

TELE-1b — Build Failure, Rollback OK

2026-05-14 (UTC) · master 31ef276 · **PARTIAL** **Rollback erfolgreich** **Telemetry weiterhin broken**

Status: psycopg2-Fix war im Build erfolgreich (verifiziert 2.9.12 im neuen Image). Aber der Bot-Container ging beim Recreate in einen **Restart-Loop (11 Restarts)** — Ursache: `RUN npm install -g openclaw@latest` wurde durch die Dockerfile-Änderung als Cache-Miss neu ausgeführt → frische openclaw-Version mit Breaking-Changes in der Config-Validation. Per Stop-Regel "Bot startet nicht" → Rollback auf altes Image. Bot ist jetzt wieder im Pre-TELE-1b-Zustand (Telemetry weiterhin still). Operator-GO für TELE-1b-v2 mit openclaw-Pin erforderlich.

1. Verlauf

Step	Aktion	Ergebnis
1 Preflight	Boundary-Snapshot	OK — master 31ef276, Bot PID 195 (Pre-TELE-1b), Worker live, BINANCE_TESTNET=true
2 Build-Recon	Dockerfile + requirements.txt + Image-IDs	OK — psycopg2-binary fehlt, Dockerfile arbeitet requirements.txt nicht ab. Alt-Image-ID: 0315dabef31f
3 Durable Fix	requirements.txt: <code>psycopg2-binary>=2.9.0 → ==2.9.12</code> ; Dockerfile + COPY trading/requirements.txt + RUN <code>pip3 install -r</code>	OK — minimaler Diff (+8 -1)
4 Build	<code>docker compose -p steve-tradingbot build clawbot</code>	OK — neues Image 267f0b14..., <code>python3 -c "import psycopg2"</code> → 2.9.12 (dt dec pq3 ext lo64)
5 Recreate	Bot only, <code>--no-deps --force-recreate</code>	FAIL — Bot-Container Restart-Loop (11 Restarts in ~5 min)
Rollback	<code>docker tag 0315dabef31f steve-tradingbot-clawbot:latest</code> + Bot recreate	OK — Bot host-PID 2974628, läuft, Bot main.py PID 102, Worker unverändert

2. Root Cause des Build-Failures

```
Gateway failed to start: Error: Invalid config at /home/node/.openclaw/openclaw.json.  
channels.telegram: invalid config: must NOT have additional properties  
channels.telegram.streaming: invalid config: must be object  
plugins.slots.memory: plugin not found: memory-lancedb
```

Im neuen Bot-Image wurde durch Cache-Invalidation (mein Edit kam vor der `RUN npm install`-Zeile) eine frische openclaw-Version installiert (`npm install -g openclaw@latest` in Zeile 39 des Dockerfile). Diese neue Version hat striktere Config-Validation, die mit der gemounteten `openclaw.json` nicht mehr kompatibel ist.

Alte funktionierende openclaw-Version (aus dem rollback-betroffenen Bot-Container): `OpenClaw 2026.4.24 (cbcfd6)`.

3. Was wir gelernt haben

Erkenntnis	Implikation
psycopg2-Fix selbst ist korrekt — Build und Import-Test grün	Edits an Dockerfile + requirements.txt bleiben gültig
<code>npm install -g openclaw@latest</code> ist eine Tretmine	JEDER Dockerfile-Edit vor dieser Zeile löst Cache-Miss aus → frische openclaw
Layer-Reorder oder Version-Pin nötig	TELE-1b-v2: <code>openclaw@2026.4.24</code> pinnen
Rollback per Image-Tag-Switch + Recreate funktioniert sauber	Stop-Regel-Pfad ist tragfähig — Bot/Worker/GUI/Boundary unverändert

