THE LINUX FOUNDATION

OPEN SOURCE SUMMIT INDIA

Hardening IoMT Medical Devices

Defense-in-Depth on Yocto-Based Embedded Linux

Practical patterns for secure, regulated, long-lifecycle medical devices

Who am I?

Abraham Gogulamudi

Senior Engineering Manager, GE Healthcare

linkedin.com/in/abraham-gogulamudi

20 Years of Experience

Platforms · Strategy · Healthcare · Cybersecurity · IoMT

Open Source & Embedded Systems

Medical-grade Linux on Yocto Project, powering multiple product lines

IoMT & Inner Source Champion

Secure, interoperable medical device data via Inner Source collaboration

Why This Matters Now

The healthcare cybersecurity crisis is escalating—medical devices are prime targets

99%

of hospitals

manage IoMT devices with **known exploited vulnerabilities** (KEVs) actively targeted by threat actors [1]

77%

targeted by ransomware

of healthcare organizations were hit in 2024, with **53% paying the ransom**—averaging $7M demands [2]

$10.22M

average breach cost

Healthcare breaches cost **2.8× more** than other industries—15 consecutive years as most expensive sector [3]

~60%

of devices are **end-of-life** with no security patches available [4]

305M+

patient records exposed in 2024 breaches [5]

20%

increase in mortality rates at hospitals affected by cyberattacks [6] [7]

The Regulatory Landscape

Cybersecurity is now a Regulatory Mandate – Not Optional

THE LINUX FOUNDATION OPEN SOURCE SUMMIT INDIA

1. Legal foundation (Non-Negotiable)

FDA Section 524B

Mandatory cybersecurity for "Cyber Device"

- SBOM
- Secure Design
- Vulnerability Management

Compliance = Market access in the US

2. Regulatory Guidance(How to Comply)

FDA Cybersecurity Guidance

Defines expectations of the submission and throughout device lifecycle

- Threat Modeling
- SBOM & CVE Mgmt
- Secure Architecture
- Lifecycle Management

The "Bread & Butter" for FDA Submission

3. Global Convergence

EU MDR / IVDR

GSPR Annex I—security as part of safety and performance requirements

IEC 81001-5-1

State-of-the-art standard for health software security—lifecycle processes

IMDRF Principles

International harmonization—total product lifecycle, transparency, security by design

Regulators are now expected secure-by-design across the entire product lifecycle

4. Lifecycle Obligations (Pre-Market to Post-Market)

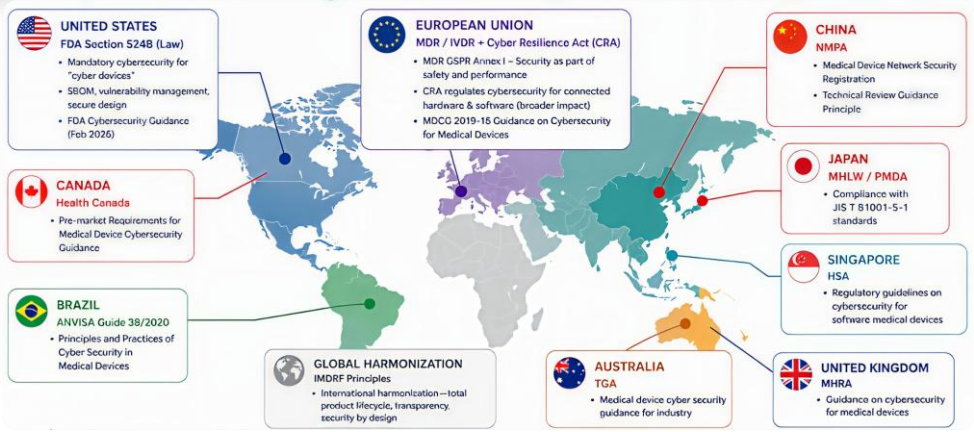
Pre-Market (Design & Development)

- Threat Modeling
- Secure Architecture
- SBOM Required
- Security Risk Analysis

Post-Market (Operations & Support)

- Continuous Monitoring
- Coordinated Disclosure
- Secure OTA Patching
- Post Market Surveillance

