CRA Scope Determination Memo — Cellar Sentry CS-2

Prepared for Kaltwerk Instruments GmbH by PartsProof · 2026-07-28 · Not legal advice

Question

Is Cellar Sentry CS-2 within scope of Regulation (EU) 2024/2847 (the Cyber Resilience Act), and if so, which obligations bind Kaltwerk Instruments GmbH and from when?

Facts (as provided in your intake)

- Product: Cellar Sentry CS-2 — a mains-powered cold-chain temperature and humidity logger with Wi-Fi, signed ESP32 firmware, a local web UI and a companion desktop app
- Digital elements: a mains-powered cold-chain temperature and humidity logger with Wi-Fi, signed ESP32 firmware, a local web UI and a companion desktop app
- EU availability: sold through its own store and a crowdfunding campaign, fulfilled worldwide including the EU
- Economic role: manufacturer (own brand)
- On the EU market since: 2024
- Repositories / firmware: https://github.example/kaltwerk/cellar-sentry-fw

Analysis

1. Product with digital elements. Cellar Sentry CS-2 is a mains-powered cold-chain temperature and humidity logger with Wi-Fi, signed ESP32 firmware, a local web UI and a companion desktop app. It includes software (firmware and/or companion software) and performs data processing, so it is a "product with digital elements" under Art. 3 — the CRA's threshold question is met.

2. Made available on the EU market in the course of commercial activity. Kaltwerk Instruments GmbH makes Cellar Sentry CS-2 available to EU customers (sold through its own store and a crowdfunding campaign, fulfilled worldwide including the EU). Supplying in the course of commercial activity — sale, paid support, or monetized distribution — places the product on the EU market for CRA purposes; this applies regardless of where Kaltwerk Instruments GmbH itself is established.

3. Excluded sector regimes. Cellar Sentry CS-2 is not a medical device, in-vitro diagnostic, type-approved vehicle system, aviation equipment, marine equipment, or military product, so no sectoral regime displaces the CRA.

4. Classification. Nothing in the intake indicates the functions listed for "important" (Annex III) or "critical" (Annex IV) products (e.g. identity management, firewalls, smart meters); Cellar Sentry CS-2 is treated as a default-class product. For an unclassified ("default") product, conformity assessment in 2027 can follow internal control (Module A) — self-assessment, no notified body required.

Determination

Cellar Sentry CS-2 is IN SCOPE of the CRA. Kaltwerk Instruments GmbH bears the manufacturer obligations; the Article 14 reporting obligations apply from September 11, 2026.

Your obligations and dates

SAMPLE · SYNTHETIC SUBJECT · NOT A DELIVERED FILE

DATE	OBLIGATION	WHERE IT'S COVERED IN THIS PACK
Sept 11, 2026	Report actively exploited vulnerabilities and severe incidents: early warning 24h → notification 72h → final report 14d (vulns) / 1mo (incidents), to ENISA's platform + BSI / CERT-Bund. Applies to products already on the market.	`reporting-runbook.md`
Ongoing	Working intake for vulnerability reports; coordinated disclosure; user notification	`cvd-policy.md`, `security.txt`, `vuln-handling-sop.md`
Dec 11, 2027	Full essential requirements (Annex I), technical documentation, conformity assessment + CE marking, declared support period	`gap-checklist-2027.md`

Penalties for reporting failures run to €10M or 2% of worldwide turnover; essential- requirement breaches to €15M / 2.5%. Enforcement is by member-state market-surveillance authorities.

Recommended next actions (this week)

1. Publish `cvd-policy.md` and `security.txt` at https://kaltwerk-instruments.example (paths inside each file).
2. Put `reporting-runbook.md` where you'd actually reach for it in a bad week, and fill in the two blanks in its §1 (platform URL, confirmed CSIRT contact).
3. Add the SBOM CI snippet to https://github.example/kaltwerk/cellar-sentry-fw so every release regenerates `sbom/`.
4. Calendar the standing duties in `vuln-handling-sop.md` §6.

Vulnerability Handling SOP — Kaltwerk Instruments GmbH

Internal · Effective 2026-07-28 · Owner: Sample Owner · Product: Cellar Sentry CS-2

This is the working procedure behind our public CVD policy. It is written for a small team: one person can execute every step. Keep it printed or pinned; in an incident you follow this document, not memory.

