interweb

Generated on 15/11/2026 00:10

KNOWLEDGE BASE · PUBLIC DOCUMENT

Reaching MySQL or PostgreSQL through an SSH tunnel, step by step

Remote Access & Code

CODE	VERSION	COUNTRY · LANGUAGE
RLX-AO-KB-748	v2026.10.18	Angola · English

Interweb Angola
RLX Angola, Lda. · Group RLX Holdings
Tax ID (NIF) 500 260 5724 · Commercial registration no. 26464-25/250724

https://www.meu.interweb.ao/knowledgebase/748

Reaching MySQL or PostgreSQL through an SSH tunnel, step by step

Remote Access & Code · Created on **October 10, 2026** · Updated on **October 18, 2026** · 3 min read



The database ports (**3306** for MySQL, **5432** for PostgreSQL) are not open from outside. An **SSH tunnel** solves that: it opens a port on your computer that leads, inside SSH, to the database on the server. To the database client, the connection is to `127.0.0.1`.

Step by step

1. **Confirm you have SSH.** On shared hosting the port is **2299**, and every plan includes SSH. The server address and the user are in **your cPanel access details**.

2. **Open the tunnel** in a terminal on your computer (Linux, macOS, or Windows 10 and 11, which ship with `ssh`). For MySQL:

```
ssh -N -p 2299 -L 3307:127.0.0.1:3306 user@server
```

3. For PostgreSQL, the same idea with other ports:

```
ssh -N -p 2299 -L 5433:127.0.0.1:5432 user@server
```

4. **Leave the window open.** The `-N` tells SSH not to open a shell, only the tunnel. Type the password when asked.

5. **In the client** (MySQL Workbench, DBeaver, HeidiSQL, `psql`), connect to **127.0.0.1**, port **3307** (or **5433**), with the **database user and password**, not the SSH ones.

6. When you finish, close the tunnel window with `Ctrl+C`.

Field	Value	Why
Local port (`-L`)	3307 or 5433	A free port on your computer. We use numbers different from the server's so they do not clash with a database you may have installed.
Destination	`127.0.0.1:3306` or `:5432`	Seen from the server. It is the database, on the server itself.
Database user	the full name, with prefix	The tunnel enters through SSH, but the database asks for its own user.
Server in the client	`127.0.0.1`	Not the server IP: the client talks to the local port of the tunnel.

If the tunnel opens but the client will not connect,
there are three usual causes: the client points at the server IP and not at `127.0.0.1`; the local port you chose is already in use (change the number); or the database user is written without the account prefix. If SSH itself will not let you in, see **the four usual causes**.

Clients that open the tunnel for you

You do not have to open the tunnel by hand. **MySQL Workbench** has the *Standard TCP/IP over SSH* connection method, and **DBeaver** has an *SSH* tab on the connection: you fill in the SSH host and port, the SSH user and password, and then the database details as if the database were on the same computer. On the first connection SSH asks whether you trust the server's fingerprint: check it and answer yes. Whenever you can, use an **SSH key** instead of a password.

Tunnels that drop?
Add `-o ServerAliveInterval=60` to the command so SSH keeps the link alive. On Windows, PuTTY does the same under «Connection, SSH, Tunnels». And if your network changes IP, the tunnel will not complain: just open it again.

NEED A HAND?

The tunnel opens and the database is still out of reach? Tell us which client you use and the error message and we will look.

Open a support ticket →

SEE ALSO

- **Letting an outside address reach your MySQL: Remote MySQL in cPanel**
- **Remote access to PostgreSQL: what works and what does not**
- **Connecting to your MySQL database with MySQL Workbench**

interweb

Interweb Angola

RLX Angola, Lda. · Group RLX Holdings

TAX ID (NIF)	500 260 5724
COMMERCIAL REGISTRATION NO.	26464-25/250724
REGISTERED OFFICE	Rua Direita do Patriota, Bairro Dangereux Belas, Luanda, Angola
CONTACT	info@interweb.ao · +244 942 011 191 · www.interweb.ao
OPENING HOURS	Monday to Friday - 08 AM to 04 PM Saturday & Sunday - Closed

Online version of this document

RLX-AO-KB-748 · v2026.10.18 · Generated on 15/11/2026 00:10