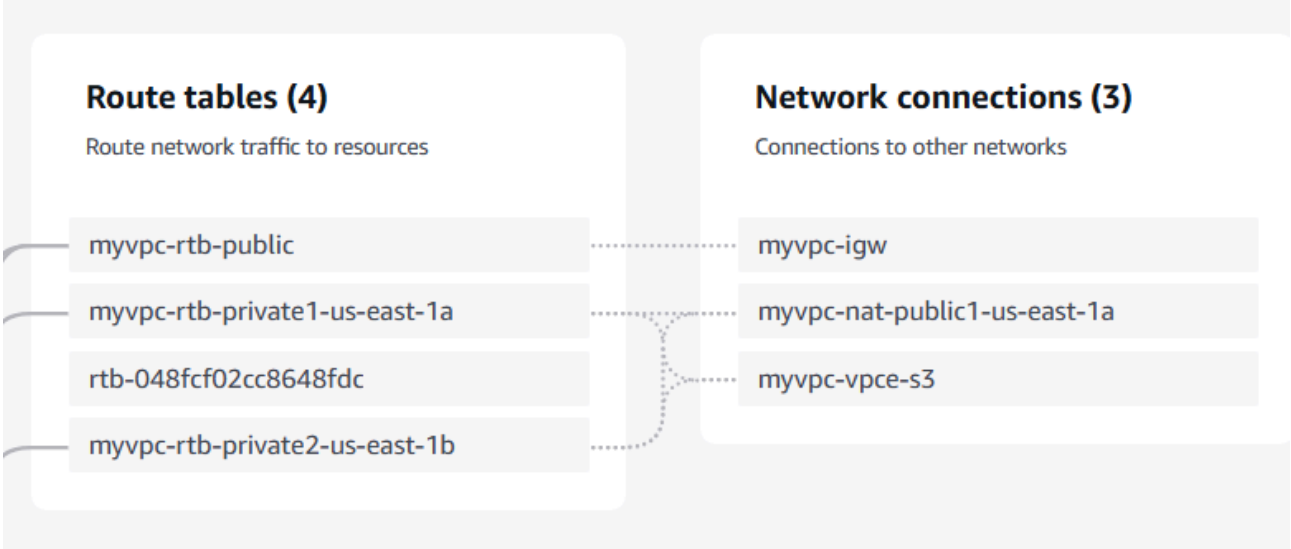
Preparar el escenario y despliegue infraestructura

- vpc
- 2 subnets pub / 2 subnets priv
- 1 igw
- 1 ngw
- security groups
    - ssh
    - http
    - mysql
- s3
- Amazon RDS

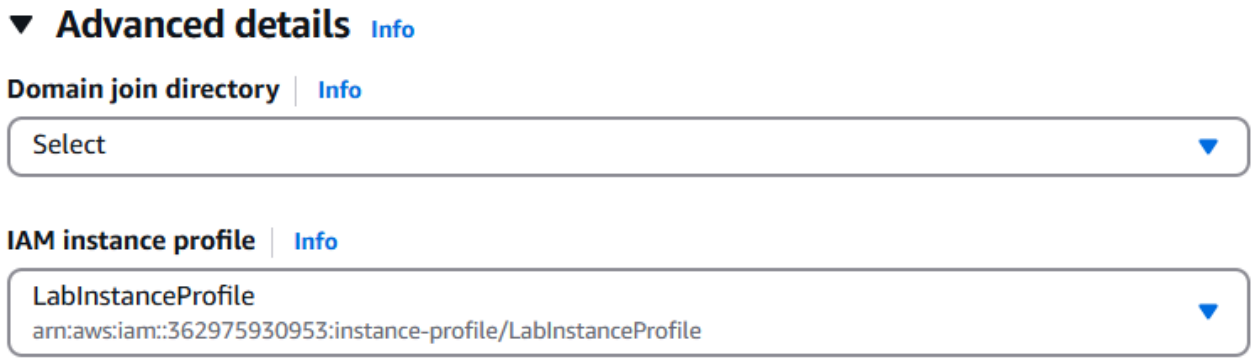
Procedimiento

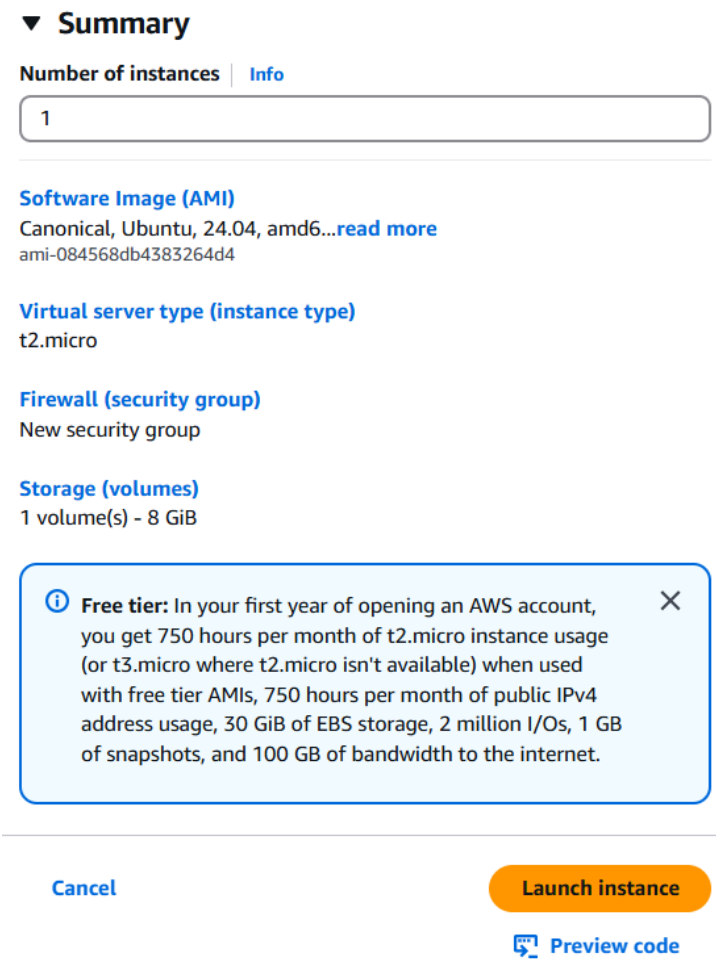
VPC




Route tables (4)
Route network traffic to resources
myvpc-rtb-public
myvpc-rtb-private1-us-east-1a
rtb-048fcf02cc8648fdc
myvpc-rtb-private2-us-east-1b
Network connections (3)
Connections to other networks
myvpc-igw
myvpc-nat-public1-us-east-1a
myvpc-vpce-s3

EC2 PRIVATE


Advanced details Info
Domain join directory Info
Select
IAM instance profile Info
LabInstanceProfile
arn:aws:iam::362975930953:instance-profile/LabInstanceProfile




```
#!/bin/bash
sudo apt update
sudo apt install -y mysql-server
tee crea-wordpress-db.sql <<EOF
CREATE DATABASE wordpressdb01;
CREATE USER 'asix01'@'%' IDENTIFIED BY 'Sup3rins3gura!';
GRANT ALL PRIVILEGES ON wordpressdb01.* TO 'asix01'@'%';
FLUSH PRIVILEGES;
exit
EOF

cat crea-wordpress-db.sql | sudo mysql -u root -pPassw0rdRootAcc0unt


sudo sed -i '/addre/s|127.0.0.1|0.0.0.0|' /etc/mysql/mysql.conf.d/mysqld.cnf
```

EC2 PUBLIC

User data - *optional* | Info

Upload a file with your user data or enter it in the field.