1. Intake — where reports arrive

Check ALL of these; each one starts the same process:

- **security@kaltwerk-instruments.example** (the address in our CVD policy + security.txt)
- Dependency alerts (GitHub Dependabot/Advisories on https://github.example/kaltwerk/cellar-sentry-fw)
- Upstream advisories for components in our SBOM (see `sbom/` in the pack)
- Platform/marketplace notices, social mentions, customer support mail

On arrival, immediately record in the vulnerability log (§5): date/time you became aware, source, raw report. **The timestamp of awareness starts every clock.**

2. Triage — within 5 business days (same day if exploitation is claimed)

1. Reproduce or credibly assess the report.
2. Severity: score with CVSS v3.1 (calculator: first.org/cvss). Bands: critical ≥9.0 · high 7.0–8.9 · medium 4.0–6.9 · low <4.0.
3. Affected surface: which versions/hardware revisions; check the SBOM to see whether the flaw is in our code or an upstream component.
4. **The regulatory question — answer it explicitly, in writing, every time:** *Is there evidence this vulnerability is being actively exploited?* (reporter says so, honeypot/telemetry data, public exploit in use, customer incident) - **YES → open `reporting-runbook.md` NOW. The 24-hour early-warning clock is already running from the awareness timestamp in §1.** - NO → continue this SOP; re-ask the question at every new piece of information.
5. Also ask: *is this a severe incident* (compromise of our build/update infrastructure, our signing keys, or a breach affecting product security)? YES → the incident track of the runbook applies.

3. Remediate

- Critical/high: drop other work; target a fix or mitigation in days, not weeks.
- Develop the fix privately (no public commits/issues that reveal the flaw pre-release).
- Test on real hardware/config before release.
- Release through the normal signed update channel; never ship an unsigned emergency binary.
- Regenerate the SBOM for the fixed release (see the CI snippet in the pack).

4. Disclose

- Users: notify impacted users without undue delay once a fix or mitigation exists — use the user-notice template (runbook, Template D) — including what to do (update, config change) and, for exploited vulns, what to check for.
- Reporter: coordinate publication per the CVD policy (default 90 days).

- Changelog/advisory: publish an honest advisory (what, affected versions, fixed version, credit). Silence is worse than a clean advisory.

5. Record — the vulnerability log

Append one entry per issue to `vulnerability-log.md` (kept alongside this SOP):

```
ID: VUL-2026-NNN
Aware: <ISO timestamp + source>
Summary: <one line>
Severity: <CVSS + band>   Exploited: <yes/no/unknown, evidence>
Affected: <versions>      Fixed in: <version, date>
Reported to CSIRT/ENISA: <n/a | dates of 24h/72h/final filings>
Users notified: <date, channel>
Closed: <date>
```

Retention: keep entries at least 10 years (CRA documentation expectations). This log is your evidence of a working process if a market-surveillance authority ever asks.

6. Standing duties (calendar these)

- Monthly: run dependency-vulnerability checks against the SBOM; triage anything new.
- Per release: regenerate SBOM; confirm security.txt `Expires` is >30 days out.
- Yearly: re-read this SOP + the CVD policy; update contacts; test that security@kaltwerk-instruments.example actually reaches you.

CRA Reporting Runbook — Kaltwerk Instruments GmbH

The 24h / 72h / 14-day procedure · Cellar Sentry CS-2 · Prepared 2026-07-28 by PartsProof

Open this document the moment you learn of an actively exploited vulnerability in Cellar Sentry CS-2 or a severe incident affecting its security. Follow it top to bottom. Every template you need is at the end. Do not draft filings from scratch under pressure.

> Not legal advice. This runbook operationalizes Article 14 of Regulation (EU) 2024/2847 (Cyber Resilience Act) as published. Where your facts are unusual, involve counsel — after the 24-hour filing, not instead of it.

0. Does the clock apply? (60 seconds)

The obligation triggers on either of:

- **Actively exploited vulnerability** — a vulnerability in Cellar Sentry CS-2 for which there is reliable evidence of exploitation against real systems without users' consent (reporter evidence, telemetry, a public exploit being used in the wild). A vulnerability with *no* evidence of exploitation does NOT trigger Article 14 — handle it via the SOP; keep re-checking.
- **Severe incident** — an incident having an impact on the security of Cellar Sentry CS-2: compromise of your build/update infrastructure or signing keys, malicious code in a release, or a breach that undermines the product's security properties.

The clock started when you became aware — the timestamp in your vulnerability log, not when you finish investigating. Write it here before continuing:

`AWARE AT (UTC):` _____ → 24h deadline: _____

1. Where to file

- **Primary: the ENISA single reporting platform** — the CRA's unified intake portal. Bookmark the live URL during onboarding: check https://www.enisa.europa.eu and search "single reporting platform" — record it here once confirmed: _____
- **Simultaneously: your designated national CSIRT.** As a manufacturer selling into the EU with Germany as your main EU market, your CSIRT of record is: **BSI / CERT-Bund** — https://www.bsi.bund.de. (Common CSIRTs for cross-border sellers are listed in Appendix 1.)
- If the platform is unreachable, file to the CSIRT directly by email with the same template content, note the platform outage, and re-submit on the platform when it's back. Never let a portal problem eat the deadline.

