

KNOWLEDGE BASE · PUBLIC DOCUMENT

MySQL, MariaDB or PostgreSQL open to the internet: closing the port without breaking apps

Security

CODE

RLX-MZ-KB-2258

VERSION

v2026.10.21

COUNTRY · LANGUAGE

Mozambique · English

Interweb Mozambique

RLX Holdings, S.A. · Group RLX Holdings

Tax ID (NUIT) 401 191 607 · Legal entity no. (NUEL) 105 045 974 · Economic activity code (CAE) 62022

<https://www.meu.interweb.co.mz/knowledgebase/2258>



For VPS and dedicated server customers with root access. The short answer: a database that answers the internet (MySQL or MariaDB on port **3306**, PostgreSQL on **5432**) gets login attempts around the clock, and a national incident response team (CERT) reports it to our network and data centre provider. It is almost never needed: applications on the same server connect through localhost. You close it in three moves: **see who connects from outside**, make the database **listen internally only**, and let only the IPs that need it through the firewall, or use an **SSH tunnel**. Before closing, watch for the two cases that break: applications on another server and applications in Docker containers.

Who is affected when you close the port

Who connects to the database	What happens
An application on the same server, with host <code>localhost</code> or <code>127.0.0.1</code>	Nothing: it keeps working.
An application in a Docker container connecting to the host's database (through <code>172.17.0.1</code> or <code>host.docker.internal</code>)	It loses the connection if the database listens on <code>127.0.0.1</code> only. Put the database on the same compose network, or make it also listen on the Docker bridge address and limit that port in the firewall.
The database in a container published with <code>-p 3306:3306</code> or <code>-p 5432:5432</code>	Docker publishes it on every address and goes around ufw. Change it to <code>127.0.0.1:3306:3306</code> , or remove the mapping.
Another of your servers, an office, a reporting tool	It loses the connection. Open the port to that IP only, or use an SSH tunnel.
You, with Workbench or DBeaver	Use an SSH tunnel: no open port needed. SSH tunnel, step by step.

Closing the port, in the right order

1

Check whether the database listens to the outside: `sudo ss -tlnp | grep -E ':(3306|5432) ' 0.0.0.0 , [::]` or the public IP: open. 127.0.0.1 : internal only. `docker-proxy` : published by a container.

2

See who connects today, before changing anything: `sudo ss -tnp state established '( sport = :3306 or sport = :5432 )'` In MySQL or MariaDB: `SELECT user, host FROM mysql.user;` `SHOW PROCESSLIST;` (accounts with host `%` accept connections from anywhere). In PostgreSQL: `SELECT client_addr, username, datname FROM pg_stat_activity;` Write down every IP other than 127.0.0.1 and work out which application it is.

3

MySQL or MariaDB, if nobody outside needs it: copy the configuration file, then in the `[mysqld]` section leave `[mysqld]` `bind-address = 127.0.0.1` The file varies: Ubuntu with MySQL, `/etc/mysql/mysql.conf.d/mysqld.cnf` ; Debian and Ubuntu with MariaDB, `/etc/mysql/mariadb.conf.d/50-server.cnf` ; elsewhere, `/etc/my.cnf` or `/etc/my.cnf.d/` . Restart (`sudo systemctl restart mysql` , or `mariadb` , or `mysqld`) and check with `sudo mysql -e "SHOW VARIABLES LIKE 'bind_address';"` .

4

PostgreSQL, if nobody outside needs it: find the files with `sudo -u postgres psql -c 'SHOW config_file;'` `sudo -u postgres psql -c 'SHOW hba_file;'` In the first, set `listen_addresses = 'localhost'` ; in the second, remove the `host` lines with `0.0.0.0/0` or `::/0` . Then `sudo systemctl restart postgresql` . More in **remote PostgreSQL connections**.

5

If an outside application really must connect directly, the database keeps listening on the public address and the firewall decides: that application's IP only. `sudo ufw status numbered`
`sudo ufw delete N`
`sudo ufw allow from 203.0.113.10 to any port 3306 proto tcp`
(`N` is the number of the rule open to everyone; replace the example IP with the application's, and 3306 with 5432 for PostgreSQL). An SSH tunnel or a VPN (**WireGuard**) is better.

6

If the database runs in Docker, change the mapping to `"127.0.0.1:3306:3306"` (or remove it) and run `docker compose up -d` : **Docker on a VPS, safely**.

7

Test from outside and change the passwords: from another machine, `nc -zv YOUR_SERVER_IP 3306` must fail. Then change the passwords of the database accounts that were exposed, delete accounts with host `%` you do not need, and update the applications: **where each application keeps its connection details**.

8

Reply to the support ticket in Meu Interweb, within the notice's deadline (short, often a single day), with a short report: the cause, what you changed and how you checked that the port no longer answers from outside. There is a template in **writing the reply to an abuse notice**. With no reply, or if the problem carries on, the server may be switched off to protect the network, with your data left intact: **why a server is switched off and how to bring it back**.

A database that was open may already have had visitors. Signs: new databases or tables with a ransom note, empty tables, accounts you never created. Do not pay: there is no guarantee at all that the data comes back. If you find signs that someone already got in, closing the port is not enough: the safest path is to **reinstall the server and restore a clean copy (which backup to restore)**. Cleaning up in place is for people who know exactly what they are doing. See also **is my VPS compromised?**.

You do not need an open port to work on the database from home: an SSH tunnel takes Workbench or DBeaver to the server's 127.0.0.1, and to anyone without your SSH key the port does not exist. On a VPS, root access is yours: we do not log in to the server and cannot change a firewall that runs inside it. If you lose SSH, tell us in the support ticket.

Not sure which application connects to the database from outside? Send us the step 2 output in the support ticket.

[Open a support ticket](#)

SEE ALSO

An abuse notice about your server: what it means, the deadline and what we need from you

Writing the reply to an abuse notice: cause, what you removed and how it will not happen again

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

Allowing remote PostgreSQL connections on a VPS, safely

Running Docker on a VPS safely: the precautions that matter

Where each application keeps its database connection details



Online version of this document

Interweb is a leading internet services company, with a mission to help businesses to be born and prosper through technology.

2020

FOUNDED

37,000+

CLIENTS

130+

SIGN-UPS A DAY

Interweb Mozambique

RLX Holdings, S.A. • Group RLX Holdings

TAX ID (NUIT)	401 191 607
LEGAL ENTITY NO. (NUEL)	105 045 974
ECONOMIC ACTIVITY CODE (CAE)	62022
REGISTERED OFFICE	Bairro 7 de Setembro Vilankulo 1304 • CEP 0411-02, Inhambane, Mozambique
CONTACT	info@interweb.co.mz • +258 84 635 1111 • +258 86 635 1111 • www.interweb.co.mz
OPENING HOURS	Monday to Friday - 08 AM to 04 PM Saturday & Sunday - Closed