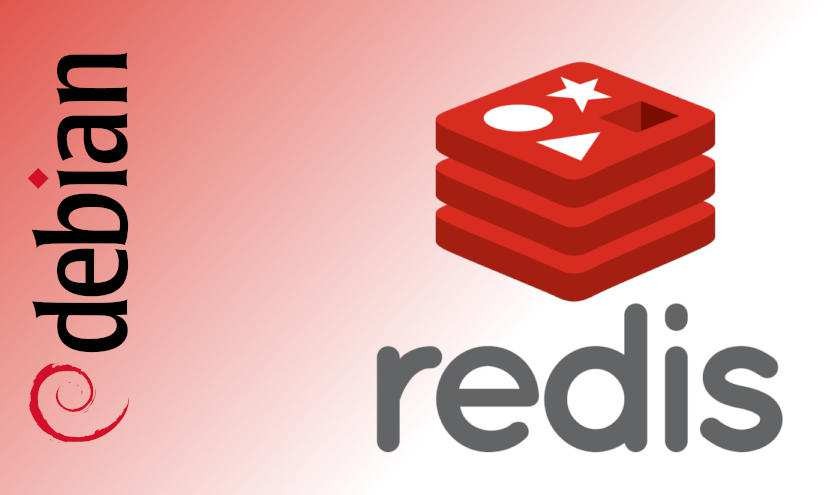
# Instalación de Servidor Redis en Debian 12



## Autor

Andrés Morales González

I.E.S. Gonzalo Nazareno (Dos Hermanas, Sevilla).

# Índice

[Instalación de Servidor Redis en Debian 12](#_bookmark0)  [Autor ](#_bookmark1)

[Índice](#_bookmark2)

[Base de datos NoSQL](#_bookmark3)  ¿Qué es Redis?

 [Diferencias entre Redis y Bases de Datos Relacionales](#_bookmark4)  [Diferencias Clave](#_bookmark5)

 [Instalación](#_bookmark6)

 [Conexión en remoto](#_bookmark7)  [Instalación del cliente](#_bookmark8)

# Base de datos NoSQL

### ¿Qué es Redis?

**Redis** es una **base de datos NoSQL** que almacena datos en **memoria RAM** en vez de en disco, lo que la hace muy rápida. Funciona con un modelo **clave-valor**, donde los datos se guardan usando una clave única que los identifica. Además, Redis soporta varios tipos de datos, como:

 **Cadenas** (texto simple)

 **Listas** (colecciones de valores ordenados)

 **Hashes** (estructuras clave-valor dentro de una clave)

 **Conjuntos** (colecciones de elementos únicos)

Redis se usa mucho para **caché**, **sesiones de usuario**, o sistemas que requieren **respuesta rápida** en tiempo real.

### Diferencias entre Redis y Bases de Datos Relacionales

En esta apartado lo que haré sera hacer una tabla comparativa entre una bases de daros relacion y no relacional.

|  |  |  |
| --- | --- | --- |
| **Característica** | **Redis (NoSQL)** | **Bases de Datos Relacionales (SQL)** |
| **Modelo de datos** | Clave-valor | Tablas con filas y columnas |
| **Esquema** | Sin esquema fijo | Esquema rígido (tipos de datos fijos) |
| **Consultas** | Basadas en clave | Consultas SQL complejas (joins) |
| **Almacenamiento** | En memoria (RAM) | En disco |
| **Velocidad** | Muy rápida | Menos rápida (usa disco) |
| **Relaciones** | No tiene relaciones entre datos | Relaciona tablas mediante claves |
| **Transacciones** | Soporte limitado | Soporte completo (ACID) |
| **Casos de uso** | Caché, sesiones, datos temporales | Aplicaciones empresariales complejas |

### Diferencias Clave

1. **Velocidad y Almacenamiento**: Redis es mucho más rápido porque usa RAM, pero es mejor para datos **temporales**. Las bases de datos relacionales almacenan datos en disco, lo que las hace más adecuadas para datos **permanentes**.
2. **Consultas**: Redis no usa SQL y trabaja con claves, mientras que las bases de datos relacionales permiten consultas complejas con SQL.
3. **Relaciones**: Redis no maneja relaciones entre datos, mientras que las bases de datos relacionales están diseñadas para relacionar diferentes tablas de datos.
4. **Uso**: Redis se usa para aplicaciones que requieren rapidez, como **caché** o **mensajería en tiempo real**. Las bases de datos relacionales son mejores para manejar **datos complejos** como los de sistemas financieros o de gestión empresarial.