2. T+24 hours — Early warning (Template A)

Minimal by design: what you know now, not what you wish you knew.

- File Template A on the platform + CSIRT.
- For an exploited vulnerability: indicate the member states where you're aware of exploitation, if identifiable. For an incident: indicate whether unlawful/malicious action is suspected and cross-border reach if apparent.

- **Filing an early warning does not obligate you to have answers.** Unknown = say "under investigation." What's sanctionable is silence, not uncertainty.
- Log the submission ID and time in the vulnerability log.

3. T+72 hours — Notification (Template B)

- General facts: nature of the vulnerability/incident, severity assessment (CVSS if scored), affected versions/hardware revisions, impact observed, indicators of compromise you can share, and any corrective or mitigating measures taken or available to users (workaround, config change, released fix).
- Update anything from the early warning that has changed.
- For incidents: this is the "incident notification" step — same deadline.

4. Interim — keep users safe while you fix

- Notify impacted users **without undue delay** once there is anything actionable (Template D): affected versions, what to do now, where the fix will land. For an actively exploited vulnerability, err early — users are being attacked.
- Preserve evidence: logs, the exploit sample, timestamps, affected units. Don't wipe compromised infrastructure before imaging it.
- If your update infrastructure is the thing compromised: freeze releases, rotate keys from a clean machine, and say so in the Template B/C filings.

5. Final report — T+14 days (exploited vulnerability) / T+1 month (incident)

Template C: root cause, full impact assessment, the fix (version, date shipped), mitigations applied, and — for vulnerabilities — the remediation guidance given to users. File it even if the fix isn't fully rolled out; state the rollout plan.

6. Afterward

- Complete the vulnerability-log entry (all filing dates + IDs).
- Publish your advisory per the CVD policy.
- Post-mortem within 2 weeks: what let this happen, what detection was missing, which gap-checklist items (see `gap-checklist-2027.md`) this incident argues for pulling forward.

Template A — Early warning (file within 24h)

```
EARLY WARNING — [actively exploited vulnerability | severe incident]
Reporting manufacturer: Kaltwerk Instruments GmbH (German GmbH)
Contact: Sample Owner, hello@kaltwerk-instruments.example
Product: Cellar Sentry CS-2 — https://kaltwerk-instruments.example/cellar-sentry
Versions affected (if known): [versions | under investigation]
Aware since (UTC): [timestamp]
Summary (2–3 sentences, no speculation): [what is known now]
Exploitation observed in (member states, if identifiable): [list | not yet known]
Suspected unlawful/malicious act: [yes | no | unknown]
Assistance requested: [none at this time | technical support requested]
```

Template B — Notification (file within 72h)

SAMPLE · SYNTHETIC SUBJECT · NOT A DELIVERED FILE

```
NOTIFICATION — follow-up to early warning [submission ID]
Manufacturer / product / contact: as above
Nature of the issue: [vulnerability class or incident type; CVE/CWE if assigned]
Severity: [CVSS vector + score | qualitative]
Affected: [versions, hardware revisions, approx. units in EU if known]
Impact observed: [what the exploitation/incident actually does]
Indicators of compromise: [hashes, IPs, log patterns — what users can check]
Corrective/mitigating measures: [fix version + date, workaround, config change]
Measures users should take now: [update / disable feature / rotate credentials]
Changes since early warning: [delta]
```

Template C — Final report (14 days / 1 month)

```
FINAL REPORT — closes [submission ID]
Root cause: [how the flaw/incident came to exist — specific, not generic]
Timeline: [aware → early warning → notification → fix shipped → users notified]
Full impact: [what happened, to how many, where]
Remediation: [fix version(s), rollout status, remaining exposure]
Mitigations for users: [final guidance]
Preventive changes: [process/engineering changes made or scheduled]
```

Template D — User security notice

```
Subject: Security update required — Cellar Sentry CS-2

We've identified a security issue in Cellar Sentry CS-2 [versions X–Y] that
[one honest sentence on impact]. [It is being actively exploited — please act today.]

What to do now:
1. [Update to version Z via the normal update channel | apply workaround]
2. [Anything to check/rotate]

What happened, in short: [2–3 plain sentences. No euphemisms.]
We've reported this to the relevant EU security authorities as required and will
publish a full advisory at [URL] on [date].

Questions: hello@kaltwerk-instruments.example — we answer these first.
— Kaltwerk Instruments GmbH
```

Appendix 1 — CSIRT quick list (verify links during onboarding)

