Verified Artifacts, Safer Patient Care

Securing Healthcare's Software Supply Chain



Copyright © 2026 JFrog Ltd.

www.jfrog.com

Table of Contents

1. Healthcare Has the Highest Average Breach Cost of Any Industry 3
2. Protecting Patient Data: Four Roles, Four Responsibilities 4
3. Block Risky Packages and Models at the Point of Entry 5
4. Verify Across the Clinical Software Supply Chain 6
5. Govern AI Assets the Same Way You Govern Other Artifacts 7
6. Regulatory Bars Keep Rising, Manual Evidence Doesn't Scale 8
7. Real Results from Healthcare and Life Sciences Customers 9
8. The Healthcare Technology Stack You Govern 10
9. Start Where Your Exposure Is Greatest 11
10. JFrog Healthcare & Life Sciences 12

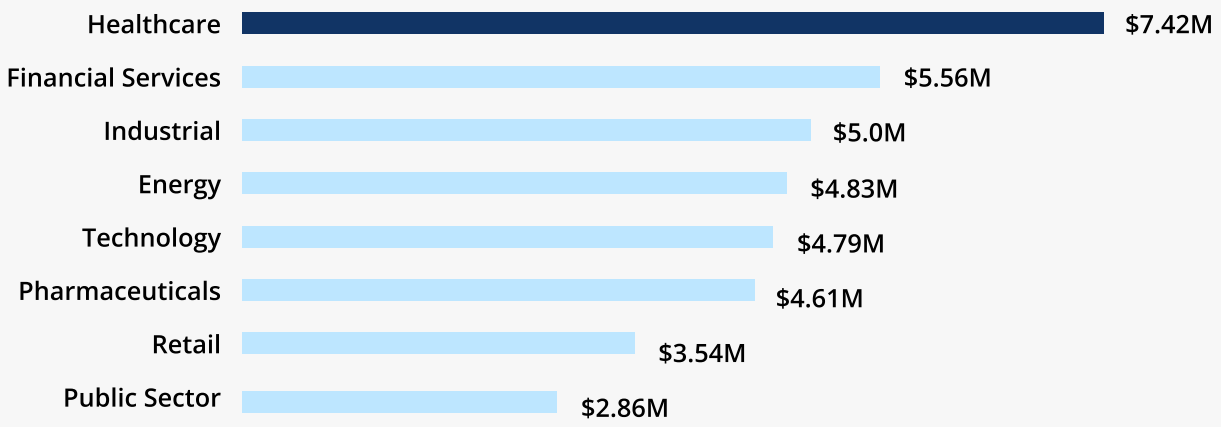
All rights reserved 2026 © JFrog Ltd.

THE STAKES

Healthcare Has the Highest Average Breach Cost of Any Industry

Healthcare organizations hold some of the most sensitive data there is: *medical histories, insurance details, and Social Security numbers, all tied to the patients you serve.* That makes your industry a natural target, and an expensive one to recover from. Healthcare data breaches cost an average of $6.64 million in 2026, 33% more than the all-industry global average of $4.99 million, and the highest of any industry for more than a decade. That total covers forensic investigation, regulatory notification, legal exposure, credit monitoring for affected patients, and often long periods of disrupted operations. The software behind your clinical systems, patient portals, and connected devices is part of that attack surface, which makes securing how it's built and delivered a core part of protecting patient data.

AVERAGE COST OF A DATA BREACH BY INDUSTRY (USD MILLIONS, IBM 2025)



Average cost of a data breach by industry, 2025 ($M). Source: IBM Cost of a Data Breach Report, 2025.

279 Days

average time to detect and contain a healthcare breach

(IBM, 2025)

27%

of healthcare organizations hit by ransomware reported higher patient mortality

(Proofpoint/Ponemon, 2025)

177K

new malicious packages detected across registries in 2025, with malicious npm packages up 451% YoY

JFrog Software Supply Chain Security State of the Union (2026)

JFrog is designed to be a single source of truth that helps healthcare software teams move at the speed AI demands while keeping patient safety in view.