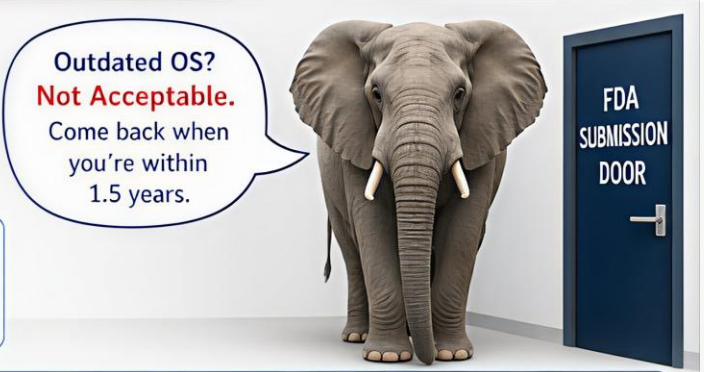
Obligations: 10–15-year lifecycle support | Incident Response | Breach notification



Global Message : Cybersecurity expectations are globally aligned – even if regulations differ, the outcome is the same: Secure by Desing , Secure in Operations

THE ELEPHANT IN THE ROOM

FDA will not generally accept an OS with less than **1.5 years** of support remaining



FDA

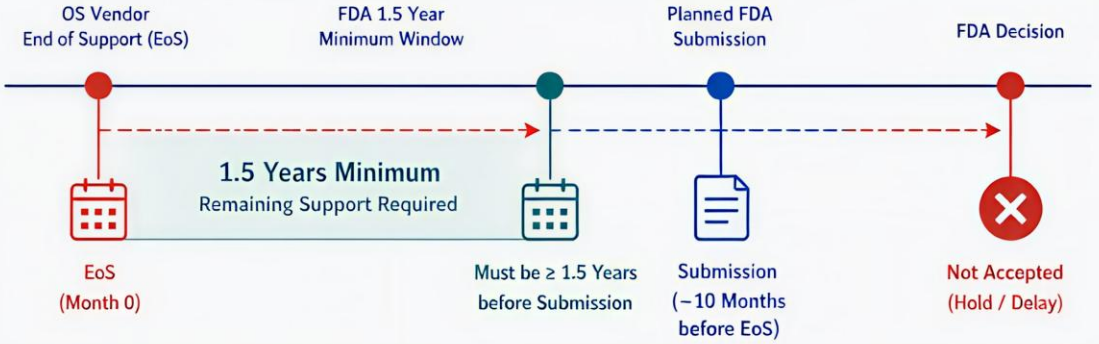
"Devices must be designed, developed, and maintained with the ability to **receive security patches and updates throughout their intended lifecycle.**" – FDA Cybersecurity Guidance for Medical Devices

REAL-LIFE EXAMPLE

A medical device platform was planned on an Embedded Linux OS with vendor support ending in 18 months.

Submission planned
10 months before End of Support (EoS).

FDA Outcome:
Submission placed on hold.
OS end-of-support does not meet lifecycle expectations.



WHY FDA CARES

- Outdated OS = Unpatchable vulnerabilities
- Increases patient, data, and hospital risk
- Violates lifecycle cybersecurity expectations
- Higher risk of post-market security failures

BUSINESS IMPACT

- Submission delays (months to years)
- Rework, retesting, and documentation
- Lost revenue & market opportunity
- Damaged customer trust & brand

COST OF DELAY (EXAMPLE)

6–12 Months Delay

$2M–$10M+ Potential Impact

PLAN EARLY. CHOOSE WISELY. STAY SUPPORTED. | Ensure your OS has ≥ 1.5 years of support remaining at the time of FDA submission.

⚠ This applies to ALL critical software components — OS, middleware, libraries, and third-party components.

Pre-Market Cybersecurity Expectations

Four pillars of secure medical device design—start early, document everything

THE LINUX FOUNDATION OPEN SOURCE SUMMIT INDIA

01 Threat Modeling

Systematic identification of threats using **STRIDE methodology**—must begin in design phase

- Attack surface analysis
- Trust boundaries identification
- Data flow diagrams (DFD)
- Threat actor profiling

Tools: Microsoft Threat Modeling Tool, OWASP Threat Dragon, IriusRisk

02 Security Risk Analysis

Integrate cybersecurity into **ISO 14971** risk management—cyber threats are safety risks

- Threat-asset-vulnerability mapping
- Risk scoring (probability × severity)
- Risk control measures
- Residual risk assessment