MARKET	CSIRT	ENTRY POINT
ENISA (platform host)	—	enisa.europa.eu → single reporting platform
Germany	BSI / CERT-Bund	bsi.bund.de
France	CERT-FR (ANSSI)	cert.ssi.gouv.fr
Netherlands	NCSC-NL	ncsc.nl
Ireland	NCSC-IE	ncsc.gov.ie
Belgium	CCB / cert.be	ccb.belgium.be

SAMPLE · SYNTHETIC SUBJECT · NOT A DELIVERED FILE

MARKET	CSIRT	ENTRY POINT
Italy	CSIRT Italia (ACN)	csirt.gov.it
Spain	INCIBE-CERT	incibe.es
Poland	CERT Polska	cert.pl
Sweden	CERT-SE	cert.se
Austria	CERT.at	cert.at

SAMPLE · SYNTHETIC SUBJECT · NOT A DELIVERED FILE

CRA 2027 Conformity Gap Checklist — Cellar Sentry CS-2

Prepared for Kaltwerk Instruments GmbH by PartsProof · 2026-07-28 · Target: December 11, 2027

Your reporting readiness (this pack) covers September 2026. Full CRA conformity — the Annex I essential requirements, technical documentation, and CE marking — lands **December 11, 2027** and is engineering work, not paperwork. This checklist is the honest map of that work for Cellar Sentry CS-2. Work top to bottom; the early items make the later ones cheaper.

Status legend: □ not started · ◐ partial · ✓ done

A. Secure development baseline (Annex I Part I)

#	REQUIREMENT	WHAT IT MEANS FOR CELLAR SENTRY CS-2	STATUS
A1	No known exploitable vulnerabilities at release	Pre-release dependency + firmware scan against your SBOM; document the check per release	□
A2	Secure-by-default configuration	Ship with unique-per-device or forced-first-boot credentials, closed debug ports, minimal enabled services	□
A3	Security updates	A working, authenticated update mechanism; signed images; rollback protection	□
A4	Attack-surface minimization	Inventory every open port/interface/service; kill or gate what isn't needed	□
A5	Protection of data (confidentiality/integrity)	TLS for transit; encrypt stored secrets; verify boot chain integrity where feasible	□
A6	Access control	AuthN/AuthZ on every management surface (device, app, cloud API)	□
A7	Resilience / DoS mitigation	Rate-limit and fail-safe on network faces; watchdog recovery	□
A8	Data minimization	Collect only what the product needs; document what and why	□
A9	Logging	Security-relevant events recordable (auth failures, update events) with user access	□
A10	Secure decommissioning	Factory reset that actually wipes user data and credentials	□

B. Vulnerability handling (Annex I Part II) — you start ahead here

#	REQUIREMENT	COVERED BY	STATUS
B1	Identify + document vulnerabilities; SBOM	`sbom/` + CI snippet (this pack)	◐
B2	Handle + remediate without delay	`vuln-handling-sop.md`	✓
B3	Regular testing/review	Monthly SBOM check (SOP §6); add an annual self-pentest	◐
B4	Publish advisories for fixed vulns	CVD policy §Disclose	✓
B5	Coordinated disclosure policy	`cvd-policy.md`	✓
B6	Intake for reports (security.txt)	`security.txt`	✓
B7	Secure update distribution	Overlaps A3 — signed, authenticated channel	□

SAMPLE · SYNTHETIC SUBJECT · NOT A DELIVERED FILE

#	REQUIREMENT	COVERED BY	STATUS
B8	Free security updates for the support period	Declare the support period (≥5 years or expected lifetime); publish it	☐

C. Paper conformity

#	REQUIREMENT	WHAT TO PRODUCE	STATUS
C1	Technical documentation (Annex VII)	Design/architecture description, risk assessment, test evidence, SBOM — start a `tech-file/` dir now, fill as you do A1–A10	☐
C2	Cybersecurity risk assessment	Written, product-specific, kept current — feed it from incidents + the log	☐
C3	Conformity assessment	Default-class product → internal control (Module A) self-assessment — no notified body needed.	☐
C4	EU declaration of conformity + CE marking	Drafted from C1–C3; CE mark on product/packaging/docs	☐
C5	Support-period declaration	User-facing statement of the update-support window	☐
C6	Instructions + security info for users	User-facing security documentation (setup, updates, EOL)	☐

Suggested order of attack

1. **Now → Q4 2026:** A2, A3, A4 (they compound: every later item assumes them), C1 skeleton.
2. **Q1–Q2 2027:** A1, A5–A7, B7, B8 declaration, C2.
3. **Q3 2027:** A8–A10, C3–C6, full tech-file assembly.
4. **Buffer:** aim to be done by **October 2027** — leave slack for the surprises.

Questions while you work through it: reply to your delivery email — answering them is part of the pack.