All rights reserved 2026 © JFrog Ltd.

PROTECTING PATIENT DATA

Four Roles, Four Responsibilities

Ask four people in the same health system what "supply chain risk" means, and you'll get four different answers, because they're each carrying a different piece of the same exposure. The CISO sees the next unpatched CVE in a connected device or clinical integration. Engineering sees the next release deadline for a system a clinician is relying on that day. The Head of Regulatory & Quality sees the next FDA or HHS audit, and whether the evidence will hold up. The CIO sees the next headline, a ransomware attack that takes a health system offline and delays patient care. None of them can close the gap alone. In our view, this isn't only a scanning problem, it's a trust problem, and it is easier to solve when all four roles are working from the same picture of what's running in production and what components it contains.

CISO / VP Security

The Pain: Trust in artifacts is assumed, not verified. A signed package or a passed scan isn't proof it's safe.

The Answer: Compute trust at the binary level, continuously, so signed doesn't mean blindly trusted.

Director of Platform Engineering

The Pain: Six disconnected tools for storing binaries, catching vulnerabilities, vetting open-source packages, and shipping releases, each with its own blind spots.

The Answer: One control plane for your artifacts and AI models, from first commit to release.

Application Security Lead

The Pain: Alert volumes can bury teams, and many alerts point to code paths that may never execute once the software goes live.

The Answer: Binary-level contextual analysis that helps identify which vulnerabilities are likely exploitable in your application, so teams can focus on the alerts that matter.

Head of Regulatory Affairs & Quality Assurance

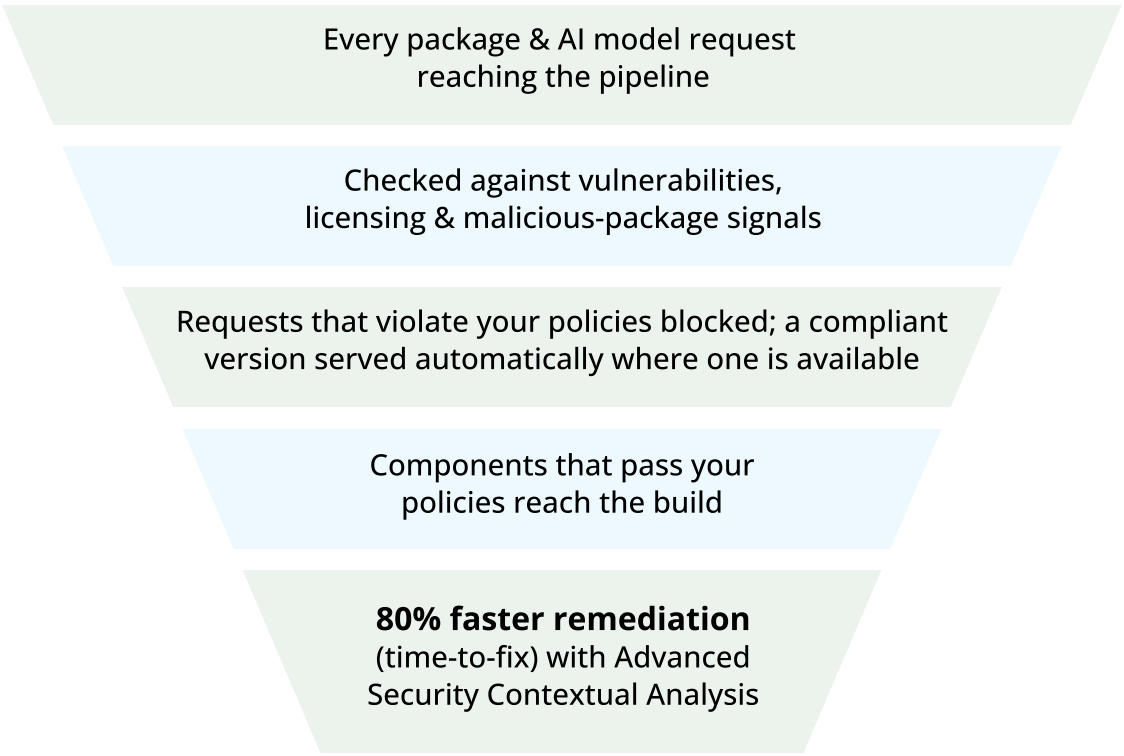
The Pain: SBOMs (software bills of materials) and audit evidence get assembled by hand, under deadline, for every submission and every audit.

