SEC-1c-0 Exposure-Audit — Public-Surface Inventory

Projekt: Steve-TradingBot · Phase: SEC-1c-0 · Author: claude-opus-4-7[1m]
Generated: 2026-05-13 15:19 UTC · master HEAD: `2fbc7b5` (SEC-1b-6 closed)
Host: `h2991365.stratoserver.net` (Strato Cloud) · OS: Ubuntu 20.04.6 LTS
Status: **READ-ONLY AUDIT** NO CODE, NO CHANGES. Fix-Plan im Anschluss — wartet auf Operator-GO.

Executive Summary

Das System hat **kein lokales Firewall-Layer** (kein `iptables`, kein `nftables`, kein `ufw` installiert). Alle 0.0.0.0-Bindings sind damit aus dem Internet erreichbar, sofern Strato kein vorgeschaltetes Cloud-Firewall stellt (Operator-Prüfpflicht im Strato-Dashboard).

4 Critical / High Findings, sofortige Fix-Priorität:

1. **CRITICAL** — `netdata` v1.19.0 unauthenticated dashboard auf `0.0.0.0:9080` (HTTP 200 extern); leakt Hostname, alle Prozesse, Container, Memory/CPU/Disk. Version aus 2020 mit bekannten CVEs.
2. **CRITICAL** — Bot-Trading-Dashboard auf `0.0.0.0:8050` ohne Authentifizierung (HTTP 200 extern); leakt Bot-Status, Positions, Logs.
3. **HIGH** — `cupsd` Snap-Service auf `0.0.0.0:631`; aktuell rejectet das externe Auth-Layer mit 403, aber binding ist falsch und CUPS hat aktive CVE-Klasse (CVE-2024-47175 et al.). Server hat keinen lokalen Drucker-Use-Case.
4. **HIGH** — **kein lokales Firewall-Layer**. Defense-in-Depth fehlt komplett auf Host-Ebene.

Steve-TradingBot-spezifische Surfaces sind sauber gebunden (GUI 8090 localhost, GUI-DB intern, Portainer 9443 localhost ab SEC-1b-0, Cockpit gelöscht ab SEC-1b-0). Die kritischen Findings sind **host-level / sibling-projects**, nicht Bot/GUI selbst.

---

1 — Methode

- `ss -tlnp / ss -ulnp`: alle host-listening TCP/UDP-Ports mit PIDs
- `docker port <each-container>`: alle docker-proxy-Mappings
- `nginx sites-enabled/*`: alle reverse-proxy-Targets
- `curl` Sondierung: 127.0.0.1 und `81.169.213.37` (extern) für jeden Port (Auth-Verhalten + Reachability)
- `systemctl list-units`: alle aktiven Services
- `fail2ban-client status`: Defensive-Layer-Stand
- `cat /proc/<pid>/cmdline`: Prozess-Identifikation
- Backup-Permissions: `stat` auf `/srv/shares/backups/`

Keine Modifikation: kein `nft` / `iptables`-Touch, kein `systemctl stop/start`, kein `docker stop/rm`, kein `compose`-Edit. Read-only.

---

2 — Vollständige Port-Matrix (Host-Listener)