Output: Risk management file with cyber risk traceability matrix

03 Secure-by-Design Architecture

Build security in from the ground up—**not as an afterthought**

- Defense-in-depth layering
- Least privilege principle
- Secure defaults configuration
- Fail-secure mechanisms

Principle: Security through design, not obscurity

04 SBOM Generation

Software Bill of Materials—now mandatory for FDA submissions

- Complete component inventory
- Dependency tree tracking
- CycloneDX or SPDX format
- Version and license info

Yocto: meta-sbom layer automates generation

Critical Success Factor: Start cybersecurity activities in the concept phase—retrofitting security costs 100× more

Threat Modeling – The Foundation

STRIDE methodology: Identify threats before they identify you

THE LINUX FOUNDATION OPEN SOURCE SUMMIT INDIA

1. Why Threat model?

- Identify threats early in the lifecycle
- Understand how the system can be attacked
- Prioritize risks and focus on what matters
- Design the right security controls

Security start before code is written

2. STRIDE Framework

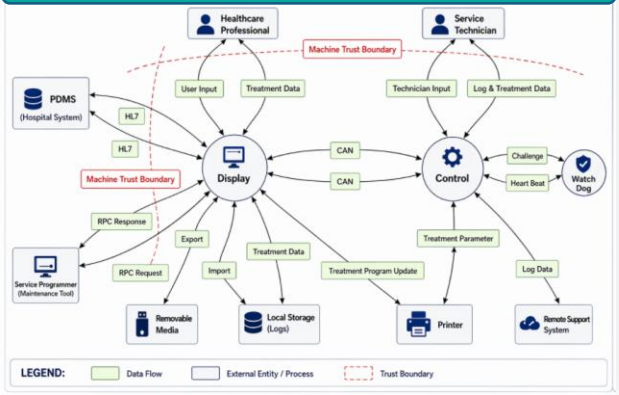
- **S** Spoofing
- **T** Tampering
- **R** Repudiation
- **I** Information Disclosure
- **D** Denial of Service
- **E** Elevation of Privilege

Covers Complete threat landscape

3. Threat Modeling Process

1. Define Architecture (DFD)
2. Define Trust Boundaries
3. Map Threats (STRIDE)
4. Assess Risk
5. Apply Mitigations

Reference Threat Model



Identifying Potential Threats (STRIDE Examples)

Understand how things can go wrong to build the right security controls

Attack Class	Attack Class	Description	Example
S	Spoofing	Pretending to be someone else	A hacker pretends to be an authorized user
T	Tampering	Modifying data or code	Malware encrypts data
R	Repudiation	Denying having done something	A user denies that they treated the patient with the device using certain parameters
I	Information disclosure	Confidential information is displayed to unauthorized people	A nurse can see the diagnosis of a VIP patient whose treatment she is not involved in and whose data she should not be able to see
D	Denial of service	DoS attack	A bot network floods a web server with HTTP requests
E	Elevation of privilege	A person or system obtains privileges they/it should not have	A user manages to make themselves an administrator of a system

Identify threats early. Assess impact. Design controls. Reduce risk.

Defining and Implementing Countermeasures

Apply the right controls to reduce risk and protect patient safety

Attack Class	Examples of Actions (Countermeasures)
S Spoofing	**Authentication, e.g., with Kerberos** Verify the identity of users, devices and processes
T Tampering	**Digital signatures** Detect unauthorized changes to data or code
R Repudiation	**Secure audit logs** Record actions in a tamper-evident, time-stamped manner
I Information disclosure	**Encryption** Protect confidentiality of data in transit and at rest
D Denial of service	**Request filtering** Limit and validate requests to prevent DoS attacks
E Elevation of privilege	**Access control lists (ACLs)** Restrict permissions to the minimum necessary

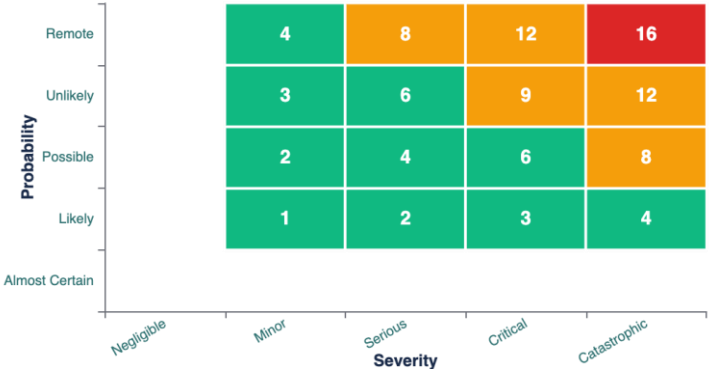
Select the right countermeasures. Implement consistently. Monitor continuously.
Strong controls today. Safer patients tomorrow.

