interweb

Generated on 15/11/2026 00:11

KNOWLEDGE BASE · PUBLIC DOCUMENT

Backing up with mysqldump and pg_dump, and restoring the copy

Export, Import & Backup

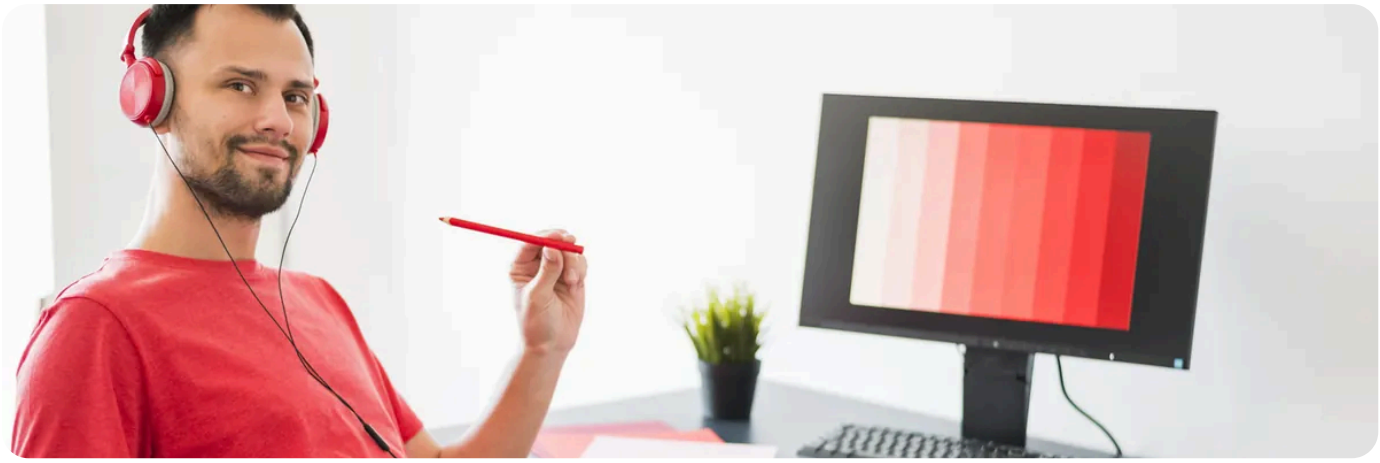
CODE	VERSION	COUNTRY · LANGUAGE
RLX-AO-KB-760	v2026.10.10	Angola · English

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/760

Backing up with mysqldump and pg_dump, and restoring the copy

Export, Import & Backup Created on **October 10, 2026** 2 min read



Over SSH, you copy a MySQL or MariaDB database with `mysqldump` and a PostgreSQL database with `pg_dump`. Each produces a file you restore with `mysql` or with `psql` / `pg_restore`. It is the right method for large databases and for automatic copies. You need an account with **shell access** (port 2299 on shared hosting) or a VPS. See **the cPanel Terminal**.

MySQL and MariaDB

1. **Copy:**

```
mysqldump -u account_user -p --single-transaction account_database | gzip > copy.sql.gz
```

The `-p` asks for the password. The `--single-transaction` gives a consistent copy of InnoDB tables without locking them.

2. **Restore:** into an empty database (or one that already has tables, if the file drops them first):

```
gunzip -c copy.sql.gz | mysql -u account_user -p account_database
```

3. **Check:** `gunzip -c copy.sql.gz | tail -n 3`. `mysqldump` ends with a comment saying *Dump completed*. If you do not see it, the copy stopped halfway.

PostgreSQL

1. **Copy,** in PostgreSQL's own (compressed) format:

```
pg_dump -h 127.0.0.1 -U account_user -Fc -f copy.dump account_database
```

2 **Restore** with `pg_restore`, into an empty database:

```
pg_restore -h 127.0.0.1 -U account_user -d account_database copy.dump
```

3 **Or as plain text,** restored with `psql`: `pg_dump -h 127.0.0.1 -U account_user account_database > copy.sql` and then `psql -h 127.0.0.1 -U account_user -d account_database -f copy.sql`.

Tool	Format	Restore with
`mysqldump`	SQL text (compress it with `gzip`).	`mysql`
`pg_dump -Fc`	Own format, already compressed.	`pg_restore`
`pg_dump` **without** `-Fc`	SQL text.	`psql -f`

A copy you have never restored is not a copy.
Test the restore into a scratch database. And take care with the password on the command line: it stays in the shell history. To avoid that, MySQL reads it from `~/.my.cnf` and PostgreSQL from `~/.pgpass`, both yours alone (`chmod 600`).

Naming the copies and putting them on cron

Give the copies a name with the date: `copy-$(date +%F).sql.gz` adds the day to the name. If you put the command on cron, mind this: **in a crontab the `%` sign needs a backslash**, written `\%`, or cron cuts the command short. Do not put the password on the cron line: use `~/.my.cnf`. Finally, decide how many copies to keep and delete the older ones, or a task that runs every night ends up filling the account. See **cron jobs**. And keep at least one copy **outside** the server that holds the database, for example on your own computer.

Automate it and keep a copy off the server.
A copy that sits on the same machine as the database is lost with it. To put it on cron, see **automatic backups of your own application**. For a VPS, see **backing up a VPS**.

NEED A HAND?

The restore fails and you cannot see why? Send us the first error message, without the password.

Open a support ticket →

SEE ALSO

- **Automatic backups of your own application: cron and mysqldump**
- **Importing a large database over SSH**
- **How to restore your data with JetBackup**

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-760 · v2026.10.10 · Generated on 15/11/2026 00:11