Port	Bind	Service	PID	Extern	Auth	Klassifikation
22/tcp	0.0.0.0 + [::]	sshd (OpenSSH)	737	reachable	key-only (`PermitRootLogin prohibit-password`), fail2ban 320 banned	**OK**
80/tcp	0.0.0.0	nginx	860/4010xxx	reachable	n/a (force-redirect zur HTTPS-Variante + `/shares/` public)	**OK**
443/tcp	0.0.0.0	nginx (dashboard-ssl)	860/4010xxx	reachable	self-signed cert, proxy zu Bot-Dashboard :8050 (zZt 502)	**SEC-1c-1 zu fixen (HTTPS + LE)**
631/tcp	0.0.0.0 + [::]	cupsd (snap.cups.cupsd)	1079	extern HTTP 403	CUPS-eigenes Allow-Layer	**HIGH — auf localhost-only setzen oder snap deaktivieren**
587/tcp	127.0.0.1	sendmail-mta	888	not reachable extern	local-only	**OK**
25/tcp	127.0.0.1	sendmail-mta	888	not reachable extern	local-only	**OK**
53/udp+tcp	127.0.0.53	systemd-resolved	411	not reachable	local-only stub	**OK**
3456/tcp	127.0.0.1	claude-max-api (node)	3430407	not reachable (000)	local-only API-proxy	**OK**
8000/tcp	127.0.0.1	docker-proxy → portainer	2781230	not reachable	localhost-only	**OK** (SEC-1b-0)
8080/tcp	0.0.0.0	docker-proxy → kw-website-caddy	2047077	reachable	nicht Steve-TradingBot-Projekt	**out-of-scope (separates caddy-stack-Projekt)**
8088/tcp	0.0.0.0	nginx /shares (HTTP)	nginx	reachable	autoindex public, `/shares/backups/` 403 dank dir-mode 700	**OK** — aber autoindex der Top-Level `/shares/` ist beabsichtigt für PDFs/Reports
8443/tcp	0.0.0.0	nginx (wp-test-ssl)	nginx	502 zurück	tot	**orphan nginx-config — aufräumen**
8050/tcp	0.0.0.0	docker-proxy → clawbot dashboard.py	566966 (proxy) / clawbot PID 8	HTTP 200 extern	**keine Authentifizierung**	**CRITICAL — sofort localhost-binden + nginx-Basic-Auth ODER löschen**
8090/tcp	127.0.0.1	docker-proxy → GUI :8000 (Filament)	3651547	not reachable extern	Filament-Login + MFA (SEC-1b-6)	**OK** — aktuell nur lokal, SEC-1c-1 wird via nginx publik
8125/tcp+udp	127.0.0.1 + [::1]	netdata (StatsD)	1445792	not reachable	local-only	**OK**
8888/tcp	0.0.0.0	docker-proxy → kw-website-caddy	2047062	reachable	nicht Steve-TradingBot	**out-of-scope (caddy-stack)**
9443/tcp	127.0.0.1	docker-proxy → portainer	2781214	not reachable extern	Portainer-Login	**OK** (SEC-1b-0)
9080/tcp	0.0.0.0	netdata v1.19.0	1445792	HTTP 200 extern	**anonymer Zugriff**, vollständige System-Telemetry leakt	**CRITICAL — bind auf localhost ODER Auth-Layer ODER Löschung**
18789/tcp	127.0.0.1 + [::1]	openclaw (host)	1531226	not reachable	local-only	**OK**
18791/tcp	127.0.0.1	docker-proxy → clawbot internal openclaw	566950	not reachable	local-only	**OK**
27017/tcp	127.0.0.1	mongod (system)	692	not reachable	local-only	**OK**
27018/tcp	127.0.0.1	docker-proxy → militaria-mongo	1596	not reachable	local-only	**OK**
38381/tcp	127.0.0.1	openclaw (host)	1531226	not reachable	local-only	**OK**

UDP-Listener

Port	Bind	Service	Klassifikation
53/udp	127.0.0.53	systemd-resolved	**OK**
8125/udp	127.0.0.1 + [::1]	netdata StatsD	**OK**
44036/45240/45286/48535/50336/50957/57691/59038/udp	*	eshop-poller-bot.jar (Java PID 418)	**ephemeral outgoing — sibling project, not a listener concern**

---

3 — Steve-TradingBot-Container-Mappings (compose)

Container	Mapping	Bewertung
`steve-tradingbot-gui`	`127.0.0.1:8090 -> container:8000`	**OK** — localhost-only, SEC-1c-1 wird via nginx exposen
`steve-tradingbot-gui-db`	kein Host-Mapping (intern `gui-db:5432`)	**OK** — nur container-network
`clawbot-worker`	kein Mapping	**OK**
`clawbot`	`0.0.0.0:8050 -> container:8050` (dashboard) + `127.0.0.1:18791 -> container:18789`	**CRITICAL** — 8050 ist public ohne Auth; 18791 OK

