Mozout

Generated on 15/11/2026 00:39

KNOWLEDGE BASE · PUBLIC DOCUMENT

Error listen EADDRINUSE address already in use: finding what holds the port

Node.js & JavaScript

CODE	VERSION	COUNTRY · LANGUAGE
RLX-INT-KB-3426	v2026.10.10	International · English

MozOut International
RLX Holdings (Pty) Ltd. · Group RLX Holdings
Tax no. (SARS) 943 907 6234 · Registration no. (CIPC) 2025/667953/07

https://www.my.mozout.com/knowledgebase/3426

Error listen EADDRINUSE address already in use: finding what holds the port

Node.js & JavaScript Created on **October 10, 2026** 3 min read



The application will not start and the message says `Error: listen EADDRINUSE: address already in use :::3000`. In Python, the version is `OSError: [Errno 98] Address already in use`. **The short answer: another process** is already listening on that port, and only one program can listen on a port at a time. Nearly always it is the same application, still running from an earlier attempt. This article is for a VPS; on shared hosting the case is different (see the end).

Finding who holds the port

1. **See who is listening on the port** (here 3000; swap in yours):

   ```
   sudo ss -tlnp | grep :3000
   ```

   The last column shows the program name and the process number (`pid=1234`). If nothing shows, the port is already free and the error came from an earlier attempt.

2. **Work out what it is.** If it is a `node` or `python` from your application, it is an old instance. If it is something else (`nginx`, a Docker container), the port really is reserved for that service and your application has to use another.

3. **Stop the old instance with care.** If PM2 or systemd manages it, stop it there: `pm2 stop name` or `sudo systemctl stop name`. Only for a stray process, end it with `kill 1234` (the `pid` number). Use `kill -9` only if the first does not work.

4. **Start again** and confirm with the same command as step 1.

The causes behind it

Cause	How to recognise it	What to do
Two instances of the same application	`ss` shows your `node`; `pm2 list` has the application twice, or one runs by hand and another as a service.	Keep one only: PM2 or systemd, not both, and not an open session at the same time.
A service restarting in a loop	The service status flips and the log repeats the error.	Stop the service, sort out the port, start again. See **reading systemctl status**.
A container publishes the same port	`docker ps` shows the published port.	Change the left-hand port in `-p`: **your first container**.
Another legitimate program	`ss` shows a service you know.	Pick another port for your application. Below 1024 only root can listen.
Development with auto-reload	An old process got stuck after an error.	End the old process, as in step 3.

Do not blindly kill the process `ss` shows.
Check what it is first: it may be `nginx` or the database, and stopping those takes other sites down. If in doubt, ask whoever set the server up. On an unmanaged VPS, managing processes is yours: **how far our support goes**.

On shared hosting (cPanel)
you do not choose ports: Passenger connects the domain to the application. If you see this error there, the code is trying to open a second server, or a fixed port with an old server still alive. Use `process.env.PORT || 3000` in `listen` and restart with the button on the application screen. For the rest, see **ports and firewall on a VPS**.

NEED A HAND?

Done all this and the port is still taken? Send us the ss output and the service name.

Open a support ticket →

SEE ALSO

- Ports and firewall on a VPS: why your app is not reachable
- Keeping a Node.js app running on a VPS with PM2
- A service will not start on my VPS: reading systemctl status

MozOut International

RLX Holdings (Pty) Ltd. · Group RLX Holdings

TAX NO. (SARS)	943 907 6234
REGISTRATION NO. (CIPC)	2025/667953/07
REGISTERED OFFICE	1111 Olive Street Saint Louis 63101, Missouri, United States
CONTACT	info@mozout.com · +27 78 823 3325 · www.mozout.com
OPENING HOURS	Monday to Friday - 08 AM to 04 PM Saturday & Sunday - Closed

Online version of this document

RLX-INT-KB-3426 · v2026.10.10 · Generated on 15/11/2026 00:39