SEC-1c-3b Plan-Review — Bot+Worker DB-Cutover

Projekt: Steve-TradingBot · Phase: SEC-1c-3b · Author: claude-opus-4-7[1m]
Generated: 2026-05-14 07:05 UTC · master HEAD: `d351a3a`
Status: **NO CODE** Plan-Review only — Operator-GO erforderlich vor jedem Container-Recreate.
Empfehlung: **Compose-env-Substitution via Root-.env + docker compose up -d --force-recreate clawbot clawbot-worker**

0 — Executive Summary

Ziel: Bot (`clawbot`) und Worker (`clawbot-worker`) von der Legacy-Rolle `tradingbot_gui` (SUPERUSER) auf die Runtime-Rolle `tradingbot_gui_app` umstellen. Defense-Effekt: bei Bot/Worker-Compromise (Market-data-RCE, dependency-injection, etc.) kann der Angreifer maximal Daten manipulieren, nicht Schema zerstoeren oder Rollen escalieren — analog zu SEC-1c-3a aber für die zwei Bot-Container.

Vorbedingung: SEC-1c-3a closed (GUI bereits umgestellt, 3-Rollen-Modell live).

Config-Source-of-Truth: Beide Container lesen ihre DB-Credentials aus der **compose-env-Section** in `/projekte/Steve-TradingBot/docker-compose.yml` . Die Werte werden via `${GUI_DB_*:-default}` -Substitution aus `/projekte/Steve-TradingBot/.env` gefüllt. Aktuell sind dort nur `GUI_DB_DSN` + `GUI_DB_CONNECT_TIMEOUT` gesetzt; `GUI_DB_USER/PASSWORD/HOST/PORT/NAME` fallen auf compose-Defaults zurück (= `tradingbot_gui:tradingbot_gui`).

Cutover: Root- `.env` updaten + `docker compose up -d --force-recreate clawbot clawbot-worker` . Bot/Worker stop+recreate ~30s pro Container; Mainnet-Block + TESTNET-Sicherung unberührt; managed_proposals = 0 (kein In-Flight-State der gefährdet wäre).

---

1 — Live-State (Pre-Cutover-Recon, 2026-05-14 07:05 UTC)

Item	Wert	Status
master HEAD	`d351a3a`	✓
git status	clean (kein uncommitted-Repo-Touch)	✓
Bot in-container PID	363 (`python3 main.py --paper`)	✓
Worker in-container PID	1 (`python3 -m trading.command_worker`)	✓
Worker heartbeat (command_audit_log)	command-bus aktiv (FOR UPDATE SKIP LOCKED Pattern in code, kontinuierliche Polls)	✓
BINANCE_TESTNET	`true` (verifiziert via Bot in-container .env)	✓
cmd 13 status	`cancelled`	✓
managed_proposals rows	0	✓
managed_assets_history rows	0	✓
GUI current_user	**`tradingbot_gui_app`** (SEC-1c-3a closed)	✓
Bot/Worker DB-Connection (compose-env)	**`tradingbot_gui` (SUPERUSER)** — muss umgestellt werden	🔴 **SEC-1c-3b target**
decision_logs Schreibrate (letzte 1h)	**474 INSERTs** — Bot+Worker schreiben aktiv	✓ (Heartbeat-Indikator)
bot_statuses max_id	9751	✓
Tracebacks in Bot-Log (letzte 50 Zeilen)	0	✓
SECRET-READ-ONCE-Files	`/root/.secrets/sec-1c-3a-app.SECRET-READ-ONCE` (mode 600, 33 bytes) + migrator-equiv. **Operator-Aufgabe vor Cutover**: lesen + shreddern (NICHT shreddern WAEHREND Cutover, Bot/Worker brauchen App-PW noch)	⏳ pending

---

2 — Konfigurations-Source-of-Truth

2.1 docker-compose.yml env-Sections (Auszug)

Service	relevante Zeilen
`clawbot` (Line 32-33)	nur `GUI_DB_DSN=${GUI_DB_DSN:-}` + `GUI_DB_CONNECT_TIMEOUT=${GUI_DB_CONNECT_TIMEOUT:-2}`. Bot liest DSN-URL-Form.
`clawbot-worker` (Line 123-129)	vollständig: `GUI_DB_DSN + HOST + PORT + NAME + USER + PASSWORD + CONNECT_TIMEOUT`. Worker kann sowohl DSN als auch einzeln lesen.