Choose file

```
</VirtualHost>
EOF
sudo a2ensite wordpress
sudo a2enmod rewrite
sudo a2dissite 000-default
sudo service apache2 reload
sudo -u www-data cp /srv/www/wordpress/wp-config-sample.php /srv/www/wordpress/wp-config.php
sudo -u www-data sed -i 's/database_name_here/wordpressdb01/' /srv/www/wordpress/wp-config.php
sudo -u www-data sed -i 's/username_here/asix01/' /srv/www/wordpress/wp-config.php
sudo -u www-data sed -i 's/password_here/Sup3rins3gura!/' /srv/www/wordpress/wp-config.php
sudo -u www-data sed -i 's/localhost/10\.0\.159\.214/' /srv/www/wordpress/wp-config.php
```

S3

Al crear un bucket el nombre es sensitive. Así que mejor que siga esta nomenclatura:s3-wp-backup-xxxx

Account snapshot - *updated every 24 hours* All AWS Regions

Storage lens provides visibility into storage usage and activity trends. Metrics don't include directory buckets. Learn mor

General purpose buckets | Directory buckets

General purpose buckets (1) Info All AWS Regions

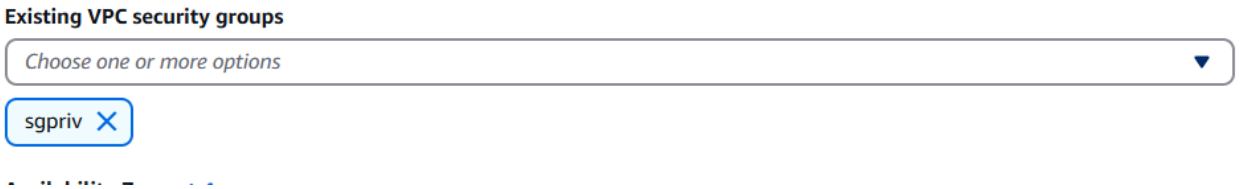
Buckets are containers for data stored in S3.

Find buckets by name

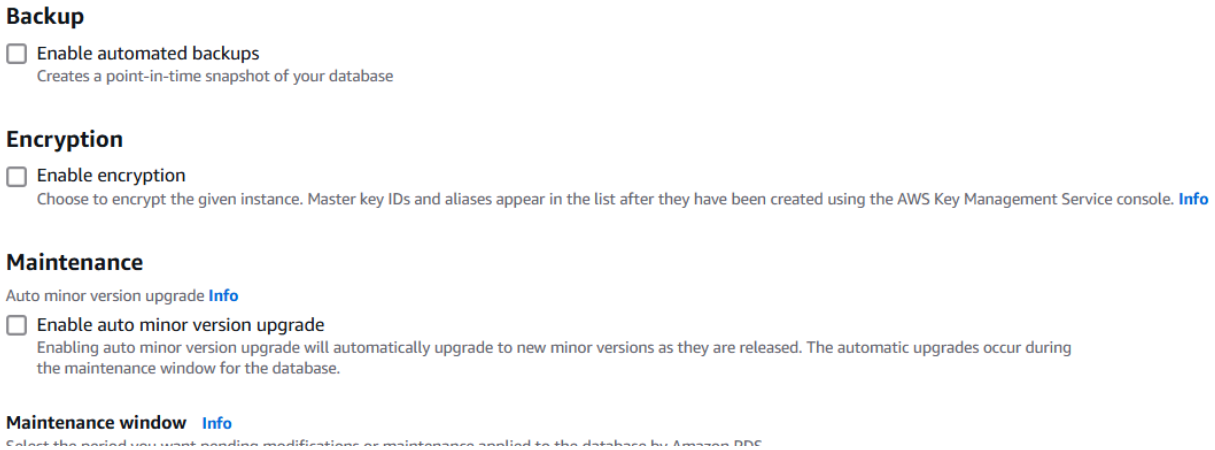
Name	AWS Region
s3-wp-backup-carla	US East (N. Virginia) us-east-1

Amazon RDS

Create database

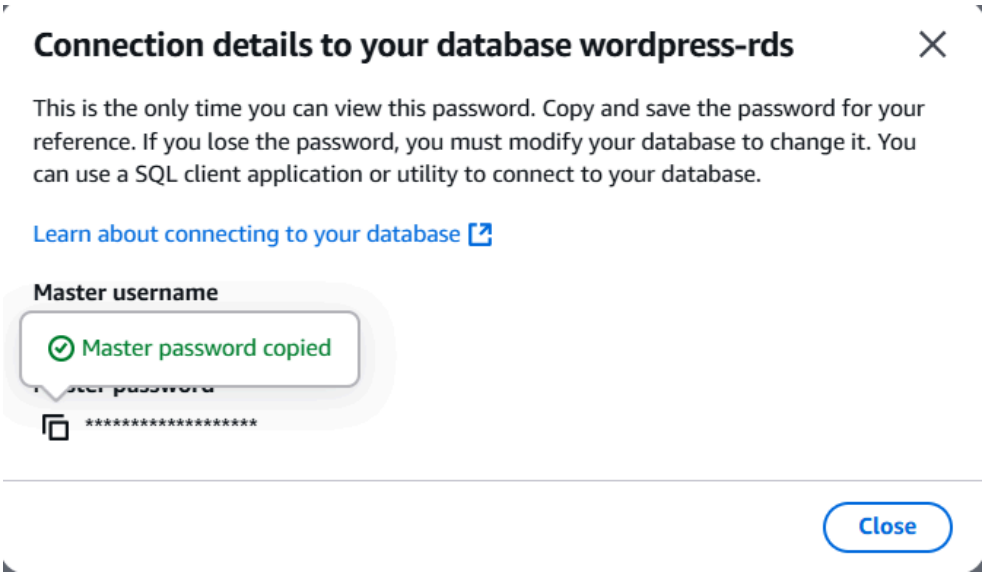


Utilizaremos el sgpriv de la ec2 que permite el tráfico del puerto 3306 desde la ec2 pub

