Mozout

Generated on 13/11/2026 19:51

KNOWLEDGE BASE · PUBLIC DOCUMENT

Looking after your API keys: one per program, expiry, revoking and replacing a key

WhatsApp & APIs

CODE	VERSION	COUNTRY · LANGUAGE
RLX-MZ-KB-2749	v2026.11.13	Mozambique · English

MozOut Mozambique
RLX Holdings, S.A. · Group RLX Holdings
Tax ID (NUIT) 401 191 607 · Legal entity no. (NUEL) 105 045 974 · Economic activity code (CAE) 62022

https://www.meu.mozout.co.mz/knowledgebase/2749

Looking after your API keys: one per program, expiry, revoking and replacing a key

WhatsApp & APIs Created on **November 13, 2026** 3 min read



The short answer: create **one key per program**, with only the permissions it needs and a short validity. If a key leaks, revoke it at once on the **API page in Meu MozOut**; the others keep working. A lost key cannot be recovered: you revoke it and create another.

Why one key per program

If your website, your backup script and your billing system share one key, a single break-in exposes all three, and revoking the key stops all three at once. With a key for each, you know which one does what and you can cut off only the one in trouble. You can have up to **20 keys** per account, which is plenty to keep things apart.

What a key can never do

Whatever its permissions, a key can never **create another key**, **change the account password or e-mail**, or **see card details**. So whoever gets hold of a key does not own the account. They can, however, do everything that key allows, which is why minimal permissions matter.

When a key stops working

Situation	API answer
The key is missing or not valid	`401 unauthorized`
The key was revoked or has expired	`401 unauthorized`
The person who created the key no longer has access to the account	`401 unauthorized`
The request comes from an IP that is not on the key's list	`403 ip_not_allowed`
The key lacks the permission the operation needs	`403 insufficient_scope`
The account cannot use the API right now	`403 forbidden`

What to do about each one is in **the API error codes**.

Replacing a key without stopping your program

When the validity is running out, or when you want different permissions, swap keys in this order:

1. **Create the new key** on the **API page**, with the right permissions and validity.
2. **Put it in the program**, in the environment variable or secrets vault where the old one lived.
3. **Check that it works** with `GET /v1/token` : it should return the new key's name, mode and permissions.
4. **Revoke the old key.** From then on, any request made with it gets a `401` .

Key exposed
in a repository, an e-mail, a chat or a web page's code? Do not wait for a tidy swap: revoke it now and create another. The full order of steps is in **what to do when a key leaks**. Remember that a Sandbox key also reads real account data.

Good habits

1. **Keep the key in an environment variable or a secrets vault**, never in your code or a repository.
2. **Use the IP list** when the program runs on a server with a fixed address.
3. **Prefer short validities** (30 or 90 days) and put the swap date in your calendar.
4. **When you contact support, never send the key.** Send the `X-Correlation-ID` from the answer, the time and the operation: that is enough to find the request.

Keys created while the API is in preparation
, in either mode, are revoked when the API opens, and the account owner gets a notice by e-mail. On that day, create new keys and swap them into your programs.

RECOMMENDED

Got a key that is expiring, lost or exposed? Revoke it and create another on the API page in Meu MozOut.

Create a key →

SEE ALSO

- Creating an API key: name, mode, permissions, validity, IP list and spending limit →
- An API key or token has leaked: what to do, in the right order →
- API permissions by area: read or write, and which permission each task needs →
- Our API's error codes: what each one means, and which ones are worth retrying →
- API Sandbox or Production: what each key really does, and moving your program to production →
- Sensitive API permissions and the spending limit: paying, buying, reinstalling, cPanel access →

Mozout

MozOut Mozambique

RLX Holdings, S.A. · Group RLX Holdings

TAX ID (NUIT)	401 191 607
LEGAL ENTITY NO. (NUEL)	105 045 974
ECONOMIC ACTIVITY CODE (CAE)	62022
REGISTERED OFFICE	Bairro 7 de Setembro Vilankulo 1304 · CEP 0411-02, Inhambane, Mozambique
CONTACT	info@mozout.co.mz · +258 84 635 1111 · +258 86 635 1111 · www.mozout.co.mz
OPENING HOURS	Monday to Friday - 08 AM to 04 PM Saturday & Sunday - Closed

Online version of this document

RLX-MZ-KB-2749 · v2026.11.13 · Generated on 13/11/2026 19:51