2.2 Substituiert aus Root-`.env`

Variable	aktueller Stand in `/projekte/Steve-TradingBot/.env`	Soll nach 1c-3b
`GUI_DB_DSN`	vorhanden, mit altem User `tradingbot_gui`	aktualisiert auf `postgres://tradingbot_gui_app:<app-PW>@steve-tradingbot-gui-db:5432/tradingbot_gui`
`GUI_DB_USER`	nicht gesetzt (compose-Default `tradingbot_gui`)	`tradingbot_gui_app`
`GUI_DB_PASSWORD`	nicht gesetzt (compose-Default `tradingbot_gui`)	`<app-PW>` aus SECRET-READ-ONCE (gelesen, NICHT geechoed; via heredoc + sed-with-temp-file-pattern)
`GUI_DB_HOST/PORT/NAME/CONNECT_TIMEOUT`	nicht gesetzt — compose-Defaults `gui-db / 5432 / tradingbot_gui / 2`	unverändert (compose-Defaults sind korrekt)

2.3 Bot/Worker DB-Connect-Pattern

- Bot: `trading/db_emitter.py` Zeile 185 verwendet `INSERT INTO {table} (cols) VALUES (placeholders)` — psycopg2 mit Standard-DSN-URL aus `GUI_DB_DSN`. KEIN CREATE/ALTER/DROP/TRUNCATE im Source.
- Worker: `trading/command_worker.py` ist Command-Bus-Poller. Verwendet `FOR UPDATE SKIP LOCKED` auf `commands`-Table und `UPDATE commands SET ...`. Zwei separate Connect-Bibliotheken (sqlalchemy via DSN ODER psycopg2 direkt mit User/PW — je nach Code-Pfad). Alles im DML-Bereich, kein DDL.

→ Beide Workloads sind DML-only und kompatibel mit `tradingbot_gui_app`-Privs.

---

3 — Cutover-Plan (atomic Block)

#	Step	Risiko	Zeit
3b.0	Pre-flight Snapshot (PIDs, decision_logs-Count, command_audit-Latest, GUI current_user, managed_proposals=0, cmd 13 cancelled, BINANCE_TESTNET=true). Operator-Aufgabe: SECRET-READ-ONCE shred KOMMT NACH 1c-3b (Bot/Worker brauchen PW noch zum Restart)	0	3 min
3b.1	Backup: `/projekte/Steve-TradingBot/.env` → `/root/sec-1c-3b-backup-<ts>/env.pre` mode 600	0	1 min
3b.2	Snapshot Bot/Worker container-config (env-keys, mounts, networks) — bestehende compose-config bleibt unverändert; nur env-Substitution aus .env greift	0	1 min
3b.3	`/projekte/Steve-TradingBot/.env` updaten: `GUI_DB_DSN` mit neuem User-PW + `GUI_DB_USER=tradingbot_gui_app` + `GUI_DB_PASSWORD=<app-PW>` . **PW-Wert via heredoc/sed-with-temp-Pattern, NICHT geechoed.**	LOW	2 min
3b.4	`docker compose config --no-interpolate` (oder Filter durch grep -v key|token|password|secret) — Sanity-Check, dass compose die neuen Werte sieht ohne sie zu leaken	0	1 min
3b.5	**Bot-Container recreate**: `docker compose up -d --force-recreate clawbot` — ~10s stop + ~20s recreate. PID wird neu (≠ 363).	HIGH (Restart-Window-Start)	2 min
3b.6	Bot Smoke: `docker exec clawbot tr "\0" "\n" < /proc/<newPID>/environ | grep GUI_DB_USER` → `tradingbot_gui_app` . Bot-Logs (`tail -50 logs/bot_stdout.log`) frei von Tracebacks, scan-cycle aktiv	HIGH	5 min
3b.7	**Worker-Container recreate**: `docker compose up -d --force-recreate clawbot-worker`	HIGH (Restart-Window-2)	2 min
3b.8	Worker Smoke: `command_worker.pid` + heartbeat-write in command_audit_log; `SELECT usename FROM pg_stat_activity WHERE application_name LIKE '%worker%'` → `tradingbot_gui_app`	HIGH	5 min
3b.9	Permission-Tests Bot/Worker: Bot kann decision_logs INSERT (= aktive Schreibrate >0 in nächsten 5 min) — aber CREATE/DROP/ALTER/TRUNCATE/CREATE-ROLE muss permission-denied geben (Negative-Tests via `docker exec clawbot psql ...`)	MEDIUM	5 min
3b.10	Boundary-Recheck: cmd 13 noch cancelled, managed_proposals=0, BINANCE_TESTNET=true	0	2 min
3b.11	(BACKLOG-relevant) `SELECT key FROM pg_stat_activity WHERE usename='tradingbot_gui'` — sollte ab jetzt LEER sein (kein App nutzt SUPERUSER). DBA-Test-Connect via SSH+psql bleibt erlaubt	0	1 min
3b.12	Closure-Pin + STOP-Report	0	5 min

Σ SEC-1c-3b: ~35 min, davon ~30s Bot-Container-Recreate + ~30s Worker-Container-Recreate (sequenziell, nicht parallel).

