opencpn.conf wijzigen op een Android-tablet

Koen Willems · 1 november 2026

Deze notitie beschrijft hoe je de regel `CatalogExpert=1` toevoegt aan `opencpn.conf` van OpenCPN op een Android-tablet, zonder root, vanaf een Mac met MacDroid. Pas met die regel toont OpenCPN op Android in **Opties** → **Plugins** de knoppen om een plugin uit een pakket te importeren en om de catalogus in te stellen; zonder die regel ontbreken ze.

De werkwijze is uitgeprobeerd met OpenCPN 5.14.0 op een Ulefone RugKing Pad Pro met Android 16. Op andere tablets en versies kunnen details verschillen, zoals de rechten van de bestanden.

Het probleem

OpenCPN bewaart zijn instellingen op Android in:

```
/storage/emulated/0/Android/data/org.opencpn.opencpn/files/opencpn.conf
```

Via MacDroid zie je die map wel, en de meeste bestanden erin kun je openen. Juist `opencpn.conf` niet: in Finder krijg je "Operation timed out", en met `adb pull` of `adb shell` "Permission denied". De oorzaak staat in de rechten:

```
-rw-rw----  u0_a265 ext_data_rw  license.txt
-rw-------  u0_a265 ext_data_rw  opencpn.conf
```

OpenCPN schrijft de config zo dat alleen de app zelf hem mag lezen en schrijven (`-rw-------`). De adb-shell (gebruiker `shell`) is wel lid van de groep `ext_data_rw` , maar dat helpt alleen bij bestanden die voor de groep open staan, zoals `license.txt` . Er zijn ook geen andere eenvoudige wegen:

- de tablet is niet geroot (`su` ontbreekt);
- `run-as org.opencpn.opencpn` weigert, omdat de app niet "debuggable" is;
- `adb backup` neemt geen appgegevens mee van apps met targetSdk 31 of hoger (OpenCPN heeft 36).

De opening

Twee dingen maken het toch mogelijk.

1. **De map zelf is schrijfbaar voor de groep.** `files/` heeft de rechten `drwxrws---` met groep `ext_data_rw` . De adb-shell mag daar dus bestanden hernoemen en nieuwe bestanden aanmaken, ook al kan hij `opencpn.conf` niet lezen.
2. **OpenCPN neemt bij het opslaan de rechten van het bestaande bestand over.** wxWidgets (`wxFileConfig` , via `wxTempFile`) schrijft de config naar een tijdelijk bestand, geeft dat de rechten van het bestand dat er al staat, en zet het daarna op zijn plaats. OpenCPN op Android slaat de config op bij `onStop` , dus als de app naar de achtergrond gaat (Home-knop). Dat is te zien in `opencpn.log` , dat wel leesbaar is.

Zet je dus een leeg `opencpn.conf` met de rechten 660 neer terwijl OpenCPN draait, dan schrijft OpenCPN bij de volgende `onStop` zijn volledige config in dat bestand, met de rechten 660. Vanaf dat moment kan de adb-shell het lezen en schrijven.

Wat je nodig hebt

- MacDroid in **ADB-modus**, met de tablet verbonden (USB of draadloze foutopsporing).
- De adb die in MacDroid zit; een eigen adb installeren is niet nodig:

```
A=/Applications/MacDroid.app/Contents/Resources/adb
D=/storage/emulated/0/Android/data/org.opencpn.opencpn/files
$A devices -l
```

`devices` moet de tablet tonen met de status `device`.

Stap voor stap

Een opdracht die over twee regels staat, met een `\` aan het eind van de eerste regel, is één opdracht: kopieer beide regels samen.

1. OpenCPN starten

Start OpenCPN op de tablet en wacht tot de kaart in beeld is. Controleer dat de app draait:

```
$A shell pidof org.opencpn.opencpn
```

Dit geeft een procesnummer. Geen uitvoer betekent dat OpenCPN niet draait: begin dan niet aan stap 2.