The Answer: SBOMs (CycloneDX, SPDX), with VEX data (records stating whether a known flaw actually affects you), included in CycloneDX exports, generated from your builds and exportable on demand, not reconstructed after the fact.

All rights reserved 2026 © JFrog Ltd.

Block Risky Packages and Models at the Point of Entry

Attackers look for where healthcare's guard is down. JFrog research identified 495 malicious AI models on Hugging Face and 969 malicious agent skills carrying high-impact payloads. Scanning after the build tells you what already got in. Blocking risky components at entry, before they reach a developer's machine, adds control earlier.



Imagine knowing which vulnerabilities in your patient-facing systems are likely reachable, before your team spends another hour chasing the ones that aren't.

Treating every vulnerability the same, whether it's exploitable or buried in code your application never touches, has a cost. The result: *security teams drown in noise, patches pile up, and the one fix that actually matters sits in the same queue as a thousand that don't*. In healthcare, that queue can include the systems reading a diagnostic scan or tracking a patient's heartbeat, and every day a real vulnerability waits behind fake ones is a day it stays exploitable.

REAL RESULT: SHEBA MEDICAL CENTER

After adopting JFrog Curation and Advanced Security, Sheba identified 42 security risks and prevented 4 vulnerabilities across a governed base of 34,448 artifacts.

All rights reserved 2026 © JFrog Ltd.

Verify Across the Clinical Software Supply Chain

Preventative checks early in development **(shift left)** and continuous monitoring after release (shift right) both matter. Neither alone is enough for software that ends up at a patient's bedside.



A component that was safe when it shipped can still be exposed by a vulnerability (CVE) disclosed six months later. Runtime monitoring of Kubernetes-based clinical systems can help surface those gaps after release. Instead of a last-minute scramble, you can give auditors evidence on demand.

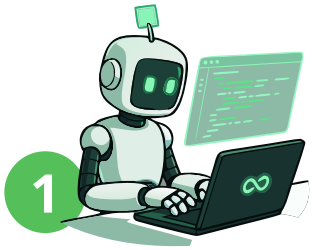
REAL RESULT: FFF ENTERPRISES

> We've seen a reduction of vulnerabilities in our cloud because we're handling them further up the chain by shifting left, so we get those vulnerabilities remediated before everything gets published. **— CISO, FFF Enterprises**

All rights reserved 2026 © JFrog Ltd.

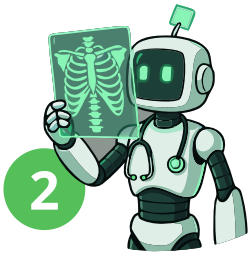
Govern AI Assets the Same Way You Govern Other Artifacts

A coding assistant, a diagnostic model, and an autonomous agent all enter clinical software through different doors. Each is an artifact: it needs provenance, version history, and a way to be recalled if it drifts.



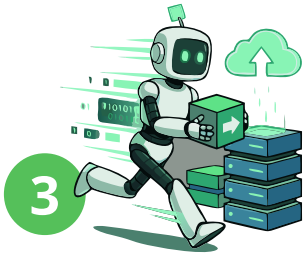
The Coding Assistant

AI-generated code without the same review as human-written code



The Diagnostic Model

A model influencing care with no version history or recall path



The Autonomous Agent

Agents shipping updates at machine speed faster than manual review

ONE GOVERNED GATE

curated, versioned & continuously monitored before critical use

JFrog helps your teams adopt AI at the pace the business demands, with governance over the models and packages they bring in, tracking where a model came from and verifying its security. Whether a model is fit for clinical use remains a regulated decision, made by your organization.