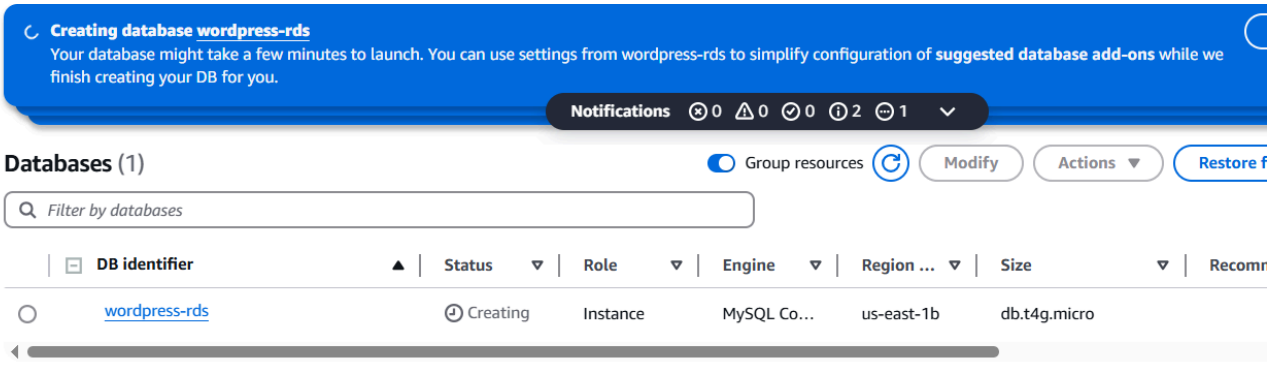


No seleccionamos por los costes.

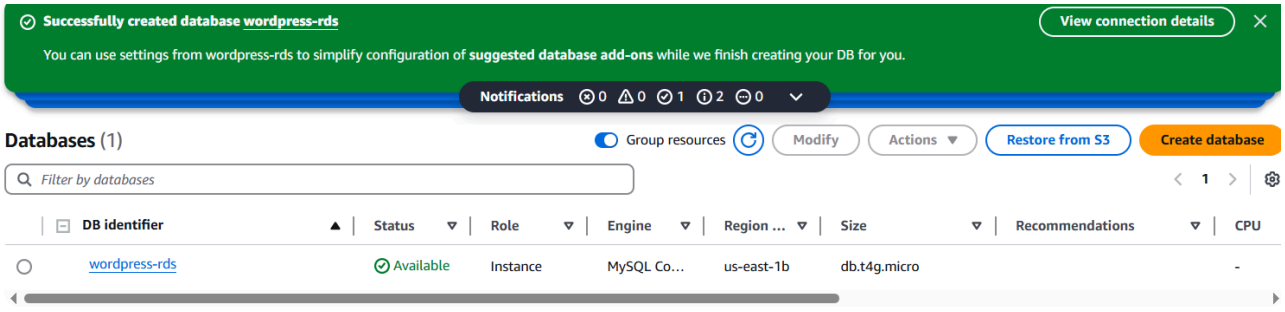
Create database



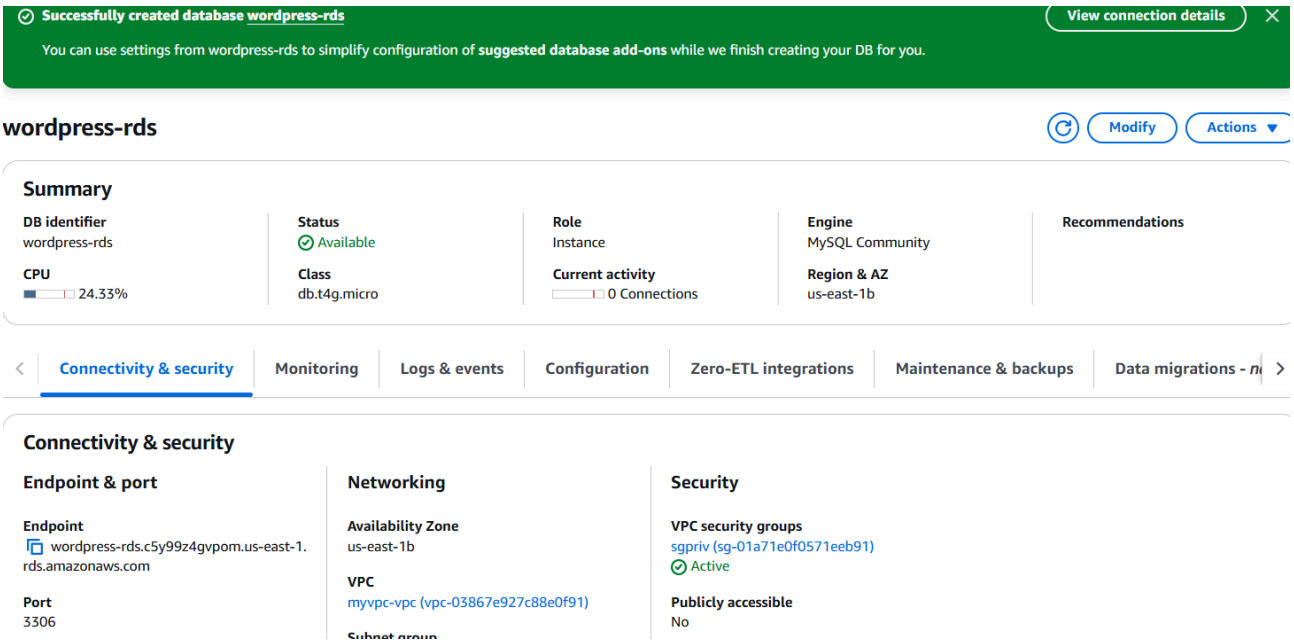
Passw0rdRootAcc0unt



Esperamos.



Listo



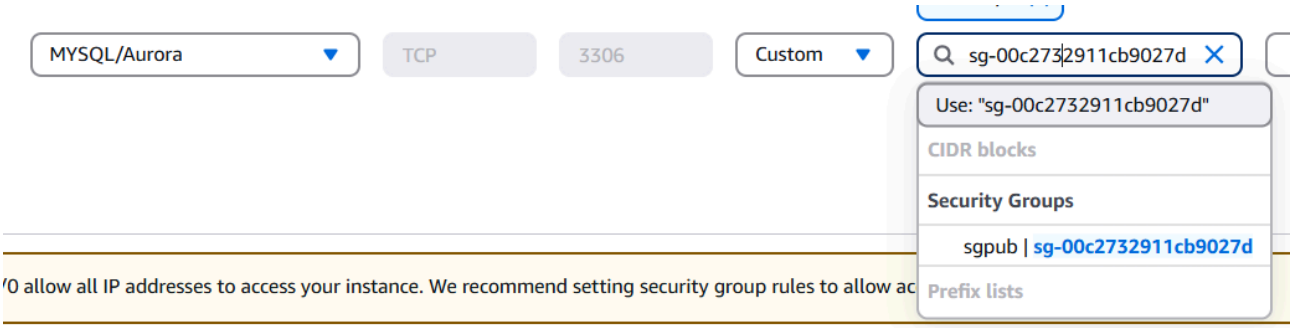
Copiamos el endpoint:

wordpress-rds.c5y99z4gvpom.us-east-1.rds.amazonaws.com la utilizaremos para hacer