## Instalación

Lo primero que haremos sera actualizar los paquetes de la máquina donde vamos a instalarla:

sudo apt update

Comprobación:

andy@servidor-redis:~$ sudo apt update && sudo apt upgrade Obj:1 <http://deb.debian.org/debian> bookworm InRelease

Obj:2 <http://security.debian.org/debian-security> bookworm-security InRelease

Obj:3 <http://deb.debian.org/debian> bookworm-updates InRelease Leyendo lista de paquetes... Hecho

Creando árbol de dependencias... Hecho Leyendo la información de estado... Hecho Todos los paquetes están actualizados.

Leyendo lista de paquetes... Hecho Creando árbol de dependencias... Hecho Leyendo la información de estado... Hecho Calculando la actualización... Hecho

0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 0 no actualizados.

Una vez que hemos actualizado sin ningún error, lo que haremos sera instalar Redis, de la siguiente manera:

sudo apt install redis-server -y

Comprobación:

andy@servidor-redis:~$ sudo apt install redis Leyendo lista de paquetes... Hecho

Creando árbol de dependencias... Hecho Leyendo la información de estado... Hecho

Se instalarán los siguientes paquetes adicionales: liblzf1 redis-server redis-tools

Paquetes sugeridos: ruby-redis

Se instalarán los siguientes paquetes NUEVOS: liblzf1 redis redis-server redis-tools

0 actualizados, 4 nuevos se instalarán, 0 para eliminar y 0 no actualizados.

Se necesita descargar 1.096 kB de archivos.

Se utilizarán 6.238 kB de espacio de disco adicional después de esta

operación.

¿Desea continuar? [S/n] s

Des:1 <http://deb.debian.org/debian> bookworm/main amd64 liblzf1 amd64 3.6-3 [10,2 kB]

Des:2 <http://deb.debian.org/debian> bookworm/main amd64 redis-tools amd64 5:7.0.15-1~deb12u1 [989 kB]

Des:3 <http://deb.debian.org/debian> bookworm/main amd64 redis-server amd64 5:7.0.15-1~deb12u1 [72,4 kB]

Des:4 <http://deb.debian.org/debian> bookworm/main amd64 redis all 5:7.0.15- 1~deb12u1 [24,6 kB]

Descargados 1.096 kB en 0s (3.861 kB/s)

Seleccionando el paquete liblzf1:amd64 previamente no seleccionado. (Leyendo la base de datos ... 72497 ficheros o directorios instalados actualment

e.)

Preparando para desempaquetar .../liblzf1\_3.6-3\_amd64.deb ... Desempaquetando liblzf1:amd64 (3.6-3) ...

Seleccionando el paquete redis-tools previamente no seleccionado. Preparando para desempaquetar .../redis-tools\_5%3a7.0.15- 1~deb12u1\_amd64.deb ...

Desempaquetando redis-tools (5:7.0.15-1~deb12u1) ...

Seleccionando el paquete redis-server previamente no seleccionado. Preparando para desempaquetar .../redis-server\_5%3a7.0.15- 1~deb12u1\_amd64.deb ..

.

Desempaquetando redis-server (5:7.0.15-1~deb12u1) ... Seleccionando el paquete redis previamente no seleccionado.

Preparando para desempaquetar .../redis\_5%3a7.0.15-1~deb12u1\_all.deb ... Desempaquetando redis (5:7.0.15-1~deb12u1) ...

Configurando liblzf1:amd64 (3.6-3) ... Configurando redis-tools (5:7.0.15-1~deb12u1) ... Configurando redis-server (5:7.0.15-1~deb12u1) ...