We aren't slowing AI adoption; we're refusing to trust the AI layer blindly. This approach helps surface shadow AI so you can address it before it turns into an incident.

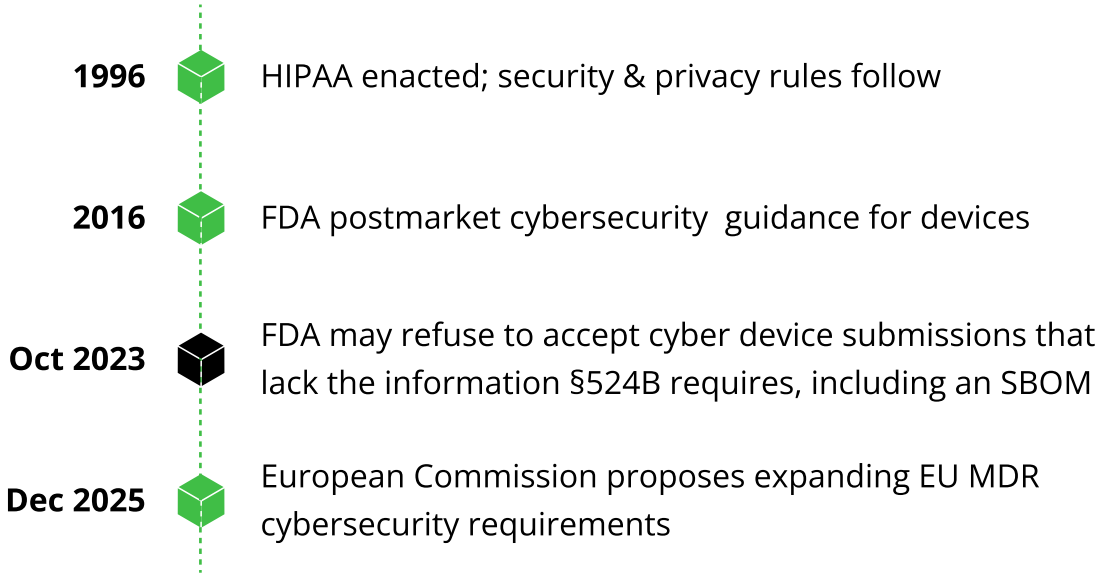
WHY NOW: Developers report that about 42% of their committed code is AI-generated or assisted, and among organizations that reported breaches involving AI models or applications, 97% lacked proper AI access controls. (Sonar 2026 / IBM 2025)

All rights reserved 2026 © JFrog Ltd.

Regulatory Bars Keep Rising, Manual Evidence Doesn't Scale

Each milestone below narrows the gap between “we think we're compliant” and “we can prove it, on demand.” Continuous, automated evidence can help you keep pace as requirements evolve.

How about a new state where every regulatory submission starts with the evidence already assembled, not a scramble to reconstruct it?



REAL RESULT: FFF ENTERPRISES

One SBOM per application to support regulatory compliance, with 1,500+ monthly software updates pushed to IoT devices in the field.

The Common Thread:
The frameworks above increasingly expect organizations to show what shipped and its security history. A machine-readable record of what shipped, who approved it, and its security history, kept current as builds change and exportable on demand, can help you avoid reconstructing it for an audit.

All rights reserved 2026 © JFrog Ltd.

PROOF

Real Results from Healthcare and Life Sciences Customers

Sheba Medical Center:

One of the world's leading hospitals, which needed to build an internal development group from scratch.