3.1 Wichtige Subtilitäten

- **Bot-Watchdog-Verhalten**: `bot_watchdog.sh */5` spawnt Bot-Process nur wenn Container läuft. Während `docker compose up -d --force-recreate` ist Container kurz down (entrypoint läuft neu); Watchdog feuert nicht in den down-Phasen.
- **command_worker**: nutzt `FOR UPDATE SKIP LOCKED` — multiple parallel Worker safe. Restart ist atomic; aktuelle pending Commands (falls vorhanden) werden vom nächsten Poll re-claimed.
- **Reihenfolge Bot-vor-Worker**: Bot in-flight scan-cycles können INSERTs queuen. Worker arbeitet Queue ab. Reihenfolge: **Bot zuerst** (sein Schreib-Output ist akkumulierter), Worker als Zweiter (Queue-Drain). Beides rückgängig-sicher.
- **NICHT mit GUI-Container parallel restart**: GUI war SEC-1c-3a behandelt, läuft schon als App-User. Kein Anlass GUI-Container in 1c-3b zu touchen.

---

4 — Permission-Tests (Pflicht in 3b.9)

Positive Tests (Bot/Worker via App-User — müssen klappen)

```
# Bot kann decision_logs INSERT:
docker exec clawbot sh -lc 'psql "$GUI_DB_DSN" -c "INSERT INTO decision_logs (decision_id, decided_at,
symbol, action) VALUES (\\'sec-1c-3b-canary\\', NOW(), \\'TEST/USDT\\', \\'sec-test\\') RETURNING id;"'
# Dann cleanup:
docker exec clawbot sh -lc 'psql "$GUI_DB_DSN" -c "DELETE FROM decision_logs WHERE decision_id=\\'sec-1c-
3b-canary\\';"'

# Worker kann commands UPDATE (DML, kein DDL):
docker exec clawbot-worker sh -lc 'psql "$GUI_DB_DSN" -c "SELECT id, status FROM commands ORDER BY id DESC
LIMIT 3;"'
docker exec clawbot-worker sh -lc 'psql "$GUI_DB_DSN" -c "SELECT current_user;"'  # → tradingbot_gui_app
```

Negative Tests (Bot/Worker via App-User — müssen alle permission-denied geben)

```
docker exec clawbot sh -lc 'psql "$GUI_DB_DSN" -c "CREATE TABLE _bot_leak (id int);"'    # → permission
denied
docker exec clawbot sh -lc 'psql "$GUI_DB_DSN" -c "DROP TABLE decision_logs;"'           # → must be owner
docker exec clawbot sh -lc 'psql "$GUI_DB_DSN" -c "TRUNCATE bot_statuses;"'              # → permission
denied
docker exec clawbot-worker sh -lc 'psql "$GUI_DB_DSN" -c "CREATE ROLE evil;"'            # → permission
denied to create role
docker exec clawbot-worker sh -lc 'psql "$GUI_DB_DSN" -c "ALTER SYSTEM SET fsync=off;"'  # → permission
denied
```

Heartbeat-Verifikation (in nächsten 5 Min)

```
# Decision-Logs Schreibrate sollte sich fortsetzen:
docker exec steve-tradingbot-gui-db psql -U tradingbot_gui -d tradingbot_gui -tAc "
   SELECT 'decision_logs_last_5min='||COUNT(*) FROM decision_logs WHERE decided_at > NOW() - INTERVAL '5
minutes';
   SELECT 'pg_stat_app_users='||string_agg(DISTINCT usename, ',') FROM pg_stat_activity WHERE
datname='tradingbot_gui';
"
# Erwartet: decision_logs_last_5min > 0 ; usernames enthält tradingbot_gui_app (NICHT mehr tradingbot_gui
für App-Connections)
```

---

5 — Stop-Regeln (Pflicht)

