BLUE GOAT CYBER

Medical Device Cybersecurity · FDA Specialists

MEDICAL DEVICE SBOM FIELD GUIDE

The Medical Device SBOM Field Guide

End-to-end operations: tools, formats, binary analysis, VEX, distribution, and the review-letter findings to pre-empt.

© Blue Goat Cyber. For educational use. Not legal or regulatory advice.

SBOM OPERATIONS

The medical device SBOM field guide.

An SBOM is only as valuable as your ability to keep it accurate and act on it. The FDA's 2026 final guidance assumes you can produce a machine-readable SBOM, monitor it daily, and turn CVE noise into reviewable VEX. This field guide is the operating manual: which tools, which formats, which workflows, and where teams typically fail.

Key takeaway: The most common SBOM failure mode is not technical - it's organizational. No single owner, no daily monitoring cadence, no VEX discipline. Fix the operating model first; tools second.

1. Pick a canonical format

CycloneDX 1.5	Default for medical devices. Native VEX, ML-BOM, services, formulation. OWASP-stewarded.
SPDX 2.3+	Generate when downstream tooling needs it. Strong on license metadata. Linux Foundation.
Format rule	Pick one as canonical, document why. Never ship a PDF or spreadsheet as the SBOM itself.

2. Tool selection by device class

SaMD / cloud	Syft, cdxgen, Trivy. Language plugins (npm, pip, maven, go) for highest fidelity.
Mobile companion app	cdxgen for iOS (CocoaPods, SwiftPM) and Android (Gradle). Combine with platform reports.
Embedded firmware	Syft + binary analysis: Binwalk for extraction, Ghidra-assisted ID, BLINT, EMBA framework.
Containers	Syft or Trivy at build time, integrated with your registry. Layer-aware.
Vendor firmware	Request vendor SBOMs in writing. Document the gap until they ship one.

3. Binary analysis - the hard part of firmware SBOMs

- Extract filesystems with Binwalk; identify components by signatures + string analysis.
- Reconcile binary findings with the build system's bill of materials - close every gap.

- Capture statically linked libraries that don't appear in package manifests.
- Document confidence per component: **high** (build-system) vs **medium** (binary match) vs **low** (heuristic).
- Repeat at every release - drift between build and binary is where vulnerabilities hide.

4. PURL + hash discipline

- Every component gets a Package URL (purl) - reviewers use these to map CVEs.
- SHA-256 minimum for binary components; SHA-512 preferred.
- Include supplier, version, and SPDX license identifier on every component.
- Validate every SBOM with cyclonedx-cli or spdx-tools before publishing.
- Reject SBOMs that fail validation - don't 'fix in post'.

5. The VEX operating model

Five worked justifications you can copy:

component_not_present	Vulnerable subcomponent isn't shipped (e.g., we use a stripped build of OpenSSL without S/MIME).
vulnerable_code_not_present	We ship the library but excluded the vulnerable file at build time.
vulnerable_code_not_in_execute_path	Code is present but the affected function is never called by our app.
vulnerable_code_cannot_be_controlled_by_adversary	Reachable but inputs are fixed by our software (no attacker influence).
inline_mitigations_already_exist	Sandbox, seccomp, network ACLs, or memory protection neutralize the threat.

6. Operating cadence

- **Every build:** regenerate SBOM, diff against previous, fail build on unreviewed critical components.
- **Daily:** automated CVE scan against latest SBOM. Auto-create tickets for High+.
- **Weekly:** security engineer triage. Set VEX status. Close out resolved.
- **Monthly:** publish updated SBOM + VEX to customers per your CVD policy.
- **Quarterly:** SBOM hygiene - drop EOL components, update PURLs, validate licenses.

7. Distribution to hospitals and operators

- Align SBOM-derived security posture with the MDS^2 form sections.
- Cover NTIA minimum elements - supplier, component, version, dependency, hash, author, timestamp.
- Customer portal with current SBOM + VEX + advisories - one source of truth.

- Signed SBOM (in-toto attestation or sigstore) for high-assurance customers.
- Clear retention policy - keep SBOMs for the full device support lifetime.

8. Common review-letter findings (and how to pre-empt them)

Format mismatch	Submission contains PDF SBOM. Fix: ship CycloneDX 1.5 JSON in the eSTAR upload.
Missing transitive deps	Top-level only. Fix: regenerate with full dependency tree.
No PURLs	Components without purl can't be CVE-mapped. Fix: add purl to every component.
Stale CVE assessment	Reviewer's scanner finds new CVEs you haven't triaged. Fix: monthly publish cadence.
No process evidence	Single snapshot looks like a one-time report. Fix: include sample VEX statements + ticket trail.

Key takeaway: If a reviewer can't reproduce your CVE list by feeding your SBOM into an open scanner, your SBOM isn't compliant - regardless of how thorough the PDF appendix looks.

READY FOR THE NEXT STEP?

Talk to a senior medical device security engineer.

We've contributed cybersecurity documentation, threat models, SBOMs, and pen test reports to **250+ FDA submissions** across 510(k), De Novo, and PMA pathways - with a **100% submission success rate**. If you're building a cyber device and want to go in with airtight artifacts on the first try, we should talk.

Book an SBOM working session
30 minutes. No slide deck. Bring your device profile and your top three questions.

bluegoatcyber.com/contact

WHAT WE DO

- **FDA premarket cybersecurity** - SPDF, threat model, SBOM, pen test, security risk assessment, eSTAR-ready package
- **FDA cybersecurity deficiency response** - rapid remediation when reviewers send a letter
- **Postmarket programs** - vulnerability monitoring, VEX, coordinated disclosure, patch validation
- **Legacy device remediation** - bring older devices up to current FDA expectations