---

4 — Kritische Findings im Detail

F1 — CRITICAL netdata v1.19.0 auf `0.0.0.0:9080`

Symptom	`curl http://81.169.213.37:9080` → HTTP 200 mit vollständigem netdata-Dashboard
Version	`v1.19.0` (Release 2020-04) — aktueller Stable ist 1.4x
Leakt	Hostname (`h2991365.stratoserver.net`), Kernel-Version, alle laufenden Prozesse (inkl. `python3 main.py --paper`), alle Container, Memory/CPU/Disk-Metriken, Netzwerk-Traffic-Patterns
CVE-Klasse	v1.19 hat dokumentierte Issues (XSS, Info-Disclosure); aktuelle Versionen haben Auth-Layer + Bind-default `127.0.0.1`
Reachability extern	verifiziert: HTTP 200, kein 403, kein Login-Prompt
Fix-Optionen	(a) `/etc/netdata/netdata.conf` `bind to = 127.0.0.1` + restart — **schnellst** (b) hinter nginx mit Basic-Auth + Update auf 1.4x (c) deinstallieren (Operator nutzt netdata aktuell wofür?)
Empfehlung	**(a) sofort**, dann (b) oder (c) als SEC-1c-0-Sub-Phase

F2 — CRITICAL Bot-Dashboard auf `0.0.0.0:8050`

Symptom	`curl http://81.169.213.37:8050/` → HTTP 200 (Trading Bot Dashboard); `/api/` → HTTP 200
Quelle	compose-Mapping `"0.0.0.0:8050:8050"` in `docker-compose.yml`; Bot-Container PID 8 = `dashboard.py`
Leakt	Bot-Status (running/idle), Strategy-Stats, evtl. Position-Snapshots, evtl. PnL, evtl. Trade-History
Auth	keine
Fix-Optionen	(a) compose-Mapping ändern auf `"127.0.0.1:8050:8050"` + Container neu starten — **Bot-Restart-Pflicht**; nicht Plan-Review-konform ohne Operator-GO (b) nginx-Reverse-Proxy mit Basic-Auth davor (kein Bot-Touch) (c) Dashboard ausschalten wenn nicht aktiv genutzt — Operator-Decision
Empfehlung	(b) **als kurzfristige Mitigation** ohne Bot-Restart: nginx auf `81.169.213.37:8050` → localhost mit Basic-Auth; danach in eigenem Bot-Maintenance-Window (b) durch (a) ersetzen. Alternativ Operator entscheidet (c).

F3 — HIGH cupsd auf `0.0.0.0:631`

Symptom	extern HTTP 403 (CUPS-eigenes Auth-Layer blockt), aber binding ist auf 0.0.0.0 — Defense-in-Depth fehlt
Quelle	Snap-Package `snap.cups.cupsd + snap.cups.cups-browsed`
Risiko	CUPS hat aktive CVE-Klasse 2024-47175ff (RCE durch IPP-Pakete). 403 schliesst HTTP-Browser-Zugriff aus, schliesst aber nicht IPP-Protokoll auf 631 aus.
Use-Case auf Server	Server ohne lokalen Drucker — CUPS hat keinen ersichtlichen Nutzen
Fix-Optionen	(a) `snap stop cups` + `snap remove cups` — sauberste Variante (b) `Listen 127.0.0.1:631` in CUPS-config
Empfehlung	**(a) deinstallieren** — kein Use-Case auf Server