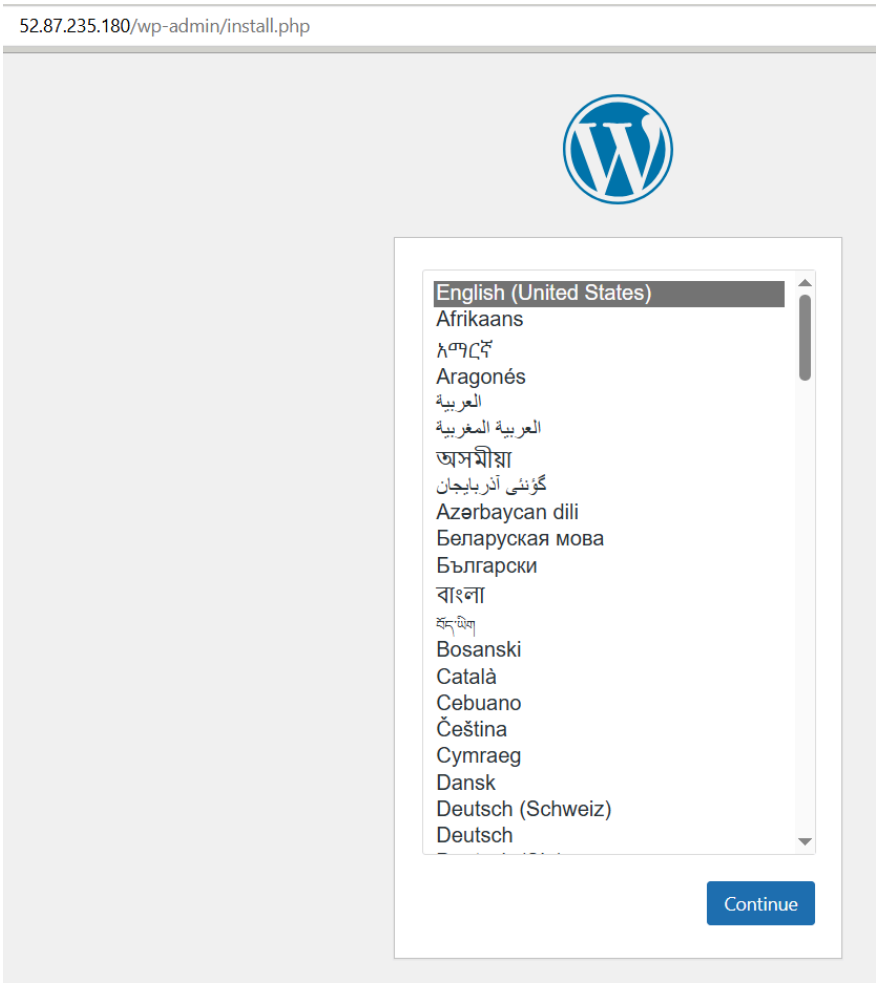
la migración y conectarnos.

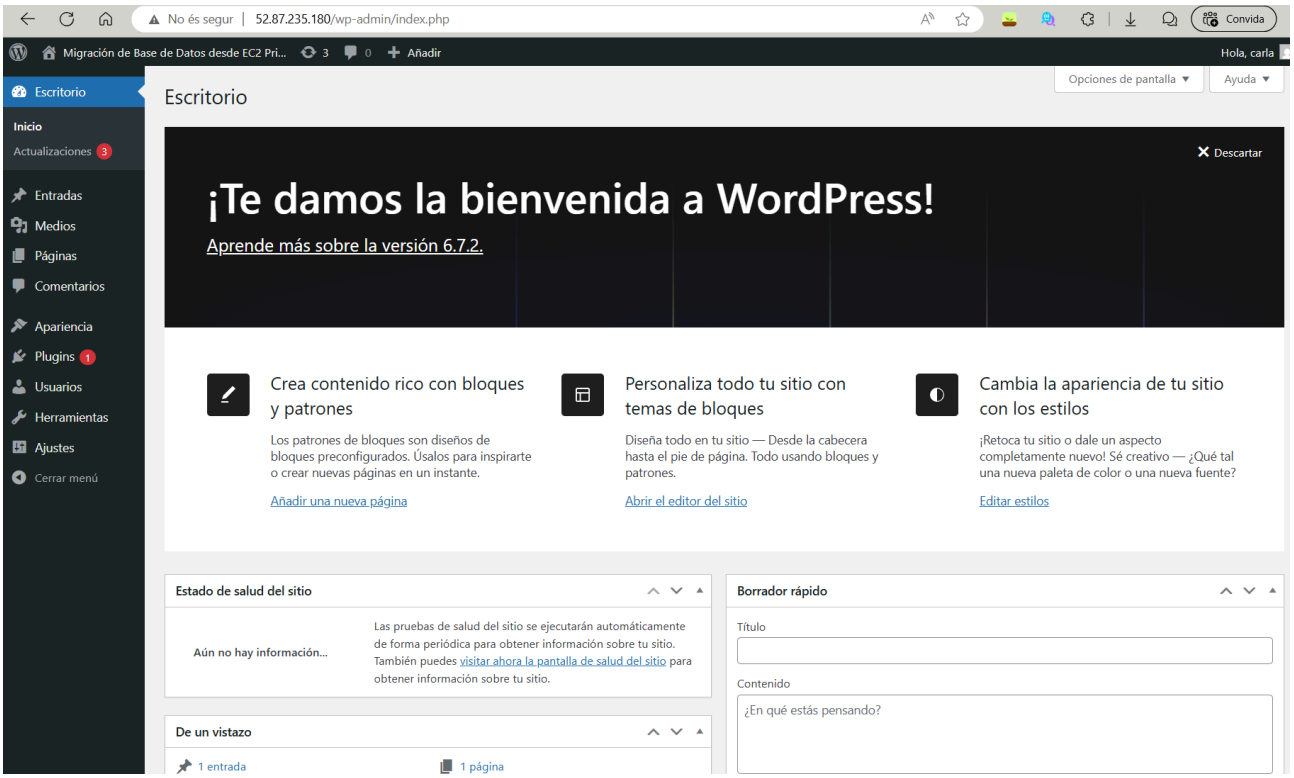
Security groups

Ahora tenemos que modificar el sg de la privada para que la ec2pub pueda acceder a la base de datos.



