

KNOWLEDGE BASE · PUBLIC DOCUMENT

# Redis open to the internet: close it with bind, protected mode, a password and the firewall

Security

CODE

**RLX-AO-KB-2252**

VERSION

**v2026.10.21**

COUNTRY · LANGUAGE

**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/2252>

# Redis open to the internet: close it with bind, protected mode, a password and the firewall

📁 Security 📅 Created on October 21, 2026 ⌚ 5 min read



**For VPS and dedicated server customers with root access. The short answer:** Redis is built to talk only to trusted applications, on the same machine or a private network. If it answers on port **6379** to any address on the internet, anyone can read and wipe the data, and automated attacks use an open Redis to write files onto the server and get in. That is why a national incident response team (CERT) reports it, and why our network and data centre provider sends us the notice we pass on to you (**what an abuse notice is**). You close it in four layers: **bind** to the local address only, **protected mode**, a **password** and the **firewall**.

## 🔍 Why Redis ended up open

| What you find                                                                                                                                     | Most likely cause                                                                                                                                                      |
|---------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>bind 0.0.0.0</code> in <code>redis.conf</code> , or the <code>bind</code> line commented out                                                | Someone opened Redis to connect from another machine, or followed a guide that said to.                                                                                |
| Redis in a Docker container published with <code>-p 6379:6379</code>                                                                              | Docker publishes the port on every address and goes around ufw: the firewall says it is closed and the port still answers. See <b>running Docker on a VPS safely</b> . |
| <code>protected-mode no</code>                                                                                                                    | It was switched off to “fix” a connection error. Without it, and without a password, Redis accepts anyone.                                                             |
| Firewall off, or a rule that lets 6379 through                                                                                                    | The rule was left over from a test and nobody removed it.                                                                                                              |
| Redis <code>dir</code> pointing at <code>/root/.ssh</code> , <code>/etc/cron.d</code> or <code>/var/spool/cron</code> , or keys you never created | A sign that someone already used the open Redis to write files: treat the server as compromised ( <b>is my VPS compromised?</b> ).                                     |

## ➔ Closing Redis without breaking your applications

- 1 See what is listening:** `sudo ss -tlnp | grep 6379`  
`redis-cli CONFIG GET bind`  
`redis-cli CONFIG GET protected-mode`  
`redis-cli CONFIG GET dir` If you see `0.0.0.0:6379`, `:::6379` or the `docker-proxy` process, it is open. If Redis already has a password, use `redis-cli --askpass`.
- 2 Before closing it, find out who uses it:** `sudo ss -tnp state established '( sport = :6379 )'`  
`redis-cli CLIENT LIST` Applications on the same machine connect from `127.0.0.1` and are not affected. If you see the IP of another of your servers, write it down: you will need it for the firewall or a tunnel.
- 3 Close it in the configuration file** (usually `/etc/redis/redis.conf` on Debian and Ubuntu; `/etc/redis.conf` on some other distributions). Copy the file first, then leave these lines: `bind 127.0.0.1`  
`protected-mode yes`  
`requirepass A-LONG-PASSWORD` Restart with `sudo systemctl restart redis-server` (on some distributions the service is called `redis`).
- 4 If Redis runs in Docker**, change `"6379:6379"` to `"127.0.0.1:6379:6379"` in `docker-compose.yml`, or remove the `ports` section if only other containers use it (they talk over the compose network). Then `docker compose up -d`.
- 5 Give the new password to the applications** that use Redis (WordPress object cache, queues, sessions): **object cache with Redis** and **where each application keeps its connection details**.
- 6 Close it in the firewall too:** `sudo ufw status numbered`  
`sudo ufw delete N`  
`sudo ufw allow from 203.0.113.10 to any port 6379 proto tcp` `N` is the number of the 6379 rule. The last line is only for when another of your servers really needs Redis: replace the example IP with its address. Full guide in **setting up ufw without locking yourself out**.
- 7 Test from another machine** (not the server itself): `redis-cli -h YOUR_SERVER_IP ping` or `nc -zv YOUR_SERVER_IP 6379`. It must fail or time out. If it answers PONG, or NOAUTH, the port is still open to the internet.

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**.

**Do not restart Redis with a new bind and password before you know who uses it:**

Applications connecting from another machine, or without a password, stop working at once. Check in step 2 and update them in step 5. 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.

**If only one of your servers needs Redis,**

an SSH tunnel or a VPN (**WireGuard on a VPS**) is safer than opening the port, even to a single IP. 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.

 **NEED A HAND?**

Received an open Redis notice? Reply in the support ticket with the output of the step 1 commands and what you changed.

[📄 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](#)[Is my VPS compromised? Checking processes, cron jobs, users, SSH keys and temporary folders](#)[Setting up the ufw firewall on Ubuntu without locking yourself out](#)[Running Docker on a VPS safely: the precautions that matter](#)[Object cache with Redis for WordPress on a VPS: what it speeds up and how to set it up](#)