F4 — HIGH kein lokales Firewall-Layer

Befund	weder `iptables` noch `nftables` noch `ufw` installiert/aktiv
Konsequenz	jeder 0.0.0.0-Bind ist **direkt aus dem Internet** erreichbar (vorbehaltlich Strato-Cloud-Firewall)
Cloud-Provider	Strato (`h2991365.stratoserver.net`) — Strato bietet je nach Produkt eine separate Firewall, die im Strato-Dashboard konfiguriert wird. **Operator-Prüfpflicht**.
Fix-Optionen	(a) `ufw` installieren + Default-Deny-Inbound + explicit Allow für 22/80/443 (8088 optional) (b) `nftables` mit eigenem Regelsatz (c) Strato-Cloud-Firewall im Strato-Dashboard pflegen
Empfehlung	**(a) ufw als Quick-Win** + (c) Strato-Cloud-Firewall als zweite Layer (defense-in-depth)

---

5 — Medium / Sibling-Project Findings

Item	Befund	Bewertung
`caddy-stack :8080 + :8888`	kw-website-caddy bindet 0.0.0.0; separates Projekt `/srv/caddy-stack/`	**out-of-scope SEC-1c** — Operator-Notiz an caddy-stack-Pflege; nicht im Steve-TradingBot-Risiko
`wp-test-ssl :8443`	nginx-config aktiv, proxy_pass dead (502)	**orphan** — nginx-config-Eintrag `wp-test-ssl` aufräumen wenn nicht mehr benötigt
`eshop-poller-bot.jar PID 418`	Java-UDP ephemeral ports; outgoing-only	**OK** — sibling-project, kein Listener
`fail2ban`	nur sshd-jail aktiv; 2302 failed attempts / 320 banned	**erweiterbar** — nginx-jail für Bot-Dashboard-/Filament-Login-Brute-Force (BACKLOG SEC-1c-1)
`/srv/shares/backups/`	HTTP 403 extern — dir-mode 0700 blockt nginx-read	**OK** — Defense-in-Depth-Layer 2 wirkt (Layer 1 wäre nginx `internal` -Block)

---

6 — Bestätigt sichere Items (kein Fix nötig)

- SSH auf `0.0.0.0:22` mit `PermitRootLogin prohibit-password` + fail2ban-sshd-jail aktiv (SEC-1b-0)
- Portainer auf `127.0.0.1:9443` + `127.0.0.1:8000` — localhost-only (SEC-1b-0)
- Cockpit deinstalliert / kein Listener (SEC-1b-0)
- GUI Filament auf `127.0.0.1:8090` + DB nur intern (SEC-1b-6)
- militaria-mongo + system-mongod auf `127.0.0.1:27017+27018`
- claude-max-api auf `127.0.0.1:3456`
- openclaw 18789 + 38381 lokal
- sendmail-mta lokal 25/587
- systemd-resolved 53 nur lokal-stub
- netdata StatsD 8125 nur lokal

---

7 — Fix-Plan (Operator-Decision — NO CHANGES YET)

SEC-1c-0 (sofort, vor SEC-1c-1)

Step	Action	Risiko	Aufwand
0.1	**netdata bind auf 127.0.0.1**: `/etc/netdata/netdata.conf` `bind socket to IP = 127.0.0.1` + `systemctl reload netdata`	LOW (lokales Monitoring bleibt funktional; externe URL bricht)	10 min
0.2	**cupsd deinstallieren**: `snap stop cups && snap remove cups` (Operator-Bestätigung)	LOW (kein lokaler Drucker)	5 min
0.3	**Bot-Dashboard hinter nginx-Auth**: neuer nginx-Server-Block `listen 8051 ssl; location / { auth_basic; proxy_pass http://127.0.0.1:8050; }` + `htpasswd` -Datei **OHNE** compose-Edit / Bot-Restart (Mitigation-Layer 1). Langfristig: 8050-binding auf localhost im nächsten Bot-Maintenance-Window (Layer 2)	LOW (kein Bot-Touch)	30 min
0.4	**ufw installieren + default-deny**: `apt install ufw && ufw default deny incoming && ufw allow 22 && ufw allow 80 && ufw allow 443 && ufw allow 8088 && ufw enable` (8050-extern-Block durch ufw deny; nginx-8051 wenn neu)	MEDIUM (Lockout-Gefahr falls SSH-Regel falsch — mit `ufw allow 22` ZUERST)	20 min
0.5	**wp-test-ssl-nginx-config aufräumen**: `rm /etc/nginx/sites-enabled/wp-test-ssl && nginx -t && systemctl reload nginx`	LOW	5 min
0.6	**fail2ban-nginx-jail aktivieren**: `/etc/fail2ban/jail.local` nginx-http-auth + nginx-limit-req	LOW	20 min
0.7	**Strato-Cloud-Firewall prüfen**: im Strato-Dashboard nachsehen ob Firewall konfiguriert ist und welche Regeln gelten	OPERATOR-AUFGABE (read-only)	10 min