Verificamos conectándonos al IP



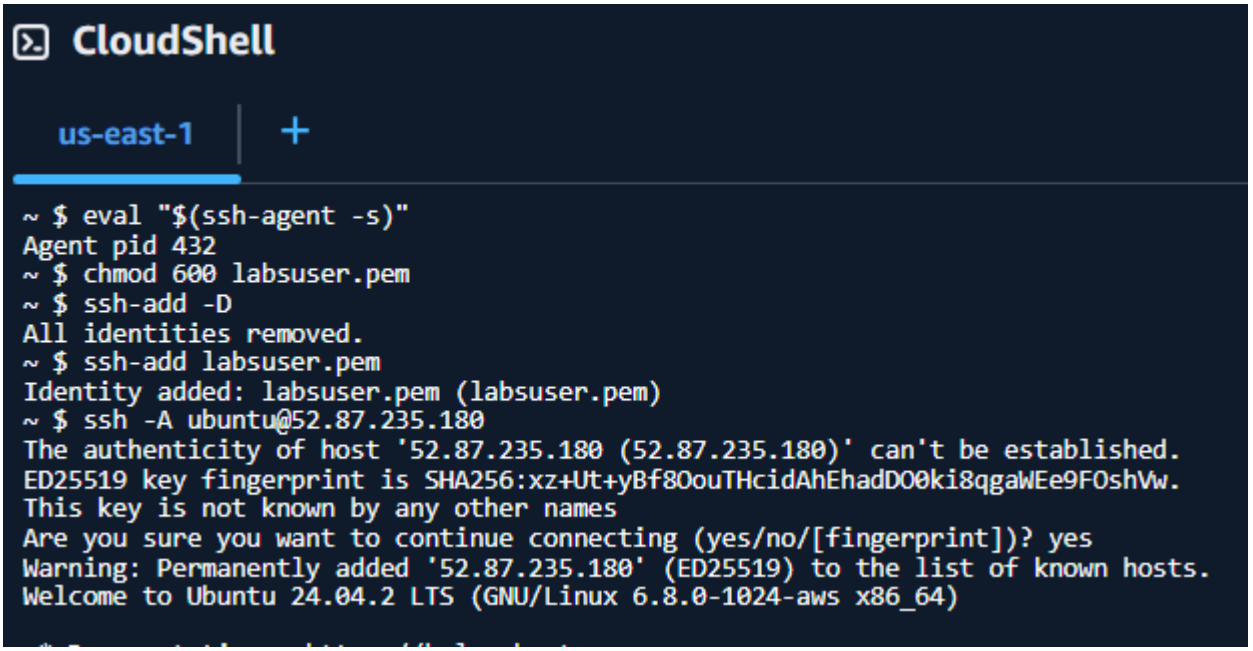


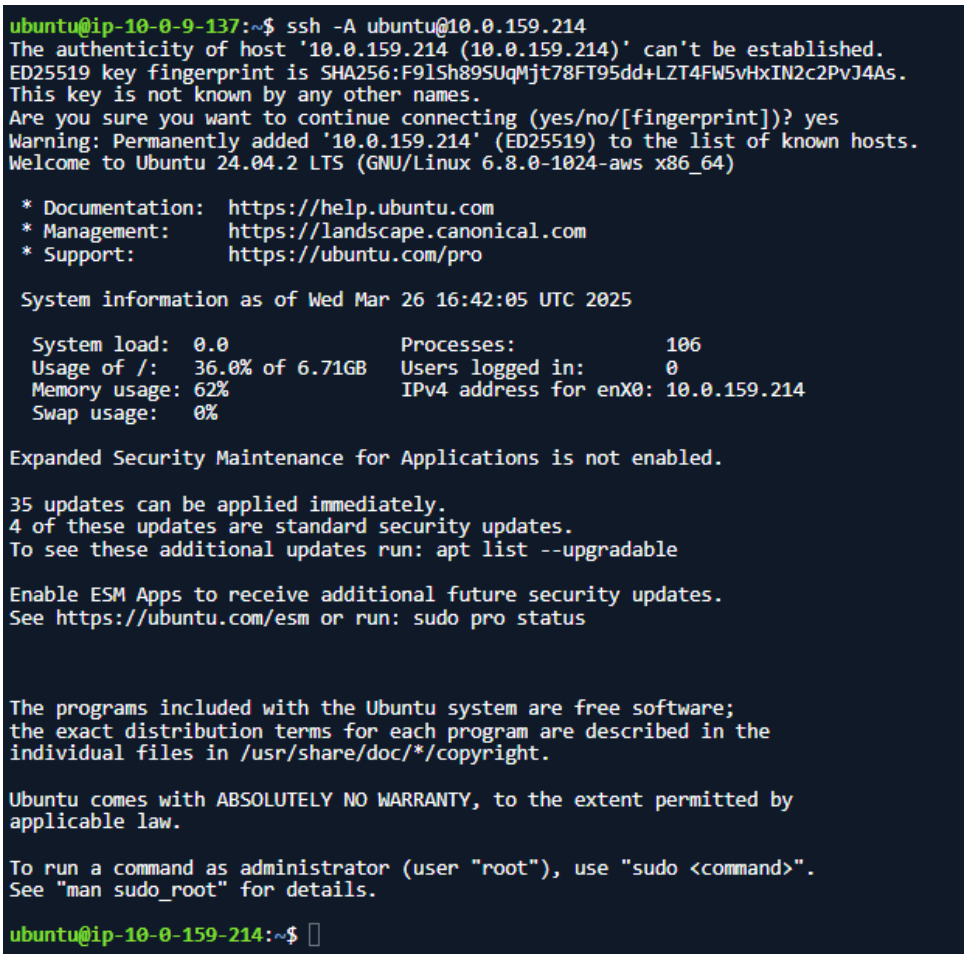
Copia de seguridad base de datos desde la instancia privada:

Ahora que tenemos todo instalado procedemos a realizar la copia de seguridad de la base de datos para luego migrar a Amazon RDS.

Para hacer la copia y luego esta copia moverla a la ec2 pub, tenemos que arrastra la clave privada es decir la labsuser.pem con la opción -A, para así mover archivos desde la ec3 privada a la ec2 pública.

Pasos previos:


CloudShell
us-east-1
+
~ $ eval "$(ssh-agent -s)"
Agent pid 432
~ $ chmod 600 labsuser.pem
~ $ ssh-add -D
All identities removed.
~ $ ssh-add labsuser.pem
Identity added: labsuser.pem (labsuser.pem)
~ $ ssh -A ubuntu@52.87.235.180
The authenticity of host '52.87.235.180 (52.87.235.180)' can't be established.
ED25519 key fingerprint is SHA256:xz+Ut+yBf8OouTHcidAhEhadDO0ki8qgaWEe9FOshVw.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '52.87.235.180' (ED25519) to the list of known hosts.
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-1024-aws x86_64)


ubuntu@ip-10-0-9-137:~$ ssh -A ubuntu@10.0.159.214
The authenticity of host '10.0.159.214 (10.0.159.214)' can't be established.
ED25519 key fingerprint is SHA256:F9lSh89SUqMjt78FT95dd+LZT4FW5vHxIN2c2PvJ4As.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.159.214' (ED25519) to the list of known hosts.
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-1024-aws x86_64)
* Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/pro
System information as of Wed Mar 26 16:42:05 UTC 2025
System load: 0.0 Processes: 106
Usage of /: 36.0% of 6.71GB Users logged in: 0
Memory usage: 62% IPv4 address for enX0: 10.0.159.214
Swap usage: 0%
Expanded Security Maintenance for Applications is not enabled.
35 updates can be applied immediately.
4 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable
Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status
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-10-0-159-214:~$

```
~ $ ssh -A ubuntu@52.87.235.180
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-1024-aws x86_64)

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

 System information as of Wed Mar 26 17:27:59 UTC 2025

  System load:  0.0               Processes:             121
  Usage of /:   32.4% of 6.71GB   Users logged in:       1
  Memory usage: 32%               IPv4 address for enX0: 10.0.9.137
  Swap usage:   0%


Expanded Security Maintenance for Applications is not enabled.

35 updates can be applied immediately.
4 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


Last login: Wed Mar 26 16:58:58 2025 from 18.206.107.29
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-10-0-9-137:~$ ssh -A ubuntu@10.0.159.214
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-1024-aws x86_64)

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

 System information as of Wed Mar 26 17:28:14 UTC 2025

  System load:  0.0               Processes:             111
  Usage of /:   36.0% of 6.71GB   Users logged in:       1
  Memory usage: 62%               IPv4 address for enX0: 10.0.159.214
  Swap usage:   0%


Expanded Security Maintenance for Applications is not enabled.

35 updates can be applied immediately.
4 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


Last login: Wed Mar 26 16:42:06 2025 from 10.0.9.137
ubuntu@ip-10-0-159-214:~$
```