4. Post-Rollback Boundary-Snapshot

Item	Wert
Bot Container Host-PID	2974628 (running)
Bot main.py PID (in-container)	102 (Watchdog-Spawn)
Bot Image	0315dabef31f (alt, vor TELE-1b — gleicher wie Pre-Phase)
Worker Host-PID	2658058 — unverändert
Worker Image	57ae9c0179f8 (clawbot-runtime, unverändert)
Worker Heartbeat age	27 s
BINANCE_TESTNET	true
cmd 13 / mp / history	cancelled / 0 / 0
Telemetry-Streams	weiterhin still (decision_logs/position_snapshots/regime_logs ≥4h stale)

5. Aktueller Repo-Stand

Datei	Status
Dockerfile	modifiziert (+COPY+RUN für requirements.txt) — uncommitted
trading/requirements.txt	modifiziert (psycopg2-binary==2.9.12 gepinnt) — uncommitted
master HEAD	31ef276 unverändert
Bot-Image-Tag	auf alte sha zurückgesetzt — funktional, aber nicht reproduzierbar aus aktueller Dockerfile

Wichtig: Die Edits liegen im Working-Tree. Bei `docker compose build` würden sie aktuell wieder das fehlerhafte Image bauen. Eintrag im git nur bei TELE-1b-v2-Erfolg.

6. TELE-1b-v2 Empfehlung

Minimal-invasiver Fix für die openclaw-Cache-Invalidation:

Option	Bewertung
V2-A — openclaw pinnen: <code>npm install -g openclaw@2026.4.24</code> statt <code>@latest</code>	empfohlen — minimaler Diff, deterministisch, kein Cache-Probleme zukünftig
V2-B — Edit-Reihenfolge: mein Edit nach <code>RUN npm install -g openclaw@latest</code> verschieben	funktioniert, aber Tretmine bleibt für die nächste Änderung
V2-C — separater Image-Layer für <code>psycopg2</code> ganz am Ende	OK, aber ändert Architektur

Konkreter Edit für TELE-1b-v2 (Vorschau, noch nicht angewendet)

```
# Dockerfile Zeile 39 (statt:)
# RUN npm install -g openclaw@latest
# wird zu:
RUN npm install -g openclaw@2026.4.24
```

+ behaltene Edits aus TELE-1b v1 (Dockerfile `COPY trading/requirements.txt` + `RUN pip3 install -r requirements.txt psycopg2-binary==2.9.12`).

7. Backlog (unverändert)

Item	Zweck	Priorität
TELE-1b-v2	openclaw-Pin + psycopg2-Edit retain + Re-Build/Recreate	P0 — direkt nach Operator-GO
TELE-2	db_emitter darf Import-Fehler nicht silent schlucken; periodischer Alarm	P1
IMAGE-1	docker commit-Runtime-Images verbieten (durable rule)	P1
IMAGE-2	Bot/Worker Runtime-Base vereinheitlichen	P1
IMAGE-3	Build-Smoke-Test für Pflicht-Imports (psycopg2, ccxt, etc.)	P1
IMAGE-4 (NEU)	npm-Dependency-Pin im Dockerfile (kein <code>@latest</code>)	P0
POS-CAP-1	42 Positionen trotz <code>Cap=5</code>	P0/P1
DATA-LINK-1	trade_logs↔decision_logs Wiring	P1
LABEL-1	stop_loss Subtypen trennen	P1

8. GO/NO-GO TELE-1b-v2

Operator-GO empfohlen für TELE-1b-v2 mit openclaw-Pin `@2026.4.24`. Erwartete Dauer: 5 min Build + 30s Recreate + 2 min Verifikation. Boundary-Profil identisch zu v1.

9. Boundaries — alle eingehalten

- 0x Strategie-Code
- 0x Trading-Parameter
- 0x `runtime_config/managed_state/baseline`
- 0x `docker cp`
- 0x Mainnet
- 0x Push
- 0x Secret-Output
- 0x DB-Migration
- 0x kuenstliche Trades/Orders erzeugt
- 0x `docker commit` als Fix
- Worker während Build und Recreate **unangetastet**