#OSSUMMIT

Security Risk Analysis (ISO 14971 + Cybersecurity)

Integrate cybersecurity threats into medical device risk management

Risk Matrix: Probability × Severity



Threat-Asset-Vulnerability Mapping

Asset: Patient Data (PHI)

Threat: Unauthorized access

Vulnerability: Weak authentication

Risk Score: 4 × 5 = **20 (High)**

Asset: Device Firmware

Threat: Malicious modification

Vulnerability: No integrity verification

Risk Score: 3 × 5 = **15 (High)**

Asset: Network Communication

Threat: Man-in-the-middle attack

Vulnerability: Unencrypted transmission

Risk Score: 3 × 4 = **12 (Medium)**

Risk Scoring Formula

Risk = Probability × Severity

Probability Levels:

1=Remote, 2=Unlikely, 3=Possible, 4=Likely, 5=Almost Certain

Severity Levels:

1=Negligible, 2=Minor, 3=Serious, 4=Critical, 5=Catastrophic

Risk Control Measures

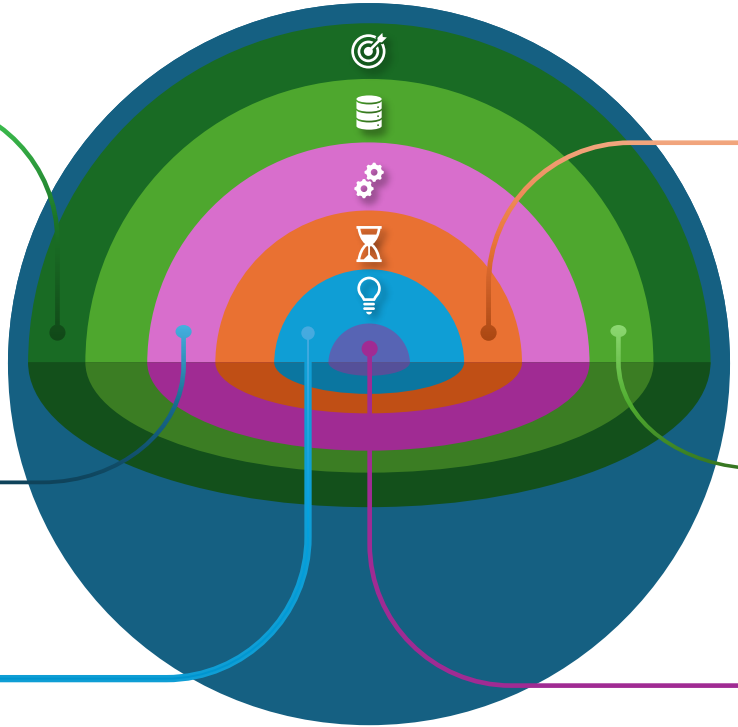
Mitigation
Reduce probability/severity

Transfer
Insurance, third-party

Accept
Document residual risk

Defense-in-Depth Overview

Six independent security layers—each protects even if others fail



SBOM & Vulnerability Mgmt 6

SPDX, CycloneDX, VEX,CVE-CHECK

Integrity Protection 3

dm-verity, dm-crypt, fs-verity, IMA/EVM, meta-integrity layer

Runtime Control & USB Guard 4

AppArmor, seccomp, USB Guard, AuditD, Rsyslog

Secure OTA & Anti-Rollback 5

SWUpdate, Signed Images, cryptographic verification, version control

OS Hardening on Yocto 2

security_flags, read-only rootfs, hardened kernel config, systemd, network firewalls

Secure Boot & Chain of Trust 1

U-Boot signing, TPM integration, anti-rollback protection

Defense-in-Depth Principle

Each layer is independent—if one fails, others continue protecting. No single point of failure.

LAYER 1

Secure Boot & Chain of Trust

The foundation—without hardware root of trust, all other layers are compromised

Why Secure Boot is Essential ?

- **Ensures device integrity & potent safety**
- **Reduces risk of field attacks and downtime**
- **Build trust in connected medical device**
- **Meets regulatory requirements (FDA,EU, IEC)**

What are secure boot key controls ?