Verificamos que podemos acceder a la base de datos:

```
ubuntu@ip-10-0-159-214:~$  sudo mysql -u root -pPassw0rdRootAcc0unt -D wordpressdb01
mysql: [Warning] Using a password on the command line interface can be insecure.
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 51
Server version: 8.0.41-0ubuntu0.24.04.1 (Ubuntu)

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

Ahora con los siguientes comandos realizamos la copia:

Generar dump de la BBDD

mysqldump -u root -p wordpressdb01 > backup_db.sql

Password: Passw0rdRootAcc0unt

```
ubuntu@ip-10-0-159-214:~$ sudo mysqldump -u root -pPassw0rdRootAcc0unt wordpressdb01 > ~/backup.sql
mysqldump: [Warning] Using a password on the command line interface can be insecure.
ubuntu@ip-10-0-159-214:~$ ls
backup.sql  labsuser.pem  ubuntu@52.87.235.180
ubuntu@ip-10-0-159-214:~$ sudo nano backup.sql
ubuntu@ip-10-0-159-214:~$
```

Hice reset ya que lo deje a media y volvi hacerlo lo mismo por eso cambian las IP y esta vez no agregue el rol LabInstanceProfile.

```
ubuntu@ip-10-0-129-37:~$ sudo mysqldump -u root -p wordpressdb01 > backup_db.sql
Enter password:
ubuntu@ip-10-0-129-37:~$ ls
backup_db.sql
ubuntu@ip-10-0-129-37:~$
```

Luego de generar la copia tenemos que pasarla a la ec2 pub para que esta la guarda en la s3 y de ahí podemos hacer la migración a RDS.

scp ~/backup_db.sql ubuntu@34.227.158.198:~/

```
ubuntu@ip-10-0-129-37:~$ scp ~/backup_db.sql ubuntu@34.227.158.198:~/
The authenticity of host '34.227.158.198 (34.227.158.198)' can't be established.
ED25519 key fingerprint is SHA256:NFngIueKAVz+3AUZ4bvajUS5LXLfeRQ+RQeUDt+2tDI.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '34.227.158.198' (ED25519) to the list of known hosts.
backup_db.sql
ubuntu@ip-10-0-129-37:~$ ls
backup_db.sql
ubuntu@ip-10-0-129-37:~$ exit
logout
Connection to 10.0.129.37 closed.
ubuntu@ip-10-0-1-245:~$ ls
backup_db.sql
ubuntu@ip-10-0-1-245:~$
```

exit Y comprobamos que esté en la ec2 pub

Desde la instancia pública:

Bueno ya que tenemos la copia en la ec2 pub podemos proceder a guardarla en la s3 que hemos creado previamente, ya que para ello tenemos que tener aws cli y sus credenciales para invocar las api y se puede enviar.

sudo snap install aws-cli --classic

```
ubuntu@ip-10-0-1-245:~$ sudo snap install aws-cli --classic
aws-cli (v2/stable) 2.25.4 from Amazon Web Services (aws✓) installed
ubuntu@ip-10-0-1-245:~$
```

sudo nano ~/.aws/credentials

```
ubuntu@ip-10-0-1-245:~$ aws configure
AWS Access Key ID [None]: ASIA2OIZ3EGTLAUU4K2R
AWS Secret Access Key [None]: Oa/CPSy1QbeIMgctQLduRTez+aLFNNNh97ZyZ+Eh
Default region name [None]: us-east-1
Default output format [None]:
ubuntu@ip-10-0-1-245:~$ aws sts get-caller-identity

An error occurred (InvalidClientTokenId) when calling the GetCallerIdentity operation: The security token included in the request is invalid.
ubuntu@ip-10-0-1-245:~$ cat ~/.aws/credentials
[default]
aws_access_key_id = ASIA2OIZ3EGTLAUU4K2R
aws_secret_access_key = Oa/CPSy1QbeIMgctQLduRTez+aLFNNNh97ZyZ+Eh
ubuntu@ip-10-0-1-245:~$ sudo nano ~/.aws/credentials
ubuntu@ip-10-0-1-245:~$ aws sts get-caller-identity
{
    "UserId": "AROA2OIZ3EGTIMUJETRIC:user3528210=cala",
    "Account": "717850616230",
    "Arn": "arn:aws:sts::717850616230:assumed-role/voclabs/user3528210=cala"
}
ubuntu@ip-10-0-1-245:~$
```

```
}
ubuntu@ip-10-0-1-245:~$ aws s3 ls
2025-03-27 23:29:21 s3-wp-backup-carla
ubuntu@ip-10-0-1-245:~$
```

Guardar la copia en la s3:

aws s3 cp backup_db.sql s3://s3-wp-backup-carla/

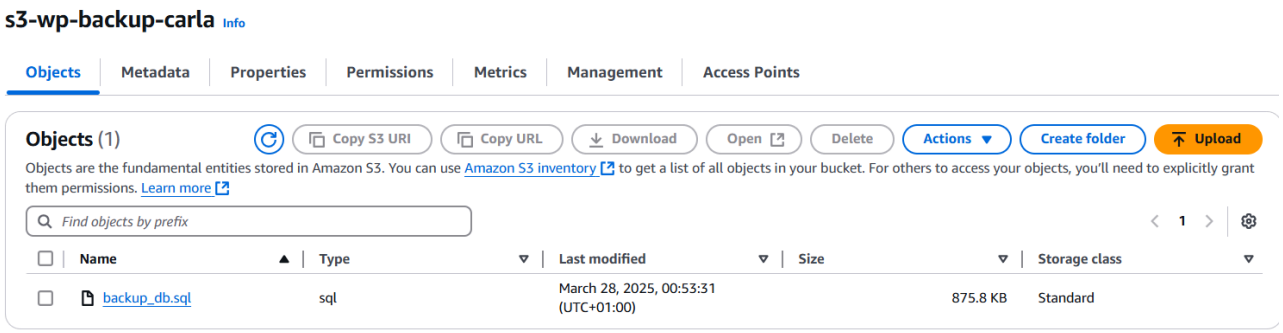
```
ubuntu@ip-10-0-1-245:~$ aws s3 cp ./backup_db.sql s3://s3-wp-backup-carla/
upload: ./backup_db.sql to s3://s3-wp-backup-carla/backup_db.sql
ubuntu@ip-10-0-1-245:~$
```

Verificamos que el archivo se subió correctamente:

aws s3 ls s3://s3-wp-backup-carla/

```
ubuntu@ip-10-0-1-245:~$ aws s3 ls s3://s3-wp-backup-carla/
2025-03-27 23:53:31     896795 backup_db.sql
ubuntu@ip-10-0-1-245:~$
```

O vamos directamente a verlo a la consola:



Migración a Amazon RDS

ENDPOINT:

wordpress-rds.c5y99z4gvpom.us-east-1.rds.amazonaws.com

Comprobamos que nos podemos entrar a la base de datos RDS antes de realizar la migración.

