Inverter-monitor notifications: hardening + tuning (ADR-012)

> How the NFE inverter-monitor decides what to alert on, and the 2026-08-10 hardening pass that killed the nightly "Battery low" noise and closed several alert-integrity gaps. Source of truth is the code + ADR-012 in nfe-modbus-energy-logger; this is the team-facing summary.

What triggered it

A "[NFE Nansana] Battery low - below 80 percent" email fired at 79% while the battery was doing normal peak-window arbitrage. 79% is not low; 80% was simply in the alert threshold set, which a normally-cycling battery crosses every evening. An end-to-end review found that plus several real gaps.

What changed (all live on the VM)

- **Thresholds 80,50,30,20,10 -> 30,20,10.** 80/50 crossed nightly on normal arbitrage. Alerts now fire only at genuinely-low SoC. Override per site with `MONITOR_SOC_THRESHOLDS`.
- **Debounce is now actually wired.** A repeat/oscillating low-SoC crossing is suppressed within a cooldown (`MONITOR_SOC_DEBOUNCE_SECONDS`, default 6h). It existed and was tested but had never been connected.
- **Severity scales with the level.** A 30% crossing is a WARNING (amber); at or under 10% it is CRITICAL (red). Not every low-SoC is the same red alarm anymore.
- **Silent-outage closed.** A dead inverter/link while the Pi stays up used to be invisible (the heartbeat kept firing). New `data_stale` (critical) / `data_restored` alerts cover it, after N failed reads (`MONITOR_DATA_STALE_AFTER`, default 3), debounced. The old "Inverter offline" is renamed "Monitor offline" - it only ever meant the Pi/host/network is down.
- **No more false outages from one bad frame.** A grid/battery source change now needs N consecutive readings to confirm (`MONITOR_GRID_CONFIRM_CYCLES`, default 2). A single corrupt Modbus frame can no longer email a false grid-lost/restored pair.
- **Boot-into-low-SoC now alerts** (previously silent until it recovered and re-crossed). grid_restored shows the outage duration.
- **Delivery is retried.** A one-cycle notifier or SMTP blip no longer silently drops a grid_lost - both the POST and the SMTP send retry.

- **Security.** The notifier refuses to bind a non-loopback interface with an empty token, so the /alert endpoint (which emails the whole team) can never be exposed unauthenticated.

The alert ladder (unchanged rungs)

- CRITICAL (red): monitor offline, data stale, SoC at/under 10%, battery empty, fault.
- WARNING (amber): grid lost (on battery), SoC below a threshold above the floor, over-temp.
- RESOLVED (green): grid restored, back online, data restored.
- INFO (blue): daily energy/arbitrage summary. Charge/discharge start-stop do not email individually.

Deploy + rollback

Built from merged main and deployed to the NFE OpenEMS VM as `nfe-monitor:local` + `nfe-notifier:local` (local build; the GHCR pipeline path unlocks once the nfe-monitor package is public). Previous images tagged `:prev` for rollback (`docker tag nfe-monitor:prev nfe-monitor:local && docker compose up -d`). Verified live: both containers stable (0 restarts), notifier listening + auth enforced (bad token -> 401), monitor heart-beating.

Traceable in repo

ADR-012-notification-hardening, docs/inverter-monitor.dsl (C4 model), docs/notification-severity-standard.md, PR #69. 255 tests, 99% coverage.

Alert recipients: an .env change needs a container recreate (22 Oct 2026)

The alert recipient list is `NOTIFY_RECIPIENTS` in `/opt/openems/.env` on the OpenEMS VM. A running container keeps the environment it was started with, so editing `.env` changes nothing until the notifier is recreated.

- On 24 Sep a personal address was removed from `.env`, but the notifier container had been started on 26 Sep with the old value still loaded, so alerts kept going to it until 20 Oct.
- Fix used on 20 Oct: `cd /opt/openems && docker compose up -d --no-deps nfe-notifier`, then check `docker exec openems-pilot-nfe-notifier-1 printenv NOTIFY_RECIPIENTS`.

- A recreate takes a few seconds; an alert raised in that window can be lost, so do it at a quiet time.
- Current recipients: Aaron, Dansturn and Hillary at their nearlyfreeenergy.com addresses.

Production deploy and the Hetzner snapshot cap (22 Oct 2026)

The openems-deployment `deploy-production` workflow takes a mandatory server snapshot before it changes anything. The Hetzner project allows 30 snapshots; once that cap was reached (26 Sep) every deploy failed at the snapshot step, so production stayed on the 23 Sep monitor until 22 Oct.

- Before a deploy, check the snapshot count; if it is near 30, delete the oldest (keep the most recent good pre-deploy snapshots).
- 22 Oct: the 5 oldest snapshots (23-24 Jul) were deleted and logger #94 was deployed (monitor and notifier only; backend and UI unchanged). The backend restarts during a deploy, so the monitors log `edge_disconnected` once and recover on their next 2-minute cycle.

Revision #3
Created 2026-08-10 18:09:46 UTC by hillary.arinda
Updated 2026-10-22 17:13:17 UTC by hillary.arinda