- **Keys:** Private keys sign images, Public keys to verify
- **Signatures :** All boot components are cryptographically signed.
- **Verification :** Each stage verifies the next stage before handling overall control
- **Anti-Rollback :** Version counters prevent execution of downgrade
- **Measurement (Optional) :** Hash measurements in TPM and attestation & integrity monitoring

Attack Vectors Mitigated

- **Tampering:** Prevents modification of Bootloader, Kernel or rootfs
- **Boot kits & Rootkits:** Blocks malicious code from executing or persisting across reboots
- **Malicious /Unsigned Updates:** Rejects unsinged or altered image during boot
- **Downgrade Attacks:** Anti-rollback mechanism prevents loading of vulnerable older versions
- **Physical Tampering :** Detects & blocks untheorized modifications.

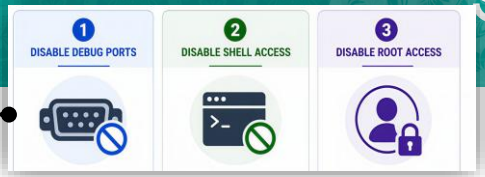
Hardware Dependencies

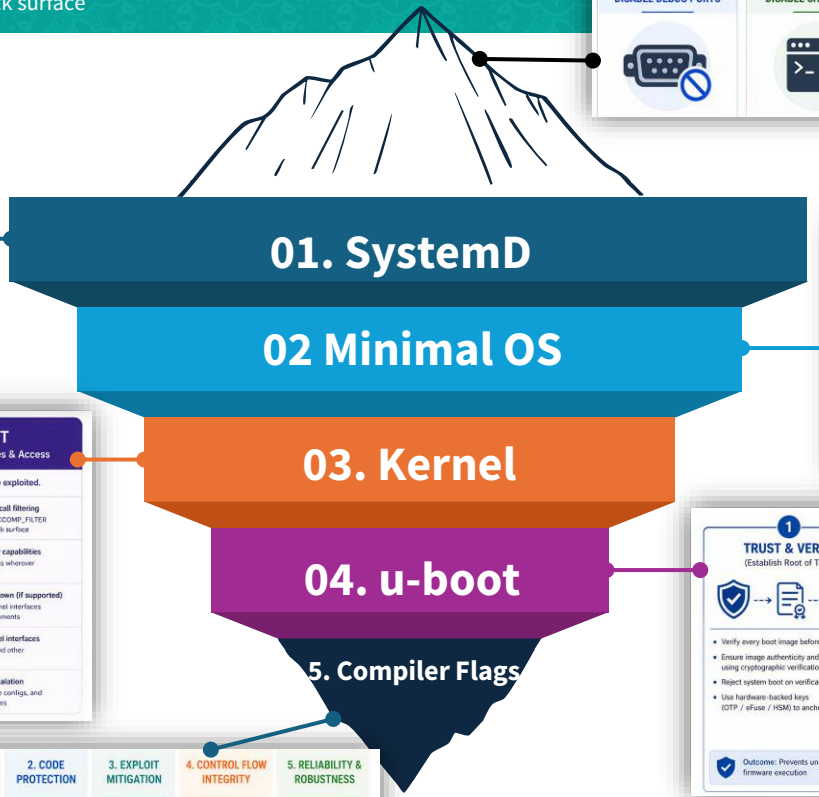
- **Hardware Root of Trust:** Secure Enclave, Boot ROM
- **Secure Key Storage :** eFUSE/OTP/TPM
- **Cryptographic Engines :** RSA / ECDSA / SHA
- **TPM for Measured Boot & Attestation (Optional)**
- **Monotonic Counter / Anti-Rollback Storage**

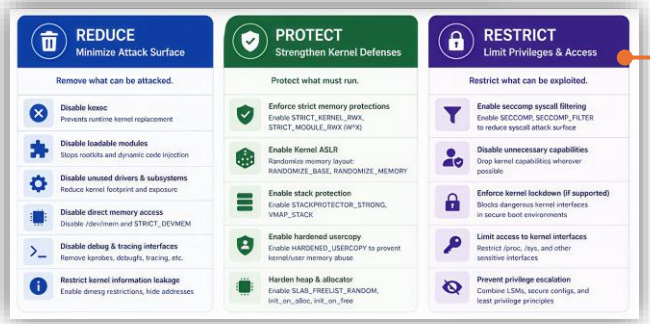
LAYER 2

OS Hardening on Yocto

Compiler hardening, kernel security, minimal attack surface







LAYER 3

Integrity Protection

Detect tampering at boot time and runtime—dm-verity, IMA/EVM, fs-verity

Filesystem Integrity (At Rest)