2. Het origineel opzij zetten en een leesbaar leeg bestand neerzetten

```
$A shell "cd $D && mv opencpn.conf opencpn.conf.orig && \
  touch opencpn.conf && chmod 660 opencpn.conf && ls -ln opencpn.conf*"
```

Het origineel blijft onaangeroerd bewaard als `opencpn.conf.orig`.

3. OpenCPN naar de achtergrond sturen

Druk op de **Home-knop** van de tablet. Niet afsluiten via het menu en niet wegvegen: OpenCPN schrijft de config bij `onStop`; bij afsluiten doet het dat niet altijd. Wacht tot het bestand gevuld is:

```
$A shell "ls -ln $D/opencpn.conf*"
```

Het nieuwe `opencpn.conf` heeft nu ongeveer dezelfde grootte als het origineel en de rechten `-rw-rw----`.

4. OpenCPN geforceerd stoppen

OpenCPN draait nog op de achtergrond met de oude instellingen in het geheugen. Bij de volgende `onStop` zou het jouw wijziging weer overschrijven. Stop het daarom nu; de config is net weggeschreven, dus er gaat niets verloren:

```
$A shell am force-stop org.opencpn.opencpn
$A shell pidof org.opencpn.opencpn
```

De tweede opdracht hoort niets terug te geven.

5. Ophalen, wijzigen en terugzetten

```
$A pull $D/opencpn.conf opencpn.conf
awk '{print} $0=="[PlugIns]"{print "CatalogExpert=1"}' \
  opencpn.conf > opencpn.conf.new
diff opencpn.conf opencpn.conf.new
$A push opencpn.conf.new $D/opencpn.conf
$A shell "chmod 660 $D/opencpn.conf; \
  grep -n -A1 '^\[PlugIns\]$' $D/opencpn.conf"
```

De `awk`-opdracht voegt de regel toe direct na de kop `[PlugIns]`. Dat is de sectie waarin OpenCPN hem zoekt. Staat er al een regel `CatalogExpert=0` in die sectie, voeg dan niets toe maar verander die `0` in een `1`.

`diff` moet precies één toegevoegde regel laten zien. `adb push` maakt het bestand `-rw-rw-rw-`, vandaar de `chmod 660` achteraf.

6. OpenCPN weer starten

Start OpenCPN. Het leest nu de gewijzigde config.

Terugdraaien

Gaat er iets mis, zet dan het origineel terug terwijl OpenCPN niet draait:

```
$A shell am force-stop org.opencpn.opencpn
$A shell "cd $D && rm opencpn.conf && mv opencpn.conf.orig opencpn.conf"
```

Valkuilen

- **Start OpenCPN nooit met een leeg `opencpn.conf`.** Dan begint OpenCPN met standaardinstellingen en schrijft die weg. Tijdens stap 2 moet OpenCPN dus al draaien. Is OpenCPN toch gestopt voordat het lege bestand er stond, zet dan meteen het origineel terug (zie *Terugdraaien*).
- **Afsluiten is niet hetzelfde als naar de achtergrond gaan.** Gebruik de Home-knop: dan schrijft OpenCPN de config weg bij `onStop`.
- **Bewerk de config nooit terwijl OpenCPN op de achtergrond draait.** OpenCPN overschrijft het bestand bij de volgende `onStop` met wat het in het geheugen heeft. Altijd eerst `am force-stop`.
- **Finder via MacDroid is hier onbetrouwbaar.** Bij het niet-leesbare bestand kan de koppeling vastlopen met "Operation timed out". De adb uit MacDroid geeft duidelijke foutmeldingen.

Een volgende keer

Omdat OpenCPN de rechten van het bestaande bestand overneemt, blijft `opencpn.conf` waarschijnlijk 660. Controleer dat eerst:

```
$A shell "ls -ln $D/opencpn.conf"
```

Staat er `-rw-rw----`, dan zijn stap 2 en 3 niet meer nodig. Dan volstaat: OpenCPN geforceerd stoppen (stap 4), ophalen, wijzigen en terugzetten (stap 5), en OpenCPN weer starten (stap 6).

Is alles in orde, dan kan `opencpn.conf.orig` weg:

```
$A shell rm $D/opencpn.conf.orig
```