

RDS: Bases de Datos

Pinceladas en la configuraciones de los servicios RDS de AWS.

El proceso de creación, como los demás productos de AWS, intenta una coherencia en la forma de desplegar-los.

Lo primero que debemos preguntarnos es:

¿Qué es RDS? Es un “Servicio de bases de datos relacionales” (RDS)

Amazon Relational Database Service (Amazon RDS)

Es un servicio un buen ejemplo de Paas. Aquí tenemos el motor de la base de datos, que con las opciones iremos configurando y en uno de los últimos pasos se dará nombre a la base de datos (vacía al principio) que más adelante usaremos.

1º le damos a crear ...

≡ [Aurora and RDS](#) > [Crear base de datos](#)

Crear base de datos Información

Elegir un método de creación de base de datos

Creación estándar

Puede definir todas las opciones de configuración, incluidas las de disponibilidad, seguridad, copias de seguridad y mantenimiento.

Creación sencilla

Utilice las configuraciones predeterminadas para su base de datos.

Elegiremos el motor, en este ejemplo “MySQL”

Opciones del motor

Tipo de motor Información

Aurora (MySQL Compatible)



MySQL



Hay que elegir la versión que por defecto no es la último modelo, este paso es importante, ya que si la versión no es compatible con el proyecto, hay que crear una RDS nueva. Y la única forma es una vez creada, probarla.

Versión del motor

MySQL 8.0.41

Aquí optamos por la capa gratuita ...

Plantillas

Elija una plantilla de ejemplo para adaptarla a su caso de uso.

Capa gratuita

Utilice el nivel gratuito de RDS para desarrollar nuevas aplicaciones, probar aplicaciones existentes o adquirir experiencia práctica con Amazon RDS. [Información](#)

Hay que poner un nombre a nuestro RDS (no a la base de datos) ...

Configuración

Identificador de instancias de bases de datos

Escriba un nombre para la instancia de base de datos. El

database-miproyecto

Y definir como loguearnos, la forma “Autoadministrado” es optar por usuario y contraseña, esta debe ser al menos fuerte.

▼ Configuración de credenciales

Nombre de usuario maestro [Información](#)

Escriba un ID de inicio de sesión para el usuario maestro de la instancia de base de datos.

administradora

1 a 16 caracteres alfanuméricos. El primer carácter debe ser una letra.

Administración de credenciales

Puede usar AWS Secrets Manager o administrar sus credenciales de usuario maestro.

Administrado en AWS Secrets Manager - *más seguro*

RDS genera una contraseña y la administra durante todo su ciclo de vida mediante AWS Secrets Manager.

Autoadministrado

Cree su propia contraseña

Conexión: aquí optamos que inicialmente “No se conecte” a la EC2, ya que eso lo controlaremos desde el grupo de seguridad de la RDS (abriendo o cerrando las conexiones)

Conectividad [Información](#)

Recurso de computación

Seleccione si desea configurar una conexión a un recurso de computación para esta base de datos. Al establecer una conexión se pueda conectar a esta base de datos.

No se conecte a un recurso informático EC2

No configure una conexión a un recurso informático para esta base de datos. Puede configurar manualmente una conexión a un recurso informático más adelante.

Tipo de red [Información](#)

Para utilizar el modo de pila doble, asegúrese de asociar un bloque de CIDR IPv6 a una subred en la VPC que

IPv4

Sus recursos solo pueden comunicarse a través del protocolo de direcciones IPv4.

Acceso público [Información](#)

Sí

RDS asigna una dirección IP pública a la base de datos. Puede permitir que los clientes conectarse a la base de datos. Elija uno o varios grupos.

Por lo tanto, debemos permitir al menos una IP pública.

No

Autenticación de bases de datos, optaremos:

“Opciones de autenticación de bases de datos” → “Autenticación **con contraseña**”

En este punto, para nuestros proyectos, voy a preferir una **VPC** [nueva](#), que la adaptaremos a nuestras necesidades.

Nube privada virtual (VPC) [Información](#)

Elija la VPC. La VPC define el entorno

[Crear nueva VPC](#)

Y por supuesto, un grupo de seguridad específico para nuestra nueva RDS, que como he dicho antes, se usará para especificar quién podrá acceder a ella ...