```
ubuntu@ip-10-0-1-245:~$ sudo mysql -u admin -p -h wordpress-rds.c5y99z4gvpom.us-east-1.rds.amazonaws.com
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 27
Server version: 8.0.40 Source distribution

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

Podemos entrar. Ahora si vamos hacerlo con el siguiente comando:

mysql -u admin -p -h wordpress-rds.c5y99z4gvpom.us-east-1.rds.amazonaws.com wordpressdb01 < backup_db.sql

```
ubuntu@ip-10-0-1-245:~$ sudo mysql -u admin -p -h wordpress-rds.c5y99z4gvpom.us-east-1.rds.amazonaws.com  wordpressdb01 < backup_db.sql
Enter password:
ubuntu@ip-10-0-1-245:~$
```

Comprobamos:

```
ubuntu@ip-10-0-1-245:~$ sudo mysql -u admin -pPassw0rdRootAcc0unt -h wordpress-rds.c5y99z4gvpom.us-east-1.rds.amazonaws.com
mysql: [Warning] Using a password on the command line interface can be insecure.
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 30
Server version: 8.0.40 Source distribution

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
| wordpressdb01      |
+--------------------+
5 rows in set (0.00 sec)

mysql> use wordpressdb01
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables
    -> ;c
+-------------------------+
| Tables_in_wordpressdb01 |
+-------------------------+
| wp_commentmeta          |
| wp_comments             |
| wp_links                |
| wp_options              |
| wp_postmeta             |
| wp_posts                |
| wp_term_relationships   |
| wp_term_taxonomy        |
| wp_termmeta             |
```

Listo

Configurar WordPress para usar RDS

Ahora vamos a configurar wordpress para usar desde RDS y no desde la local.

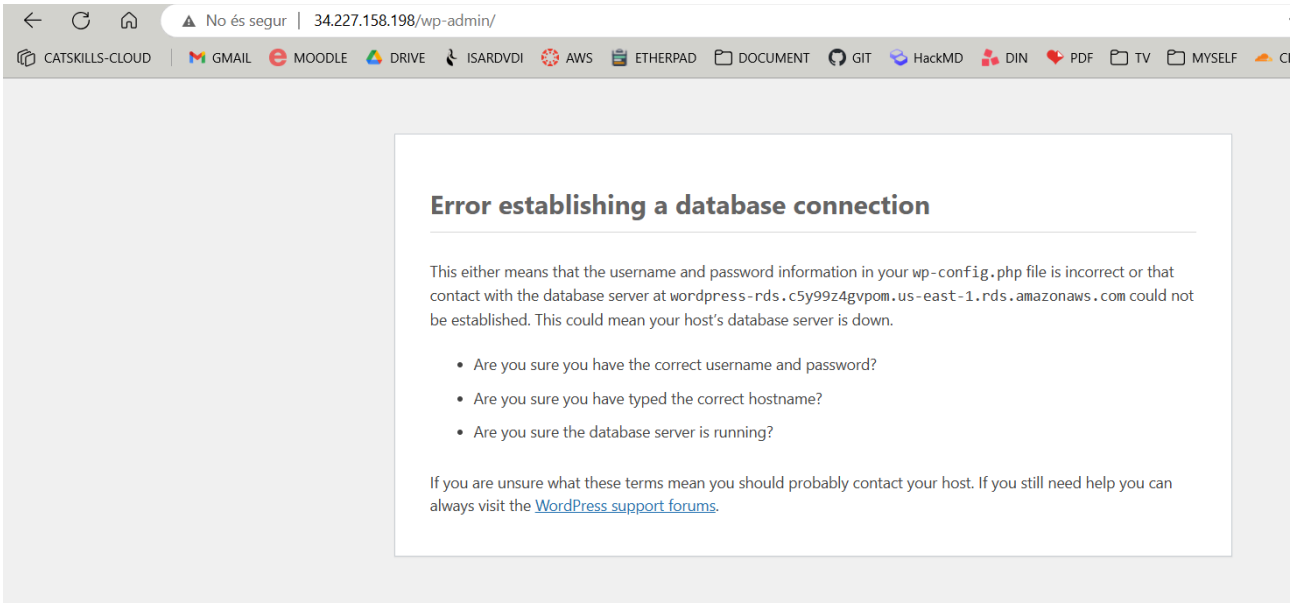
```
sudo -u www-data sed -i 's/localhost/10\.54\.153\.64/' /srv/www/wordpress/wp-config.php
```

Tiene que apuntar desde el endpoint de la RDS y no desde la ec2priv.

sudo nano /srv/www/wordpress/wp-config.php

```
/** Database hostname */
define( 'DB_HOST', '10.0.129.37' );
```

```
/** Database hostname */
define( 'DB_HOST', 'wordpress-rds.c5y99z4gvpom.us-east-1.rds.amazonaws.com' );
```



No podemos porque también hay que modificar el user y la contraseña de la base de datos RDS:

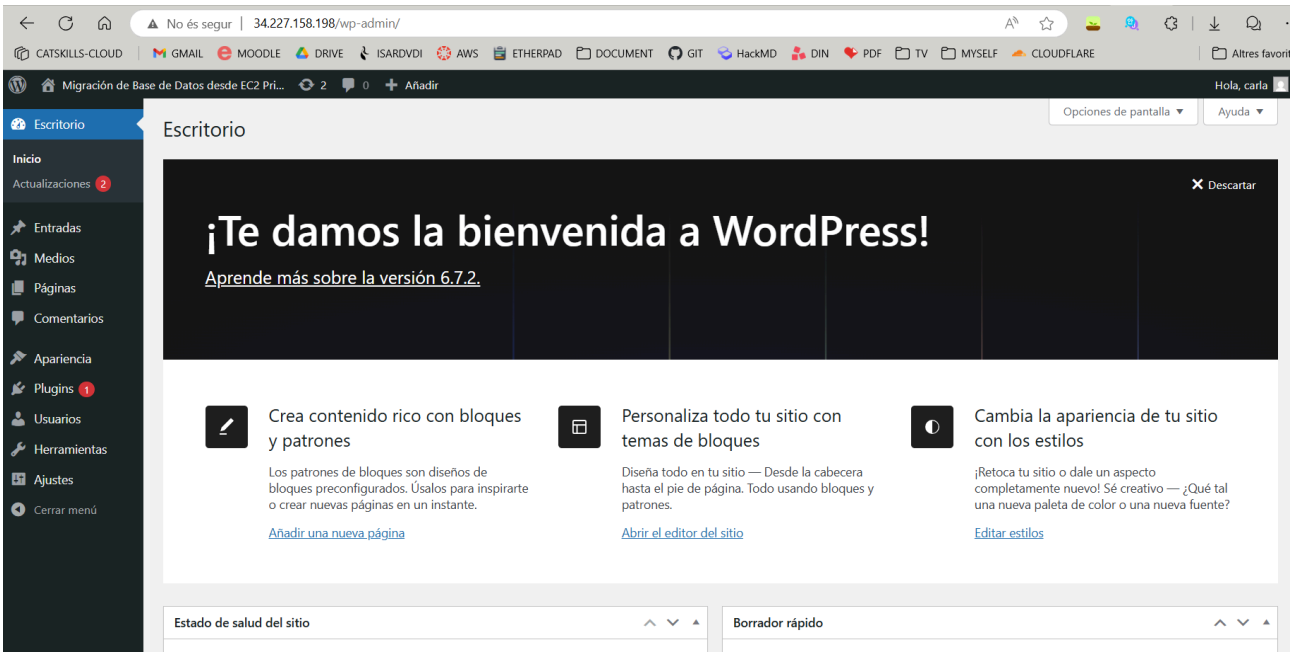
```
/** Database username */
define( 'DB_USER', 'asix01' );
```

```
/** Database username */
define( 'DB_USER', 'admin' );
```