34,448 artifacts governed
42 security risks identified
4 vulnerabilities prevented

"With JFrog, we now have more security, control, and agility to better prioritize our engineering efforts, enabling us to deploy better software, faster."

FFF Enterprises:

A specialty pharmaceutical distributor, uses IoT inventory tracking devices to manage its supply chain.

1 SBOM per application for regulatory compliance
1,500+ monthly software updates to IoT devices
40 hrs/mo developer time shifted from security to coding

Leading Healthcare Company:

Petabytes of data, dozens of business units, and a six-month deadline to go all-in on SaaS.

40,000+ developers supported globally
100K+ repositories at scale
6 month target SaaS migration window

"With JFrog, we've been able to support tens of thousands of developers around the world without compromising speed or availability." — Senior Business Consultant, Engineering Team

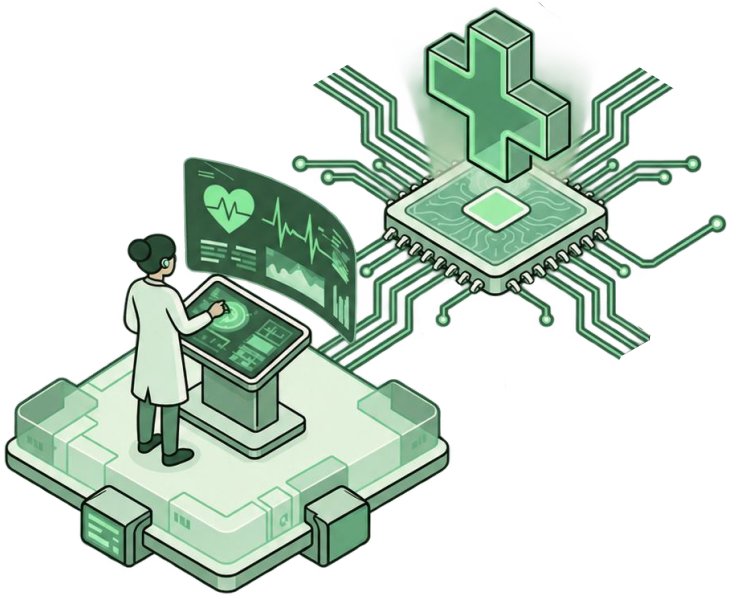
All rights reserved 2026 © JFrog Ltd.

UNDER THE HOOD

The Healthcare Technology Stack You Govern

Not every dependency carries the same weight. A vulnerable package's risk depends on where it runs, what data it can reach, and whether it's exposed externally. Treating every finding the same spreads effort evenly across very different risks. Here's where JFrog could fit into the specific systems, runtimes, and package ecosystems healthcare and life sciences teams actually run.

- **EHR & Clinical Platforms**: Teams building for Epic, Oracle Health (Cerner), and Meditech run real CI/CD pipelines to govern.
- **Medical Device / RTOS:** FreeRTOS and VxWorks are widely used embedded environments in FDA-regulated devices; Green Hills INTEGRITY is purpose-built and FDA-cleared for Class II and Class III devices. OTA update packages fit locked-down clinical networks well, since devices need only outbound HTTPS, no public IPs or inbound firewall rules required.
- **Package Ecosystems**: Python (NumPy, PyTorch, scikit-learn) for clinical AI/ML; R for biostats and trials; HL7 FHIR SDKs (Java, .NET, JavaScript) for interoperability; OCI/ Docker images for Kubernetes-based clinical platforms.
- **Regulatory-Adjacent Tech**: DICOM libraries for imaging pipelines, and HL7 v2/FHIR APIs, both common in EHR integrations and both real supply chain risk if an upstream library is compromised. AI/ML Stack: Models from the Hugging Face hub (clinical NLP), NVIDIA MONAI (medical imaging AI), and TensorFlow and PyTorch, governed through JFrog as versioned artifacts rather than pulled directly from public sources.



All rights reserved 2026 © JFrog Ltd.