Grupo de seguridad de VPC (firewall) [Información](#)

Elija uno o varios grupos de seguridad de VPC para permitir el acceso a su base de datos. Asegúrese de que las reglas del grupo de seguridad permitan el tráfico de MySQL.

Elegir existente

Elegir grupos de seguridad de VPC existentes

Crear nuevo

Crear un grupo de seguridad

Nuevo nombre del grupo de seguridad de VPC

seguridad-miprojectonube

*Aconsejable, elegir una zona, ya que ese dato lo usaremos más adelante.

Zona de disponibilidad

Por último, tenemos la configuraciones adicionales, que dejaremos el puerto por defecto de MySQL y le pondremos nombre a “la base de datos”

▼ Configuración adicional

▼ Configuración adicional

Opciones de base de datos, cifrado activado,

Puerto de la base de datos [Información](#)

Puerto TCP/IP que la base de datos escucha

3306

Opciones de base de datos

Nombre de base de datos inicial [Información](#)

mibaseparawordpressmiprojectonube

Como es sólo para el trabajo, deshabilitaremos todas las copias ya que son un coste adicional.

Copia de seguridad

Habilitar las copias de seguridad automatizadas.

Mantenimiento

Actualización automática de la versión secundaria [Información](#)

Habilitar actualización automática de versiones secundarias

Y ya sólo queda lanzarla, un proceso lento.

¡OJO!: Las **RDS** son servicios caros, y cuando no la usemos **debemos detenerlas temporalmente**.

Una vez lanzada, y vemos la recién creada y otra **detenida temporalmente**, que es la única forma que tiene AWS de detenerla.

Aurora and RDS < Bases de datos (3)

Filtrar por bases de datos

Identificador de base de datos | Estado

- database-baserds | Creando
- database-miprojectonube | Detención temporal

Panel

Bases de datos

Editor de consultas

Información sobre rendimiento

Instantáneas de

Exportaciones en Amazon S3

Después de un rato ...

ya se puede usar

Bases de datos (3)

Filtrar por bases de datos

Identificador de base de datos

database-baserds

Estado

Disponible

Haciendo clic en ella, vemos sus características y su ip pública, para poder conectarnos.

database-baserds

Resumen

Identificador de base de datos: database-baserds

Estado: Disponible

CPU: 4.31%

Clase: db.t4g.micro

Rol: Instancia

Actividad actual: 0 Conexiones

Conectividad y seguridad | Supervisión | Registros y eventos | Configuración | Integraciones sin extracci

