PIERRE BOCKNER · EMBEDDED SYSTEMS & EDGE AI

Yocto-Lizenz-Risiko-Checkliste

Praktische Selbstbewertung für Embedded-Linux-Produktteams · bockner.com

> Sechzehn Prüfpunkte in vier Risikobereichen. Gehen Sie sie anhand Ihres *tatsächlichen Release-Images* durch, nicht anhand Ihrer Absichten. Jeder offene Punkt in Abschnitt A oder B ist ein potenzieller Release-Blocker; die Abschnitte C und D zeigen Wartbarkeits- und Beschaffungsrisiken. Version 2026-11.

A · Lizenz-Inventar & Traceability

- ☐ **Erfasste Lizenzen.** Jedes Paket im Auslieferungs-Image hat eine erfasste Lizenz als SPDX-Identifier — kein Freitext, kein „unbekannt“.
- ☐ **Image-basiertes SBOM.** Ein SBOM (SPDX oder CycloneDX) wird aus dem tatsächlichen Release-Image erzeugt, nicht aus Build-Abhängigkeiten geraten.
- ☐ **Geprüfte Recipe-Metadaten.** LICENSE-Angaben der Recipes sind gegen die Upstream-Quellen verifiziert; generische Fallback-Lizenzen werden nicht als Abkürzung genutzt.
- ☐ **Trennung Auslieferung vs. Build.** Reine Build-Werkzeuge (native/nativesdk-Pakete) sind klar von Target-Paketen getrennt, die tatsächlich an Kunden ausgeliefert werden.

B · Copyleft- & GPLv3-Exposure

- ☐ **Bekannter GPLv3-Bestand.** Sie wissen genau, welche GPLv3-/AGPL-Komponenten im Auslieferungs-Image landen — z. B. über das Lizenz-Manifest eines Release-Builds.
- ☐ **Dokumentierte Grenzen.** Copyleft-Grenzen sind dokumentiert: welcher Code proprietär ist, welcher von Copyleft-Code abgeleitet ist und wie beide getrennt werden.
- ☐ **Kompatibilität geprüft.** Die Lizenz-Kompatibilität kombinierter und gelinkter Werke ist geprüft (z. B. GPL vs. Apache-2.0, Werbeklauseln, Exception-Klauseln).
- ☐ **Quarantäne-Plan.** Für Komponenten, die Rechtsabteilung oder Einkauf ablehnen, existiert ein Ersatz- oder Quarantäne-Plan — vor dem Design-in, nicht danach.

C · Build-Reproduzierbarkeit

- ☐ **Version-Pinning.** Alle Layer und release-relevanten Recipes sind versionsgepinnt; kein floating AUTOREV in Release-Builds.
- ☐ **Nachbau verifiziert.** Ein Release-Image lässt sich Monate später bitgenau oder mit nur dokumentierten, kontrollierten Abweichungen nachbauen.
- ☐ **Archivierte Inputs.** Shared-State-Cache und Source-Downloads sind pro Release archiviert; tote Upstream-URLs sind kein Single Point of Failure.
- ☐ **Sauberer CI-Build.** Die CI baut das Release aus einem frischen Checkout ohne manuelle Schritte oder Zustand einer Entwicklermaschine.

D · Wartung & Beschaffung

- ☐ **LTS-Strategie.** Kernel- und BSP-Langzeitwartung deckt die vorgesehene Feldlaufzeit ab (z. B. 10 Jahre), inklusive Security-Backports.

- [ ] **Compliance-Nachweise auf Abruf.** SBOM, Lizenz-Hinweise und Source-Offers lassen sich innerhalb von Tagen an Kunden liefern — nicht nach wochenlanger Recherche.
- [ ] **Einkaufs-Freigabe.** Der Einkauf hat die Open-Source-Komponenten vor dem Design-in freigegeben — mit einem Nachweis, auf den Sie verweisen können.
- [ ] **Verantwortlicher für Lizenz-Monitoring.** Eine benannte Person beobachtet Upstream-Lizenzänderungen und neue Versionen der ausgelieferten Komponenten.

> **So lesen Sie Ihr Ergebnis:** Offene A/B-Punkte sollten das nächste Release blockieren, bis sie geklärt sind. Offene C/D-Punkte sind Schulden, die sich mit jedem Jahr im Feld verzinsen — einplanen, nicht ignorieren.

Soll das an Ihrem echten Build-System geprüft werden? Starten Sie mit einer kostenlosen, vertraulichen 15–20-minütigen privaten technischen Review: bockner.com/review. Fest umrissenes Folgeprojekt: Yocto Build- & Lizenz-Risikoscanner — bockner.com/yocto-risk-scan.

Pierre Bockner · Senior Embedded Software Engineer · Nürnberg · bockner.com · info.bockner@posteo.de · Diese Checkliste ist allgemeine Orientierung, keine Rechtsberatung.