1 Read-only FS (Immutability)

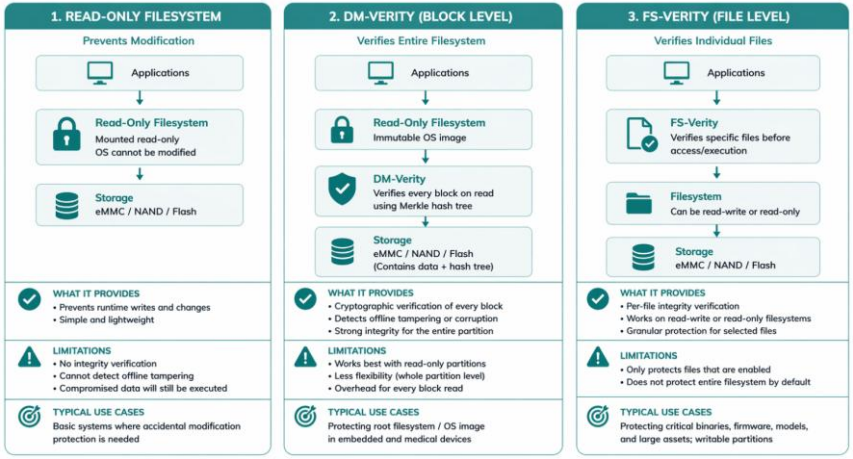
Prevents modification of OS components

2 DM-Verity (Storage Integrity)

Block-level verification using cryptographic has tree

3 FS-Verity (File Integrity)

File-level verification for critical binaries/files



Run Time Integrity (In Execution)

1 IMA

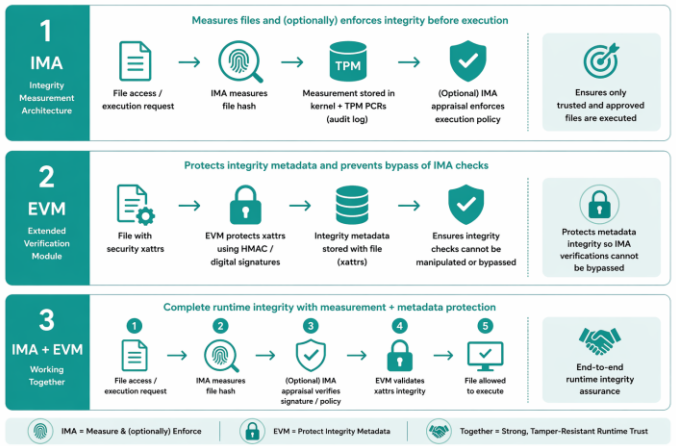
Integrity Measurement Architecture

2 EVM

Extended Verification Module

Detects unauthorized changes and blocks untrusted code execution

Protects integrity metadata and prevents tampering/bypass of IMA checks



LAYER 4

Runtime Control & Peripheral Protection

Restrict what compromised processes can do—least privilege in action

1 MAC
Mandatory Access Control

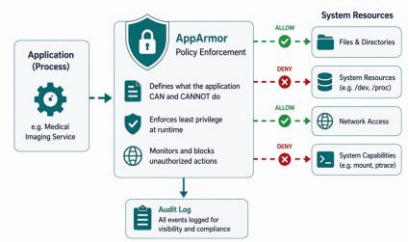
AppArmor ensure applications can do only what they need – nothing more

Define Policies

Enforces Least privilege

Monitors activity

Logs events



2 Seccomp
Secure Computing Mode

Seccomp Filter (BPF) Restrict system calls to reduce kernel attack surface

Prevents privilege escalations

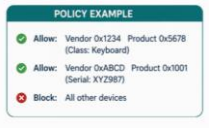
Lightweight & efficient

3 USB Guard
USB Device Control

Allow only trusted USB devices, Block unknown and malicious devices

Prevents against data exfiltration

Reduce physical attack surface

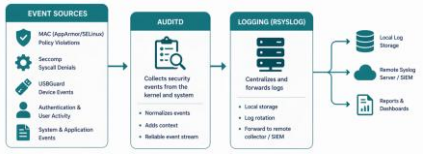


4 AuditD & Logging
Ensure nothing is missed, everything is traceable

Capture, Centralize and retain security events for visibility, forensics & compliance

