// THE AIRLOCK IN FRONT OF A LIVE INSTANCE

Operate a live X1 instance without exposing it.

External testers drive a running ZEUS X-Trust X1 instance over the internet. **The application is never published, the NN and its interface never leave the operator's machine, no router port is opened, and no configuration, log or file store of the instance is reachable from a tester session.** NN and interface live outside the application folder by design — the gate never learns their paths and ends at the application's HTTP boundary.

01
No exposed app
Application and Doorman bind **127.0.0.1** only. Without the tunnel, nothing is reachable.

02
No inbound port
Cloudflare Quick Tunnel built **outbound**. No forwarding, no static address.

03
No operator data
Setup, settings, monitor, access log and every file-transfer path blocked at the proxy.

04
No standing service
Two ephemeral processes, no autostart. Closing both windows ends external access.

ARCHITECTURE — TWO TRUST BOUNDARIES

Tester — BROWSER
Code, password and — by own choice or operator policy — a TOTP factor. Never sees the tunnel address.

HTTPS ↓

Gate — PHP 8 · SQLITE · SHARED HOSTING
Authentication, seats, policy. Request-driven only. **Boundary 1: knowledge (code + password) · possession (TOTP) · origin (IP).**

↓ SIGNED HANDOVER TOKEN · HMAC-SHA256 · 120 S · SINGLE USE

Quick Tunnel — CLOUDFLARE · OUTBOUND
Address changes on every start, reported to the Gate over the signed heartbeat only. Testers are redirected, never told.

Doorman — PYTHON 3.12 · STDLIB · :7500
Reverse proxy, positive decision per request. **Boundary 2: shared secret, HMAC-signed messages both ways.**

Application — APP.PY · :7777
The unmodified live instance, started by the operator's own launcher. The gate ends here.

Interface → NN — OUTSIDE THE APP FOLDER
Invisible to the gate: no path known, no access made.

The Doorman trusts no header and no parameter that is not signed. One 256-bit secret in exactly two files — `config.php` (refuses to serve unless included from the Gate) and `doorman.json`. Both sides display `sha256("zxgate|"+secret)[:8]`: eight characters prove correct pairing without revealing anything. Rotation is one click.

ENFORCEMENT — WHAT ACTUALLY HOLDS

Handover token

```
b64url(payload) "." b64url(HMAC-SHA256(secret,
b64url(payload)))
payload = { v, seat, label, ip, iat, exp = iat+120,
seat_exp, nonce }
```

Verified in order: **signature** (`hmac.compare_digest`, constant time) → **expiry** → **nonce unseen** (600 s window) → **token IP = source IP**. Only then does a session exist. An intercepted link is worthless after two minutes and cannot be redeemed twice inside them.

Path deny-list — prefix match, 403 + log

```
/setup · /settings · /api/setup/ · /api/reinit · /api/upload
· /api/staged · /api/download/ · /api/monitor ·
/api/accesslog · /api/bericht · /api/berichte · /logs/
```

Maintained centrally in the Gate — no file access on the operator's machine to change policy. Open stays what a real session needs: **startup seal, baseline, vaults, auth runs, simulation**. Hiding the Setup/Settings tabs by CSS is convenience; the deny-list is the enforcement.

Heartbeat, remote kill, session

```
POST …a=heartbeat · every 60 s · X-Gate-Sig: HMAC-
SHA256(secret, "<ts>.<body>")
zx_sess = {seat, ip, exp, sid} + HMAC · HttpOnly; Secure;
SameSite=Lax
```

A seat locked in the Gate ends the running session within **60 s**, mid-run; policy comes back on the same channel. No server-side session file: validity = signature + expiry + IP equality + seat status. Unreachable Gate > 600 s → fallback to cookie lifetime instead of locking live testers out, capped by the token lifetime. On start the Doorman probes `/_gate/health` and aborts if the port already answers — no stale process with an old secret answers handovers.

WHAT A TESTER GETS — AND HOW LONG

- **Access code** ZX-XXXX-XXXX from `random_bytes` · **password** 14 chars, server-generated, shown once, stored only as `bcrypt` (cost 12).
- **TOTP** per RFC 6238 (SHA-1, 6 digits, 30 s, ±1 step); tester chooses at first login, operator can force or disable. QR rendered as SVG inside the Gate — no external service, the secret never leaves the server. A used time step is rejected.
- **IP pinned** at first successful login; every later request from another IP is refused — in the Gate and in the Doorman alike.
- **24 h** from first login · unredeemed codes expire after **24 h** · **5** failed attempts lock the seat (operator-only unlock, plus a per-IP counter); every attempt logged with time, IP, result.
- **Admin:** bcrypt + TOTP, no bypass. 5 failures lock it; unlock needs a one-time code by e-mail or Telegram *plus* a valid TOTP.

STATED LIMITS — NOT TO BE MISTAKEN FOR GUARANTEES

- The **tunnel address is not secret**, only short-lived. Without a valid session cookie it leads to the login; a call without handover is redirected back to the Gate.
- **No WAF, no rate limiting** at tunnel level beyond what Cloudflare brings by itself. Limiting is by seat count, lockouts and lifetimes.
- **TOTP over SHA-1** matches RFC 6238 and common authenticator compatibility — it is not a phishing-resistant factor. That would require WebAuthn.
- **IP binding is proof of origin**, not a cryptographic proof, and can misfire on rotating mobile addresses. `mail()` without own DKIM/SPF may land in spam filters — hence Telegram as a second unlock channel.

Stefan Trauth — CEO & Founder · GeoFold AI (Trauth Research LLC) · geofoldai.com · info@trauth-research.com · ORCID 0009-0003-9852-9788