interweb

Generated on 15/11/2026 00:12

KNOWLEDGE BASE · PUBLIC DOCUMENT

Updating and backing up Evolution API

Evolution API

CODE	VERSION	COUNTRY · LANGUAGE
RLX-AO-KB-978	v2026.10.10	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/978

Updating and backing up Evolution API

Evolution API · Created on **October 10, 2026** · 3 min read



The rule is simple: **backup first, update second, one version at a time**. Evolution API depends on an unofficial library that follows WhatsApp's changes, so new versions appear at a pace you do not control. What you control is when you update and what you have saved if it goes wrong.

What has to be copied

What	Why
The database (PostgreSQL or MySQL)	It holds the instances and the data. It is the copy that counts most. See **backups with mysqldump and pg_dump**.
The instance volumes	The sessions that the compose file tells Docker to keep. Without them, the database copy may not be enough to come back linked.
The settings file	It holds the API key and the passwords. Without it you cannot recreate the service. Keep it somewhere private, off the server.
The compose file	It says which version was running and how it was put together.

A copy that sits on the same server protects against your own mistake, not against losing the server. Take it off. For the options, see **backing up a VPS: snapshots or files**.

Updating, step by step

1. **Make the backup** of the database, the volumes and the files, and check that you can open it.

2. **Read the release notes** in the project's repository. Look for changes that break the configuration, renamed settings and warnings about the database.

3. **Change the image version** in the compose file to the one you chose. Prefer a specific version to following “latest”.

4. **Pull and recreate:** `docker compose pull` then `docker compose up -d`.

5. **Read the logs** with `docker compose logs -f`. The API must start without errors and the instances must come back linked.

6. **Test for real:** send a message and receive another through the webhook before you call the update good.

> **Going back is not always just swapping the version.**
> If the new version changed the database, the old one may no longer understand it. That is why the backup comes before the update: restoring that copy and going back to the previous version is the safe way back. For the general principle, see **updating a Docker application without losing data**.

When to update

1. **When there is a problem the new version fixes**, such as an instance stuck on “connecting”. See **the usual causes**.

2. **When there is a security fix.**

3. **Outside usage hours.** The update interrupts the service for a while, and messages arriving in that gap can be lost.

Each time an update works, note the version and the date. Next time something fails, knowing the last good one saves hours. And test a restore once, on a rehearsal server, before you need it for real. Maintenance is yours, as the **Support Policy** says.

RECOMMENDED

Need room for backups and a rehearsal server? Have a look at the VPS plans.

See VPS servers →

SEE ALSO

- **Backups with mysqldump and pg_dump**
- **Updating a Docker application without losing data**
- **Backing up a VPS**

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-978 · v2026.10.10 · Generated on 15/11/2026 00:12