interweb

Generated on 28/11/2026 20:43

KNOWLEDGE BASE · PUBLIC DOCUMENT

Using aio on a server or in scripts: AIO_API_KEY, --json and the exit codes

The aio Command

CODE	VERSION	COUNTRY · LANGUAGE
RLX-AO-KB-3086	v2026.11.28	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/3086

Using aio on a server or in scripts: AIO_API_KEY, --json and the exit codes

The aio Command Created on **November 28, 2026** 3 min read



`aio` also works with nobody at the keyboard: on a server, in a scheduled task or in a CI pipeline. Three things make the difference: the key in the `AIO_API_KEY` variable, the `--json` option and the exit codes.

The key: AIO_API_KEY

On a server you do not need `aio auth login`. The `AIO_API_KEY` variable takes priority over the saved key and is not written in any project file:

```
export AIO_API_KEY="prod_..."
```

In a CI pipeline, keep the key in the system secrets and pass it as `AIO_API_KEY`. Use a key just for this, with `ai:read` and `ai:write`: if you ever have to revoke it, your other programs keep working. The IP list and the spending limit of the key also help here: see **creating an API key**.

Answers for another program: --json

Every command accepts `--json`. The answer comes out as JSON, ready for another program to read, instead of the table or text made for people:

```
aio balance --json
aio models --type image --json
```

The exit codes

Code	When
`0`	All went well.
`1`	There was an error: the key, the balance, the model, the request limit or something else.
`2`	The command is written wrong.

In a shell script, check the code right after the command:

```
aio generate image "a lighthouse at sunset" --out images/
code=$?
if [ "$code" -ne 0 ]; then echo "aio ended with code $code"; exit "$code"; fi
```

The error message has the code, the HTTP status and the request id. Keep it in the script log: it is what support needs to find the request. The key never appears in the messages.

Other useful variables

Variable	What it is for
`AIO_LANG`	The language of the messages: `pt` or `en`. Without it, `aio` follows the system `LANG`.
`NO_COLOR`	Turns off colors and bold, like the `--no-color` option. Logs stay clean.
`AIO_TIMEOUT`	The waiting time for each API request, in seconds. By default, 120.
`AIO_POLL_SECONDS`	How many seconds `aio` waits between checks of a generation.
`AIO_MODEL` and `AIO_IMAGE_MODEL`, `AIO_VIDEO_MODEL`, `AIO_AUDIO_MODEL`	The default models for chat and for each type of generation.
`AIO_CONFIG_DIR`	Another folder for the settings.

Each command counts towards the requests per minute limit of the key.
In a loop, space out the requests. If you get a `429`, wait a little and try again. The numbers are in **API request limits**.

Test the script with a `sandbox_` key. Chat always answers "Resposta de teste (sandbox).", generations are completed at once with a sample file and `0` credits, and the model list, the prices and the balance are the real ones.

RECOMMENDED

Want to put AI in your scripts? See All in One AI.

See All in One AI →

SEE ALSO

- **Creating an API key: name, mode, permissions, validity, IP list and spending limit**
- **API request limits: per key, per account and per task, the headers and what to do after a 429**
- **Testing All in One AI in the Sandbox at no cost: what each answer looks like**
- **aio errors: no key, 401, 402, the waiting time and Node.js 18**
- **All in One AI**

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-3086 · v2026.11.28 · Generated on 28/11/2026 20:43