Rsyslg Centralizes & forwards logs

Helps to meet regulatory & audit requirements



LAYER 5

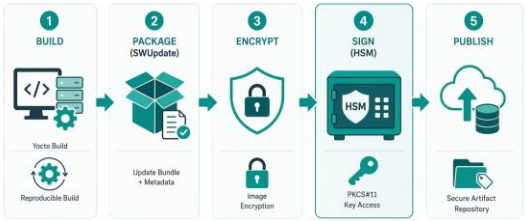
Secure OTA & Anti-Rollback

Cryptographically verified updates with downgrade attack protection

THE LINUX FOUNDATION
OPEN SOURCE SUMMIT INDIA

1 Image Build Infrastructure

- Reproducible Yocto builds
- Update bundle + metadata (SWUpdate)
- Image Encryption
- PKCS#11 Key access and Signing
- Secure & Publish Artifcats



2 Image Deployment Infrastructure

MENDER RAUC hawkBit

- Device provisioning
- Staged Rollouts
- Strong Device Identity
- Zero Trust Connectivity

3 Image Installation & Anti-Rollback

- Decrypt using device protected key
- Verify authenticity and Integrity
- Install to Inactive slot and set boot target
- Anti-Rollback Protection

#OSSUMMIT

LAYER 6

SBOM & Automated Vulnerability Management

Transparency and continuous monitoring—required by FDA Section 524B

THE LINUX FOUNDATION OPEN SOURCE SUMMIT INDIA

1 SBOM Creation with SPDX

- Machine-readable inventory of all software components, licenses & meta data
- Enable trace from source to deployed device
- Foundation for vulnerability and audits

Format: SPDX 2.3 | **Tooling :** meta-spdxscanner | **Outputs :** .spdx, .spdx.json

2 SBOM Creation with CycloneDX

- Complements SPDX for broader ecosystem compatibility
- Support risk scoring and prioritization
- Optimized for security tools, dashboards

Format: CycloneDX 1.5 | **Tooling :** meta-cyclonedx | **Outputs:** .vex , .bom

3 CVE-CHECK Report

- Identified affected components and fixes availability
- Scan SBOM against NVD,KEV and Vendor advisories
- Provides security, CVSS score

Tool: cve-check

4 Open-Source License Information Disclossure

- Extract and aggregate license data from SBOM
- Generate Human-readable, Machine-readable license reports
- Identify license types, obligations and compatibility

Output : License report (DOCX,CSV,JSON)

Post-Market Cybersecurity Obligations

Security doesn't end at launch—10-15year lifecycle commitment

Continuous Monitoring

- **Threat Intelligence Feeds**
  NVD, OSV, vendor advisories, security mailing lists
- **Vulnerability Databases**
  Continuous CVE monitoring against SBOM components
- **Security Advisory Tracking**
  Upstream project notifications, CERT alerts

Frequency: Daily automated scans + weekly manual review

Coordinated Vulnerability Disclosure

- **PSIRT Establishment**
  Product Security Incident Response Team
- **Responsible Disclosure Process**
  Clear reporting channels, researcher engagement
- **CVE Management**
  Request CVE IDs, publish advisories, document mitigations

Timeline: Acknowledge reports within 5 business days

Secure OTA Patching

- **Patch Deployment Pipeline**
  Automated build, test, sign, deploy workflow
- **Emergency Patch Capability**
  Fast-track process for critical vulnerabilities
- **Staged Rollout**
  Canary deployment, gradual fleet update

Target: Critical patches within 30 days of CVE disclosure

Post-Market Surveillance

- **PMS Plan**
  Post-Market Surveillance plan with security focus
- **Periodic Security Assessments**
  Annual penetration testing, architecture review
- **Risk Re-evaluation**
  Update risk analysis based on new threats

Lifecycle: 10-15 year support commitment

Incident Response: Report to FDA within 30 days of breach discovery; notify affected users without unreasonable delay

FDA Section 524B
EU MDR Article 87

Key Takeaways

Actionable patterns for securing regulated medical devices

THE LINUX FOUNDATION
OPEN SOURCE SUMMIT INDIA

1 Defense-in-Depth is Non-Negotiable

Every layer must be **independent**—no single point of failure. If one layer falls, others continue protecting. This is not optional for regulated medical devices.