Created symlink /etc/systemd/system/redis.service →

/lib/systemd/system/redis-se rver.service.

Created symlink /etc/systemd/system/multi-user.target.wants/redis- server.service

→ /lib/systemd/system/redis-server.service. Configurando redis (5:7.0.15-1~deb12u1) ... Procesando disparadores para man-db (2.11.2-2) ...

Procesando disparadores para libc-bin (2.36-9+deb12u8) ...

Una vez que lo hemos instalado, verificaremos el estado para ver si se esta ejecutando correctamente:

sudo systemctl status redis

Y aqui vemos la comprobación:

andy@servidor-redis:~$ sudo systemctl status redis

* redis-server.service - Advanced key-value store

Loaded: loaded (/lib/systemd/system/redis-server.service; enabled;

preset:>

Active: active (running) since Fri 2024-10-11 19:10:52 CEST; 20s ago Docs: [http://redis.io/documentation,](http://redis.io/documentation)

man:redis-server(1) Main PID: 1256 (redis-server)

Status: "Ready to accept connections" Tasks: 5 (limit: 9474)

Memory: 9.3M CPU: 72ms

CGroup: /system.slice/redis-server.service

└─1256 "/usr/bin/redis-server 127.0.0.1:6379"

oct 11 19:10:52 servidor-redis systemd[1]: Starting redis-server.service - Adva>

oct 11 19:10:52 servidor-redis systemd[1]: Started redis-server.service - Advan>

lines 1-15/15 (END)

Si en el caso en el que no este activo o mejor dicho no se haya iniciado automaticamente, lo que tendremos que hacer será meter el siguiente comando:

sudo systemctl enable redis

Ahora probaremos desde la *máquina local* que este funcionando, con el siguiente comando, el cual nos tendrra que devolver *PONG*, para que todo este correcto:

redis-cli ping

Verificación del comando:

andy@servidor-redis:~$ redis-cli ping PONG

## Conexión en remoto

Como hemos visto en los anteriores SGBD donde hemos hecho la instalación , este sistema de gestión de bases de datos NoSQL. también permite contectar desde otras máquinas, para ello lo que tendremos que hacer sera modificar el fichero de configuración */etc/redis/redis.conf*, en el cual tendremos que hacer lo siguiente:

1. sudo nano /etc/redis/redis.conf
2. Buscar la linea la cual pone:
   1. *bind 127.0.0.1 ::1*
3. Sustituir esta línea por: *bind 0.0.0.0*
4. Busccar la linea donde pone *protected-mode yes* y la puedes comentar o poner: *protected-mode no*, esto lo hacemos para que acepte conexiones remotas.
5. Y por ultimo guardar los cambios y reiniciar el servicio de Redis, con el siguiente comando: *sudo systemctl restart redis*

## Instalación del cliente

Para ello lo que tendremos que hacer sera irnos a la máqyuina donde queremos el cliente de Redis, y hacer esta secuencia de comandos:

sudo apt-get update

sudo apt-get install redis-tools

y por ultimo conectarnos al servidor Redis en remoto:

andy@cliente-mariadb-mongo-postgres:~$ redis-cli -h 192.168.1.155 -p 6379

192.168.1.155:6379> PING PONG

192.168.1.155:6379>

Para hacer la comprobación de que la conexión remota funciona perfectamente meteremnos por comando

*PING*, y su respuesta deberia de ser *PONG*, esto nos indica que nos hemos conectado remotamente.

Sin embargo si hubiera algun fallo de conexión, nos saldria lo siguiente:

andy@cliente-mariadb-mongo-postgres:~$ redis-cli -h 192.168.1.155 Could not connect to Redis at 192.168.1.155:6379: Connection refused not connected> ping

Could not connect to Redis at 192.168.1.155:6379: Connection refused not connected> PING

Could not connect to Redis at 192.168.1.155:6379: Connection refused not connected> EXIT

Lo cual indica que o no hay conexión a internet, o bien lo que tendriamos que hacer es ir al archivo el cual configuramos, y ver que tenemos los valores perfectamente:

bind 0.0.0.0 protected-mode yes