define('DB_PASSWORD', 'Passw0rdRootAcc0unt');

```
/** Database password */
define( 'DB_PASSWORD', 'Sup3rins3gura!' );
```

```
/** Database password */
define( 'DB_PASSWORD', 'Passw0rdRootAcc0unT' );
```

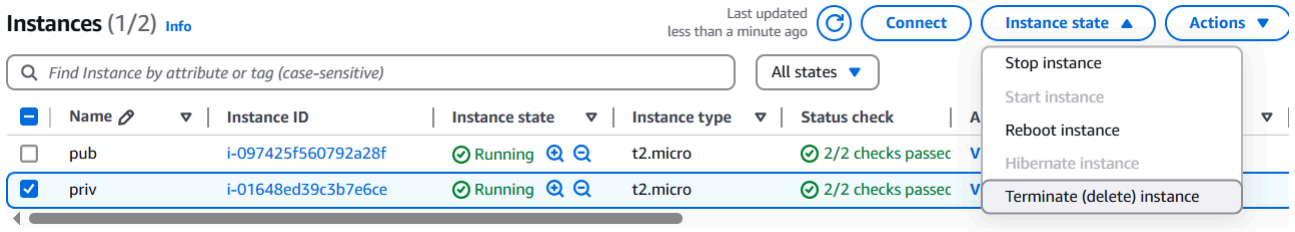


No hace falta restaurar el servicio de mysql, con solo guardar el archivo ya funciona y refrescamos la página y ya nos dejará.

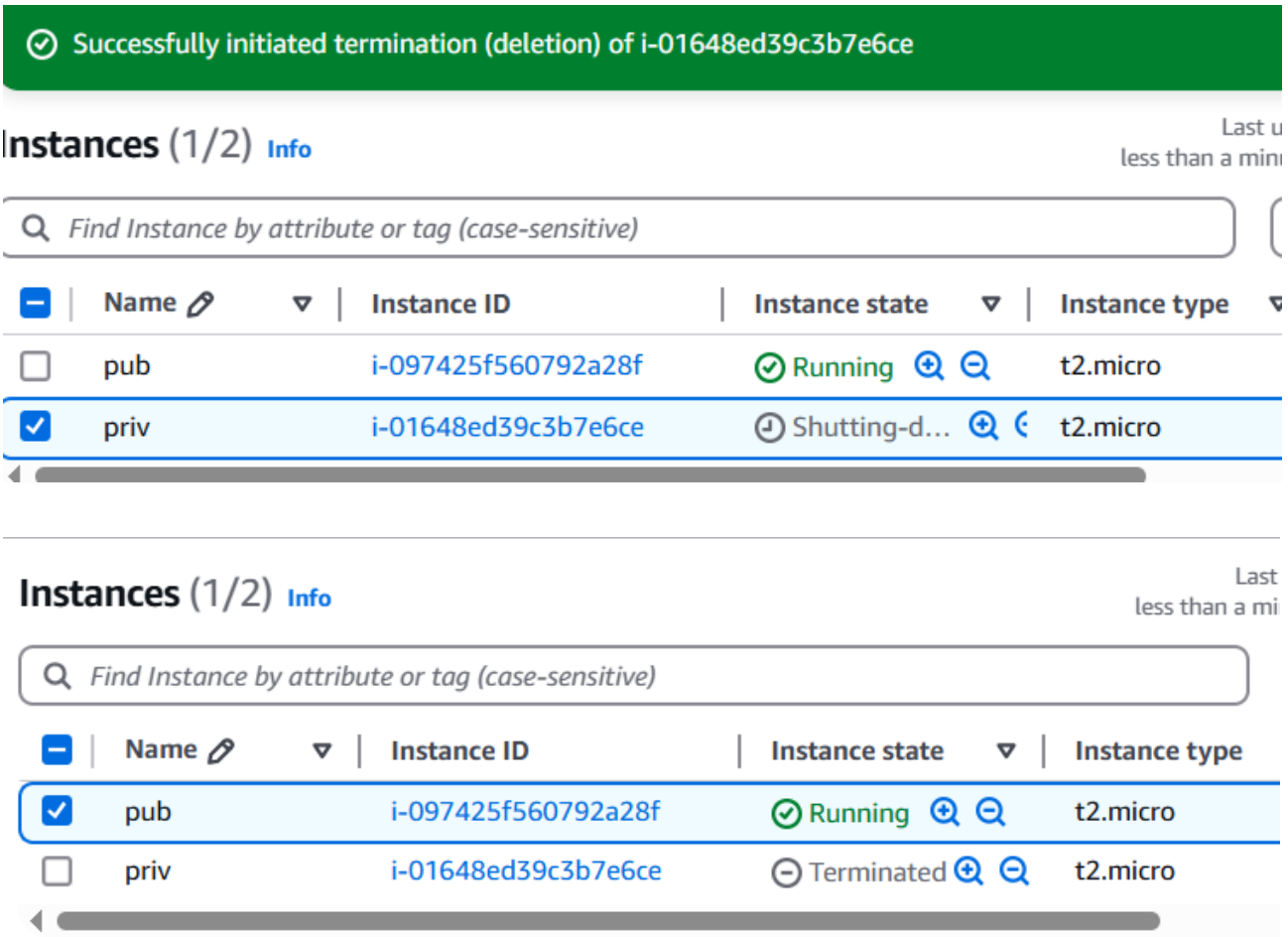
Bueno ahora vamos a eliminar la copia y detener la ec2priv.

```
ubuntu@ip-10-0-1-245:~$ ls
backup_db.sql  snap
ubuntu@ip-10-0-1-245:~$ rm ~/backup_db.sql
ubuntu@ip-10-0-1-245:~$ ls
snap
ubuntu@ip-10-0-1-245:~$
```

Igual la tenemos en la s3 también.



Terminate.



Listo hemos terminado:)