2 Secure Boot is the Foundation

Without hardware root of trust, all other layers are compromised. Implement chain of trust from ROM → U-Boot → Kernel → Rootfs.

3 Yocto Hardening Wins

Enable security_flags, use read-only rootfs, apply hardened kernel config—immediate security improvements with minimal effort.

4 Sandboxing Dramatically Reduces Attack Surface

AppArmor, seccomp, USB Guard, AuditD, Rsyslog—restrict what compromised processes can do.

5 Secure OTA & Anti-Rollback

Cryptographically verified updates with downgrade attack protection

6 SBOM is Now a Regulatory Requirement

Automate SBOM generation. Maintain throughout lifecycle.

7 Map Every Control to Regulatory Requirements

FDA Section 524B, EU MDR, IEC 81001-5-1—document how each technical control satisfies regulatory expectations.

Remember: Security is a journey, not a destination. Start with the basics, automate relentlessly, and continuously improve.

Future Cybersecurity Trends 2026–2030

Emerging technologies reshaping medical device security

THE LINUX FOUNDATION
OPEN SOURCE SUMMIT INDIA

Zero Trust for IoMT

Principle: Never trust, always verify—assume breach

- Continuous verification of every device and user
- Microsegmentation of medical device networks
- Identity-based access control for IoMT devices
- Least privilege by default for all communications

Status: Early adoption in 2026, mainstream by 2028

AI-Driven Automated Defense

Capabilities: Machine learning for threat detection and response

- Behavioral analytics for anomaly detection
- Autonomous threat containment and response
- Reduced MTTD (Mean Time to Detect) from days to minutes
- Predictive vulnerability prioritization

Impact: 90-99% faster detection, 60% SOC cost reduction

Post-Quantum Cryptography

Urgency: Protect against "harvest now, decrypt later" attacks

- NIST PQC standards: ML-KEM, ML-DSA, SLH-DSA
- Hybrid modes: Combine classical + PQC algorithms
- Crypto-agility: Easy algorithm migration
- Timeline: Start pilots now, broad rollout by 2028-2030

Mandate: NSM-10 sets 2035 goal for quantum-proof encryption

Agentic AI & IAM

Evolution: AI-powered identity and access management

- Intelligent access control decisions
- Dynamic risk-based authentication
- Automated privilege management
- Context-aware authorization

Integration: With Zero Trust architectures

Looking Ahead: Cyber-Physical Systems (CPS) Security—integrating safety and security for life-critical medical devices

Convergence of safety & security

Thank You!

Questions & Discussion

Happy to Dive Deeper Into:

- Yocto security implementation details
- FDA submission experiences
- Secure Boot on specific SoCs
- OTA update strategies
- SBOM automation
- Regulatory compliance mapping
- Post-market surveillance
- Threat modeling methodologies

Senior Engineering Manager

GE HealthCare

20 years in medical device design, Yocto-based Embedded Linux, OS security

abraham.gogulamudi@gehealthcare.com

Abraham Gogulamudi | LinkedIn

opensourcesummit.in

Reference Links

1. 99% Of Healthcare Orgs Managing IoMT Devices with Known Exploited Vulnerabilities
2. More Than Half of Healthcare Orgs Attacked with Ransomware Last Year
3. Cost of a data breach: The healthcare industry | IBM
4. FBI Warns Healthcare Providers About Unpatched and Outdated Medical Device Risks
5. Healthcare Data Breaches 2024: A Stark Reality - Cyber Insurance News
6. Study Confirms Increase in Mortality Rate and Poorer Patient Outcomes After Cyberattacks
7. Ransomware Attack Contributed to Patient's Death: A Wake-Up Call for Health-Tech and Healthcare
8. The Status of Healthcare Ransomware Attacks in 2024 – ComplianceJunction
9. Healthcare Data Breaches 2024: A Stark Reality - Cyber Insurance News
10. Average Cost of a Data Breach 2026 | 46 Facts From IBM & Verizon
11. Cyber-Attacks on Hospital Systems: A Narrative Review – ScienceDirect
12. Cybersecurity in Medical Devices: Quality Management System Considerations and Content of Premarket Submissions
13. Threat modeling – An introduction