- **STOP**: jeder Secret-Wert (App-PW, Migrator-PW) erscheint im Output / Log / Memory / Commit / PDF → sofortiges Re-Rotate
- **STOP**: Bot-Container startet nach `docker compose up` nicht (Exit-Code != 0, kein Process auf `main.py`) → rollback aus Backup
- **STOP**: Bot-Container läuft, aber Logs zeigen `FATAL: password authentication failed` → .env-Wert falsch, rollback + Re-Try mit verifiziertem PW
- **STOP**: Worker startet nicht oder kein `command_worker.pid` in `/var/log/...` → rollback
- **STOP**: Worker heartbeat fehlt 5+ Minuten in `command_audit_log` nach Restart → rollback
- **STOP**: Positive-Test (Bot/Worker INSERT auf erlaubter Tabelle) gibt permission-denied → GRANT-Problem, rollback
- **STOP**: Negative-Test passt (Bot/Worker kann CREATE/DROP/TRUNCATE) → Defense kaputt, rollback
- **STOP**: cmd 13 wechselt von `cancelled` zu etwas anderem → Boundary-Verletzung
- **STOP**: BINANCE_TESTNET nicht mehr `true` → sofortiger Halt
- **STOP**: managed_proposals oder _history rows > 0 nach Cutover (Worker schreibt unerwartet) → investigation
- **STOP**: GUI-Container vorzeitig restarted oder broken — nicht im Scope von 1c-3b

---

6 — Rollback-Pfad

Failure-Zeitpunkt	Rollback
nach 3b.3 (.env-Edit, vor Container-Restart)	`cp /root/sec-1c-3b-backup-<ts>/env.pre /projekte/Steve-TradingBot/.env` — kein Container-Restart nötig, da compose-env noch alt im laufenden Container
nach 3b.5 (Bot recreated, Bot failt)	Restore .env + `docker compose up -d --force-recreate clawbot` — Bot connectet wieder mit SUPERUSER
nach 3b.7 (Worker recreated, Worker failt)	analog — Restore .env + `--force-recreate clawbot-worker`
komplettes Rollback	.env-Restore + 2× `--force-recreate` — < 5 min, kein Datenverlust (alle Bot/Worker-Writes idempotent)

---

7 — Boundaries (Plan-Review-End)

master HEAD	`d351a3a` unverändert
git status	clean
Bot in-container PID	363 unverändert
Worker in-container PID	1 unverändert
BINANCE_TESTNET	true
managed_proposals / history	0 / 0
cmd 13	cancelled
GUI current_user	`tradingbot_gui_app` ✓
Bot/Worker DB-User	`tradingbot_gui` (SUPERUSER) — geplant zu ändern in 1c-3b
Mainnet	0
Push	0 (Plan-Review)
DB-Migration	0 (kein Schema-Touch)
docker cp	0
managed_state / runtime_config / baseline-Touch	0
Secret-Werte im Output	0 ✓

---

8 — GO/NO-GO Einschätzung

GO-Voraussetzungen (alle erfüllt)

- ✓ SEC-1c-3a closed (Rollen vorhanden + GRANT + ALTER DEFAULT PRIVILEGES)
- ✓ `tradingbot_gui_app` + `tradingbot_gui_migrator` mit App-PW + Mig-PW in SECRET-READ-ONCE-Files
- ✓ Source-of-Truth identifiziert: `/projekte/Steve-TradingBot/.env` via compose-Substitution
- ✓ Bot/Worker Code-Pfad DML-only (kein DDL); kompatibel mit App-User-Privs
- ✓ Aktive Schreibrate gesund (decision_logs +474/h) — Heartbeat-Mess-Basis bekannt
- ✓ cmd 13 cancelled, managed_proposals=0 (kein In-Flight-State)
- ✓ Live-State sauber: 0 Tracebacks, BINANCE_TESTNET=true

Pflicht-Pre-1c-3b Operator-Action

1. **SECRET-READ-ONCE Files NOCH NICHT shreddern** — Bot/Worker brauchen `tradingbot_gui_app`-PW noch zum Restart. Shredden NACH erfolgreichem Cutover
2. GUI Browser-Smoke-Test (war SEC-1c-3a Closure Punkt 2) — falls noch nicht geschehen, kurz `https://gui.gewerbespeicher-rechner.de/admin/login` aufrufen und einen Read-Endpoint bestätigen

Empfehlung

GO SEC-1c-3b atomic 13-Step-Block mit explizitem Restart-Window-Approval. Aufwand ~35 min. Bei jedem Stop-Trigger sofortiger Rollback (< 5 min).

Alternativ: weiter aufsplitten in 3b-1 (Bot only) + 3b-2 (Worker only), falls Operator nur einen Container zur Zeit recreaten will. Weniger Risiko, aber unschön weil .env-Wechsel atomar; bevorzugte Variante: **beide zusammen in einem Block**.

Status

PLAN-REVIEW Plan fertig. Warte auf:

- (empfehlung-anchor) Bestätigung Bot+Worker-Restart-Window OK
- (optional) Operator-Pin: Split in 3b-1 + 3b-2 vs atomic 13-Step
- GO Aussage

Kein Code geschrieben. Keine Container/DB-/`.env`-Änderung. Pure analysis only.