Gesamt SEC-1c-0: **90 min** + Strato-Check.

Korrekturen am SEC-1c-Plan (aus Operator-Pin)

- **APP_KEY-Rotation** aus SEC-1c herauslösen → eigenes **SEC-1e**. Begründung: rotiert encrypted-cast für MFA-Secret + Recovery-Codes; Fehler = Login/MFA-Tot. Eigenes Risk-Profile mit `APP_PREVIOUS_KEYS` -Compat-Layer + Staging-Test.
- **HSTS** starten mit `max-age=86400` (1 Tag); nach stabiler Woche auf 1 Jahr ohne preload; preload erst später (BACKLOG).
- **CSP** startet als `Content-Security-Policy-Report-Only` mit Reporter-URI; nach 1 Woche fehlerfrei → enforce. Filament/Livewire-Breakage so eliminierbar.
- **Admin-Rotation** bleibt nach HTTPS-Block, korrekt im Plan.
- **DB-LP App-User**: nur `SELECT/INSERT/UPDATE/DELETE` + `TEMPORARY` auf DB (CREATE TEMP-Tables erlaubt). Kein `CREATE` auf permanente Tabellen. Operator-Pin explizit bestätigt.

Neue empfohlene Reihenfolge

1. **SEC-1c-0**: Port-/Exposure-Fix (90 min, dieser Audit-Output)
2. **SEC-1c-1**: HTTPS + Cookies + Phase-1-Headers + nginx-RL
3. **SEC-1c-2**: Admin-Rotation
4. **SEC-1c-3**: DB-Least-Privilege (2-User)
5. **SEC-1c-4**: Audit-Alerts
6. **SEC-1c-5**: CSP Report-Only → enforce
7. **SEC-1c-6**: SSH-Finetuning + ufw-Refinement
8. **SEC-1d**: Passkey/WebAuthn (gated auf SEC-1c-1)
9. **SEC-1e**: APP_KEY-/Secret-Rotation (separat, eigenes Hochrisiko-Projekt)

---

8 — Boundaries (Audit-End-State)

master HEAD	`2fbc7b5` unverändert
`git status`	unverändert (Audit ist read-only)
Bot in-container PID	`290` (`python3 main.py --paper`) **unverändert**
Worker in-container PID	`1` (`python3 -m trading.command_worker`) **unverändert**
`BINANCE_TESTNET`	`true`
`managed_proposals` rows	0
docker cp / restart / stop / rm	0
systemctl stop/start/disable	0
nginx/cups/netdata config-touch	0
Modifications	0 — **pure read-only audit**

Status: **SEC-1c-0 AUDIT FERTIG** Fix-Plan oben. **STOP** per Operator-Pin.
Warte auf:

- **GO SEC-1c-0 Step 0.1** (netdata localhost-bind) — schnellster Quick-Win (10 min)
- **GO SEC-1c-0 Step 0.2** (cupsd snap remove)
- **GO SEC-1c-0 Step 0.3** (Bot-Dashboard nginx-Auth-Wrapper, kein Bot-Touch)
- **GO SEC-1c-0 Step 0.4** (ufw default-deny — Operator-Risk-Tolerance prüfen wegen Lockout-Pfad)
- **GO SEC-1c-0 Step 0.5/0.6/0.7** (orphan-cleanup, fail2ban-nginx, Strato-Check)

ODER: explizites **GO SEC-1c-0 alle 7 Steps** als ein atomarer Block.