Conectividad y seguridad

Punto de enlace y puerto

Punto de enlace: database-baserds.c98esgsyg9z1.us-east-1.rds.amazonaws.com

Redes

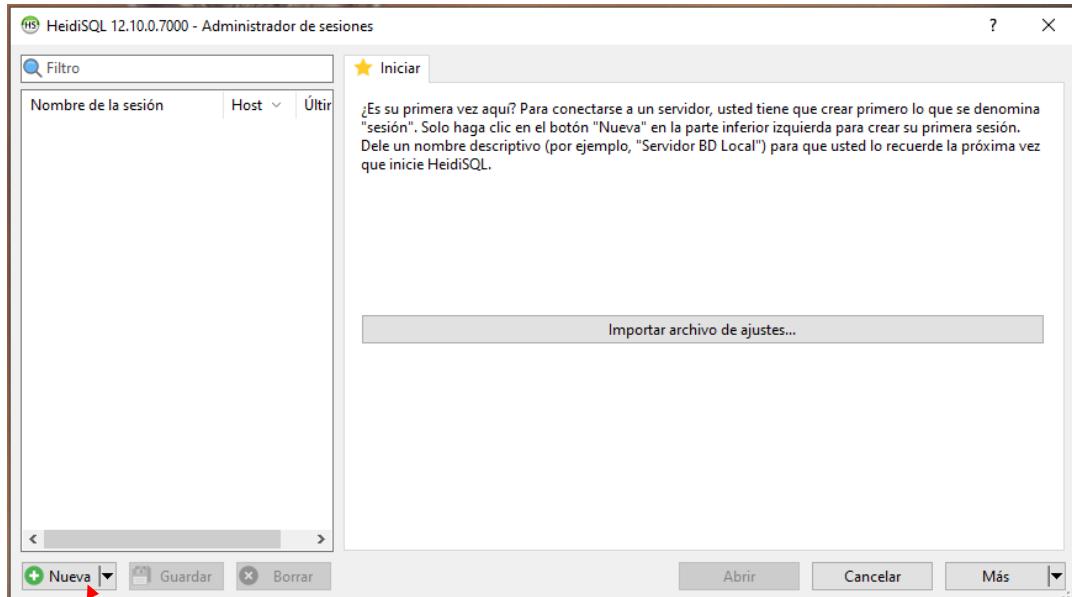
Zona de disponibilidad: us-east-1a

VPC

Seguridad

Grupos de seguridad: segubaserds (sg-054c) Activo

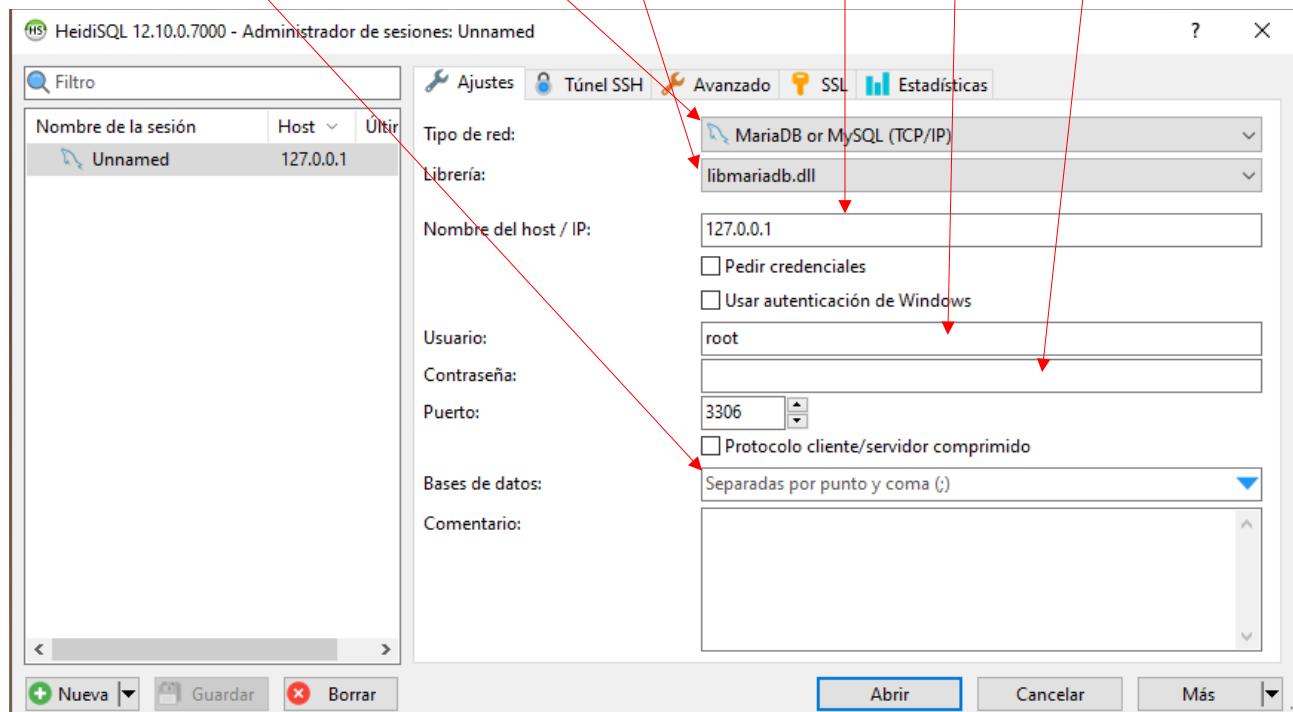
Ahora usaremos algún cliente de MySQL o genérico como **HeidiSQL**:



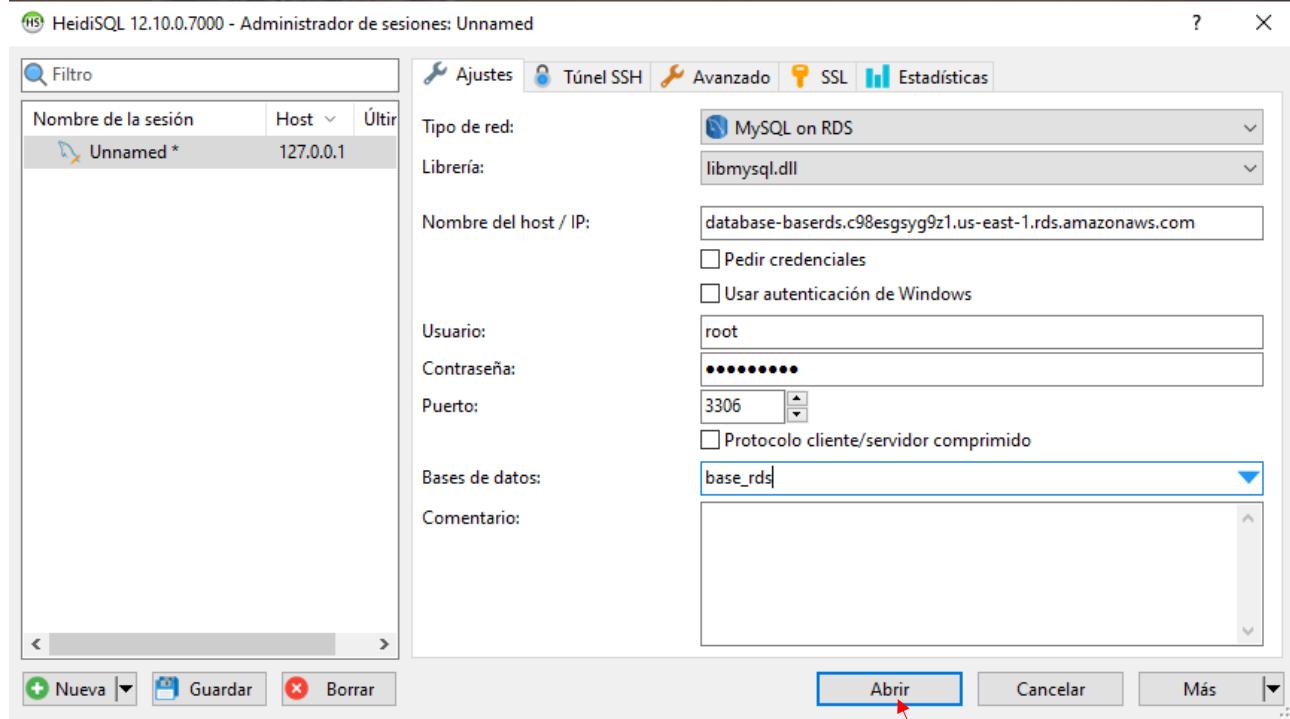
Este es su aspecto, antes de conectarla a un motor de bases de datos.

Damos a "**Nueva**" para empezar a configurar su conexión.

Tenemos que indicar el **motor**, la **versión** de librería, la **IP**, el **usuario** y la **contraseña** y por último el **nombre** de la base de datos.

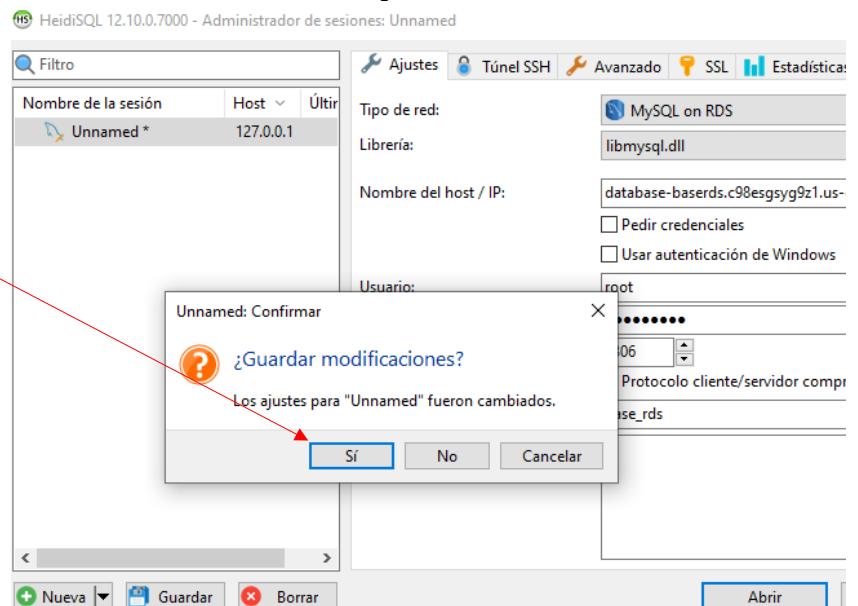


En nuestro ejemplo, queda así:

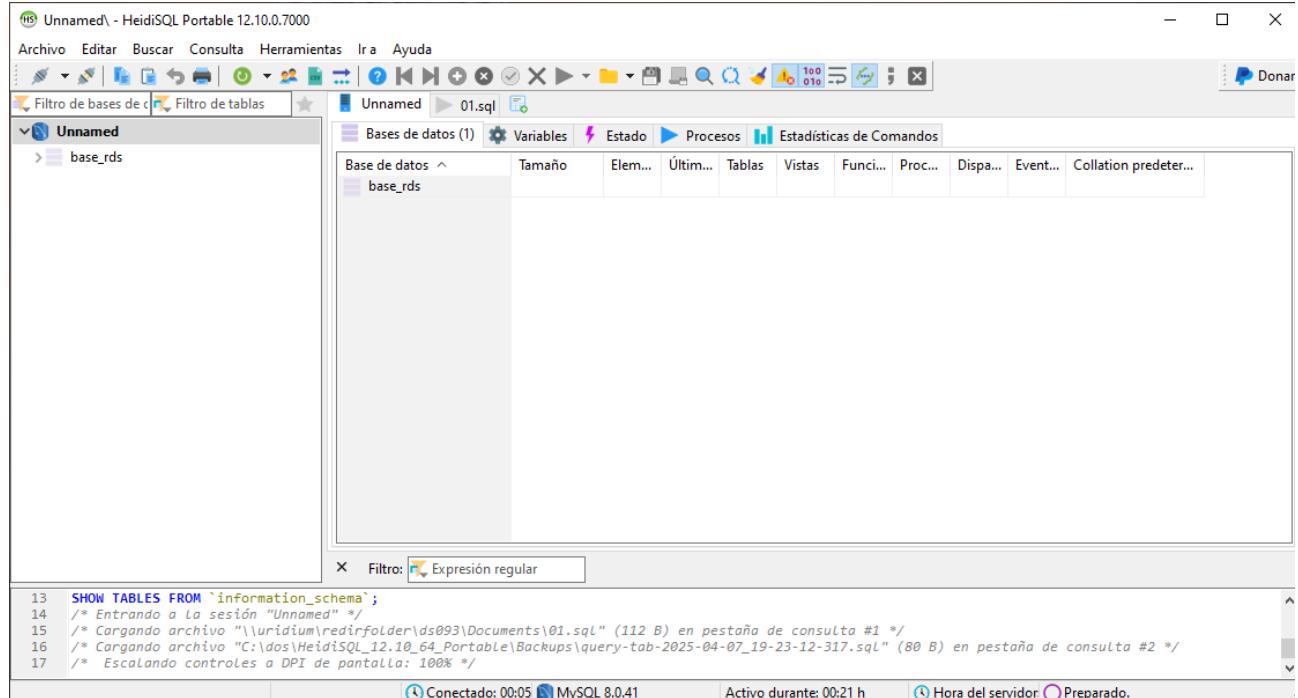


Se pulsa "Abrir" e intentará conectarse.

La primera vez suele indicarlo así
(sólo da Sí)



Si se ha producido la conexión, saldrá :



Sí se ha conseguido conectar!!!.

Como cualquier cliente de bases de datos (cliente, es cualquier programa que se conecta a un motor de bases de datos y desde él se puede consultar las tablas que tiene, además de añadir datos o hacer consultas), nos despliega visualmente los datos que contiene nuestra base de datos, que como está recién creada será de 0 kb (hay margen, la capa gratuita son 20 GB).

Esta la usaremos para meter datos desde nuestras futuras instancias.

Una vez que hemos cerrado el programa HeidiSQL y lo volvemos a ejecutar, este recuerda la última configuración. Queda organizado ponerle nombre a la sesión (botón derecho), sobre todo y usamos ese programa para conectarnos a diferentes motores de bases de datos (dando a Nueva, añadimos otra).

