interweb

Generated on 15/11/2026 00:09

KNOWLEDGE BASE · PUBLIC DOCUMENT

Importing a large database over SSH

Export, Import & Backup

CODE	VERSION	COUNTRY · LANGUAGE
RLX-AO-KB-526	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/526

Importing a large database over SSH

Export, Import & Backup Created on **September 27, 2026** Updated on **October 18, 2026** 5 min read



phpMyAdmin handles most imports. But there is a size beyond which it gives up: either the file will not upload, or it uploads and the import stops halfway without saying anything. That is not a fault; those are the time and size limits that exist to protect everybody on the machine.

Over SSH there is no such ceiling. The import runs on the server itself, with no browser in the middle, and a file of several gigabytes goes in without drama.

First: does your account have SSH?

Every plan has SSH. You can find it in half a minute:

1. **Open cPanel and look for *Terminal* or *SSH Access*.** Both appear on every plan. *Terminal* is arguably better than SSH at home: it runs in the browser and needs nothing installed.

2. **If neither is there**, try connecting anyway from your own computer with `ssh -p 2299 user@server`. The port here is **2299**, not 22. If you are refused, do not keep trying: read the next point.

3. **Ask us.** If the connection is refused, tell us what it is for and we will look into it, or run the import for you. If all you want is to get one database in once, sending us the file may well be quicker.

! **Repeated failed attempts get your IP blocked.**
The server firewall counts failed logins and, after a few, locks out the address they came from, which also cuts off your website and your e-mail. If you get the password wrong twice, stop and check the details. If you are already locked out, see **how to unblock your IP address**.

The import, step by step

1. **Create the database and user in cPanel** and attach them with full privileges. The file you are importing does NOT create the database: it expects to find it there. See **creating databases and using phpMyAdmin**.

2. **Get the file onto the server.** Over SFTP on port 2299, through the cPanel file manager, or, if the file is already on the internet, with `wget` from the command line, which is far quicker than routing it through your own computer.

3. **If it is compressed, do not unpack it first.** You can read and import in one go, which saves both disk space and time: `gunzip < copy.sql.gz | mysql -u user -p databasename` .

4. **If it is plain text**, the line is `mysql -u user -p databasename < copy.sql` . It asks for the password and starts.

5. **Wait, in silence.** A successful import prints nothing at all: it simply finishes and gives you the prompt back. If text appears, that is an error, and the error names the line of the file it stopped on.

Do not put the password on the command line.
Use `-p` on its own and let it ask. Stuck onto `-p` , the password lands in your shell history and shows up to anyone listing server processes at that moment.

Long imports, and connections that drop

On a big database an import can take a long while. If your internet wobbles, the SSH session dies and takes the import with it, halfway, leaving the database in an incomplete state that is worse than empty.

The usual cure on other servers is `screen` or `tmux` . **Neither exists here.** What does exist is `nohup` , which does the same job for this case: it launches the work so that it survives the session dropping.

To	The line
Launch and walk away	`nohup bash -c "mysql -u user -pPASSWORD db < copy.sql" > ~/import.log 2>&1 &`
Watch it go	`tail -f ~/import.log`
Check it finished	`ps aux | grep mysql`

Note that here the password does have to go on the line, because nobody is around to type it afterwards. Change it straight after, or clear your shell history.

And the other way round: exporting

The same machinery in reverse, and the most reliable way to take a copy of a large database: `mysqldump -u user -p databasename | gzip > copy.sql.gz` . The file lands in whatever folder you were in, and you fetch it over SFTP.

Importing overwrites.
If the database already had tables with the same names, the file replaces them, and what was there is gone with no warning and no confirmation. Export what is already there BEFORE importing over it, or take a copy of the whole account: **restoring with JetBackup**.

If the site will not open after the import, the connection details in the configuration file are almost certainly still pointing at the old database: see **where each application keeps its connection details**.

NEED A HAND?

Database too big, or SSH not working for you? Send us the file and we will import it.

Open a support ticket →

Where to see it in your account

Before importing, check the space. The .sql file and the database it fills both take room from your plan (10 GB on Pro, 20 GB on Plus, 30 GB on Ultra, 50 to 200 GB on the Business plans), together with your site and mailboxes. In cPanel, the Statistics sidebar shows the space used and the limit.

A long import is also a heavy job: afterwards, Resource Usage shows whether it reached the CPU or memory limit and when. To know whether your account has SSH, look for Terminal or SSH Access in cPanel; if neither is there, use the last step of this article and send us the file.

SEE ALSO

- **Hosting plans and what each one includes** →
- **Hosting migration: we bring the site across for you** →

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-526 · v2026.10.18 · Generated on 15/11/2026 00:09