NEXT STEPS

Start Where Your Exposure Is Greatest

Each role sees a different piece of the same problem: *knowing what your software depends on, and whether you can trust it*. You don't have to address all of it at once. Start with the area where your exposure is greatest, whether that's what enters your pipeline, which vulnerabilities are actually exploitable, or the AI models your teams are already using. Each step you take makes the next one easier.

What Should Change Now:

- Measure exposure, not volume. A thousand CVEs is a statistic. The handful reachable in production systems that touch patient data or clinical workflows is your actual risk.
- Move controls to the entry point. Controls that run after an artifact is already in your pipeline catch problems later; controls at the entry point catch them earlier.
- Treat models and agents as untrusted binaries. Verify where they came from, who approved them, and what data they can touch before they're promoted.
- Make compliance evidence a byproduct. With a trusted inventory and provenance for your components, SBOM and audit requests can become a report, not a project.

Where To Start:

- If you can't say what's in your pipeline, start at the gate. Route every package source through one controlled point, block malicious and policy-violating components, and build an inventory you can trust.
- If your backlog outpaces your team, start with exploitability. Test your highest-severity findings for reachability and runtime exposure. That number becomes your new baseline.
- If AI is already in production, start with governance. Inventory the models and agents in use, trace their origin, and require scanning, signing, and provenance checks before promotion.

Track one metric from day one: *the time between an exploitable vulnerability appearing and being fixed*. Every step you take should shorten it. Attackers aren't waiting for you to finish your roadmap, so begin with the area that carries the most risk, such as the packages entering your pipeline, the vulnerabilities that are actually exploitable, or the AI models already in production.

All rights reserved 2026 © JFrog Ltd.

About JFrog

JFrog Healthcare & Life Sciences

JFrog Ltd. ("JFrog") (NASDAQ: FROG), the creator of the software supply chain system of record, is on a mission to enable trusted software delivery in healthcare without friction, from development to production, whether software is written by developers or by AI. Driven by a "Liquid Software" vision, the JFrog Platform helps healthcare and life sciences organizations manage, secure, and govern the binaries, models, agents, and AI-generated assets behind electronic health record integrations, clinical applications, connected medical devices, and AI-driven diagnostics, while distributing trusted software with the speed, scale, and control that patient safety and regulatory compliance demand.

With built-in traceability and policy enforcement, the platform supports compliance with frameworks such as HIPAA and FDA software requirements, and helps protect sensitive patient data from supply chain threats. The universal JFrog Platform is available as a SaaS service across all major cloud providers, as well as via self-managed and hybrid deployments, giving healthcare organizations the flexibility to meet data residency and security requirements.

Millions of users and thousands of organizations worldwide, including a majority of the Fortune 100, depend on JFrog as a single source of truth to securely adopt AI and accelerate digital transformation in care delivery.

Verified artifacts.
Safer patient care.
Learn more here.



This document is provided for general informational purposes only. It is not legal, compliance, or security advice and does not guarantee any security outcome. Results depend on your environment and configuration. Product capabilities vary by edition and deployment model and are governed by your agreement with JFrog. JFrog does not author or warrant third-party AI assets listed in or governed through its products.

JFrog is a registered trademark of JFrog Ltd. in the United States. Epic, Oracle Health, Cerner, MEDITECH, FreeRTOS, VxWorks, Green Hills, INTEGRITY, Hugging Face, NVIDIA, MONAI, TensorFlow, PyTorch, NumPy, scikit-learn, HL7, FHIR, DICOM, Docker, Kubernetes, Java, JavaScript, .NET, Python, npm, IBM, Proofpoint, Ponemon Institute, and Sonar are trademarks of their respective owners. All other trademarks referenced in this document are the property of their respective owners. Use of these names does not imply any affiliation with or endorsement by them.

Third-party statistics are reported as published by their sources and have not been independently verified by JFrog.

All rights reserved 2026 © JFrog Ltd.