Mozout

Gerado a 15/11/2026 02:07

BASE DE CONHECIMENTO · DOCUMENTO PÚBLICO

Error listen EADDRINUSE «address already in use»: descobrir quem tem a porta

Node.js e JavaScript

CÓDIGO	VERSÃO	PAÍS · IDIOMA
RLX-AO-KB-3426	v2026.10.10	Angola · Português

MozOut Angola
RLX Angola, Lda. · Grupo RLX Holdings
NIF 500 260 5724 · Matrícula 26464-25/250724

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

Error listen EADDRINUSE «address already in use»: descobrir quem tem a porta

Node.js e JavaScript Criado em **10 de outubro de 2026** 3 min de leitura



A aplicação não arranca e a mensagem diz `Error: listen EADDRINUSE: address already in use :::3000`. Em Python, a versão é `OSError: [Errno 98] Address already in use`. **A resposta curta:** há **outro processo** a ouvir nessa porta, e só um programa pode ouvir numa porta de cada vez. Quase sempre é a mesma aplicação, ainda a correr de uma tentativa anterior. Este artigo é para um VPS; na hospedagem partilhada o caso é outro (veja o fim).

Descobrir quem tem a porta

1. **Veja quem ouve na porta** (aqui a 3000; troque pela sua):

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

   A última coluna mostra o nome do programa e o número do processo (`pid=1234`). Se não aparecer nada, a porta já está livre e o erro era de uma tentativa anterior.

2. **Perceba o que é.** Se for um `node` ou `python` da sua aplicação, é uma instância antiga. Se for outra coisa (o `nginx`, um contentor Docker), a porta está mesmo reservada para esse serviço e a sua aplicação tem de usar outra.

3. **Pare a instância antiga com cuidado.** Se a gere o PM2 ou o systemd, pare por lá: `pm2 stop nome` ou `sudo systemctl stop nome`. Só se for um processo solto, termine-o com `kill 1234` (o número do `pid`). Use `kill -9` apenas se o primeiro não resultar.

4. **Arranque outra vez** e confirme com o mesmo comando do passo 1.

As causas por trás

Causa	Como se reconhece	O que fazer
Duas instâncias da mesma aplicação	O `ss` mostra o seu `node`; `pm2 list` tem a aplicação duas vezes, ou há uma a correr à mão e outra no serviço.	Fique só com uma: o PM2 ou o systemd, não os dois, e não uma sessão aberta ao mesmo tempo.
Serviço a reiniciar-se em ciclo	O estado do serviço oscila e o registo repete o erro.	Pare o serviço, resolva a porta, volte a arrancar. Veja **ler o systemctl status**.
Um contentor publica a mesma porta	`docker ps` mostra a porta publicada.	Mude a porta da esquerda no `-p` : **o seu primeiro contentor**.
Outro programa legítimo	O `ss` mostra um serviço que conhece.	Escolha outra porta para a sua aplicação. Abaixo de 1024 só o root pode ouvir.
Desenvolvimento com recarga automática	Um processo antigo ficou preso depois de um erro.	Termine o processo antigo, como no passo 3.

Não mate às cegas o processo que o `ss` mostra.
Confirme o que é antes: pode ser o `nginx` ou a base de dados, e parar isso deita abaixo outros sites. Em caso de dúvida, pergunte a quem montou o servidor. Num VPS não gerido, a gestão dos processos é sua: **até onde vai o nosso suporte**.

Na hospedagem partilhada (cPanel)
não escolhe portas: o Passenger liga o domínio à aplicação. Se aí vir este erro, é porque o código tenta abrir um segundo servidor, ou uma porta fixa com um servidor antigo ainda vivo. Use `process.env.PORT || 3000` no `listen` e reinicie pelo botão do ecrã da aplicação. Para o resto, veja **portas e firewall num VPS**.

PRECISA DE AJUDA?

Fez tudo isto e a porta continua ocupada? Envie-nos o resultado do ss e o nome do serviço.

Abrir um pedido de suporte →

VEJA TAMBÉM

- Portas e firewall num VPS: porque a aplicação não é alcançável
- Manter uma aplicação Node.js a correr num VPS com o PM2
- Um serviço não arranca no meu VPS: ler o systemctl status

MozOut Angola
RLX Angola, Lda. · Grupo RLX Holdings

NIF 500 260 5724
MATRÍCULA 26464-25/250724
SEDE Rua Direita do Patriota, Bairro Dangereux
Belas, Luanda, Angola
CONTACTOS info@mozout.com · +244 942 011 191 · www.mozout.com
HORÁRIO 2ª. à 6ª. Feira - 08 à 16 Horas
Sábado & Domingo - Fechado

Versão online deste documento

RLX-AO-KB-3426 · v2026.10.10 · Gerado a 15/11/2026 02:07