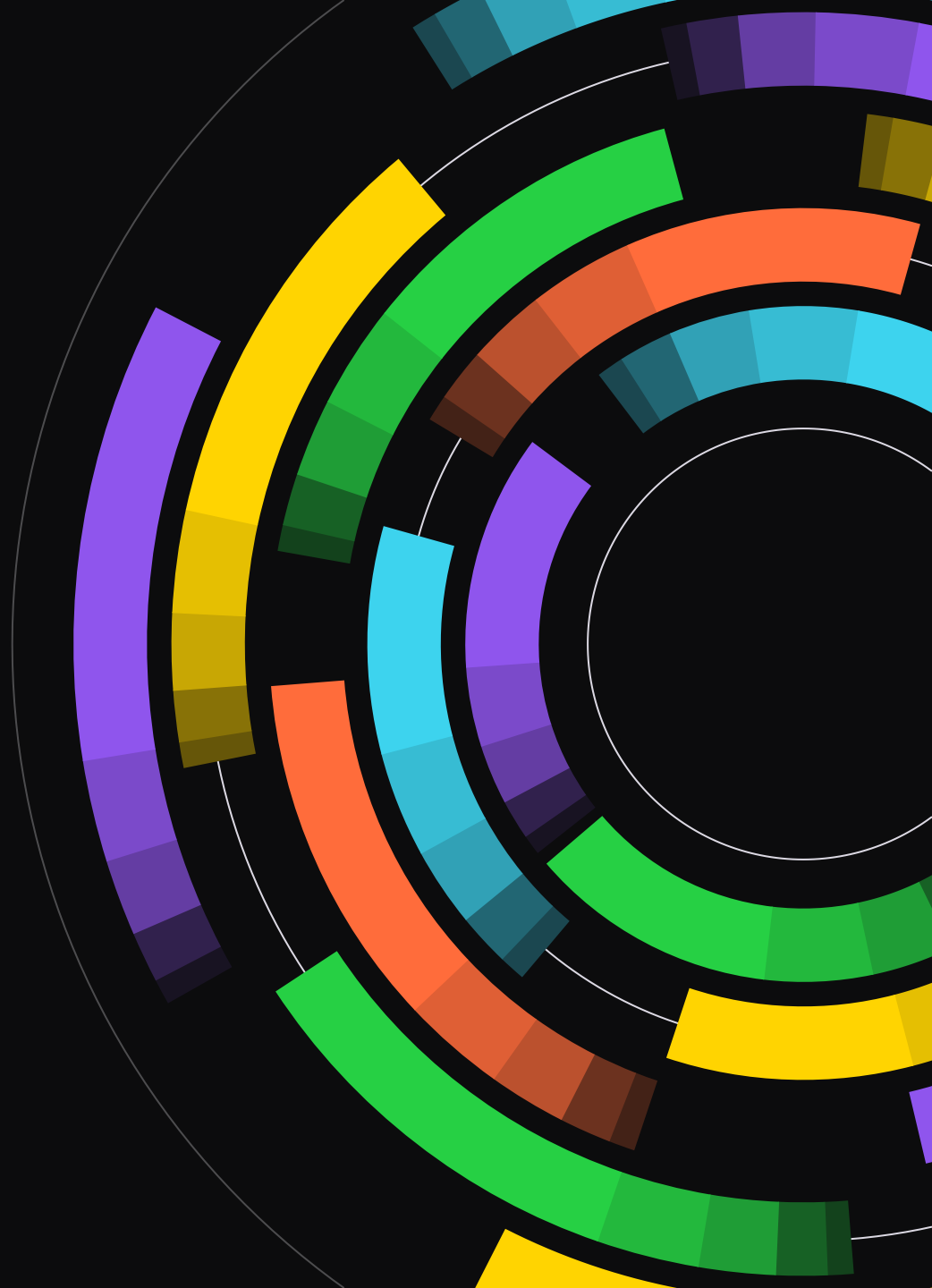
mariaDB

Curated Open Source

The Compliance Layer Regulators Assume Already Exists

Max Mether

MariaDB | OSFF NY



The Premise: Structural Reliance Meets Regulatory Reality

Financial services now rely on open source for critical operations - structurally, not experimentally.

Core Infrastructure

According to recent FINOS “State of Open Source in Financial Services reports”, open source software constitutes the core infrastructure of modern banking, moving far beyond edge tooling and sandbox environments.

The Compliance Gap

“Open source” downloaded by engineering communities is fundamentally different from “open source” required by a regulator.

The community provides innovation and velocity, but not compliance.

Unmanaged Risk

Treating community code as a fully managed vendor product without adding an accountability layer creates severe breaches, audit findings, and risks modern regulatory frameworks no longer tolerate.

Harder to Defend the Stack

Shifted Attack Surface

The reality of the modern software supply chain is that the attack surface has fundamentally shifted.

It is no longer just about securing your proprietary code; it is about securing everything your code depends on.

Real-World Examples

PHP2Shell (CVE-2026-109618):

Two flaws in PHP could let attackers run code remotely.

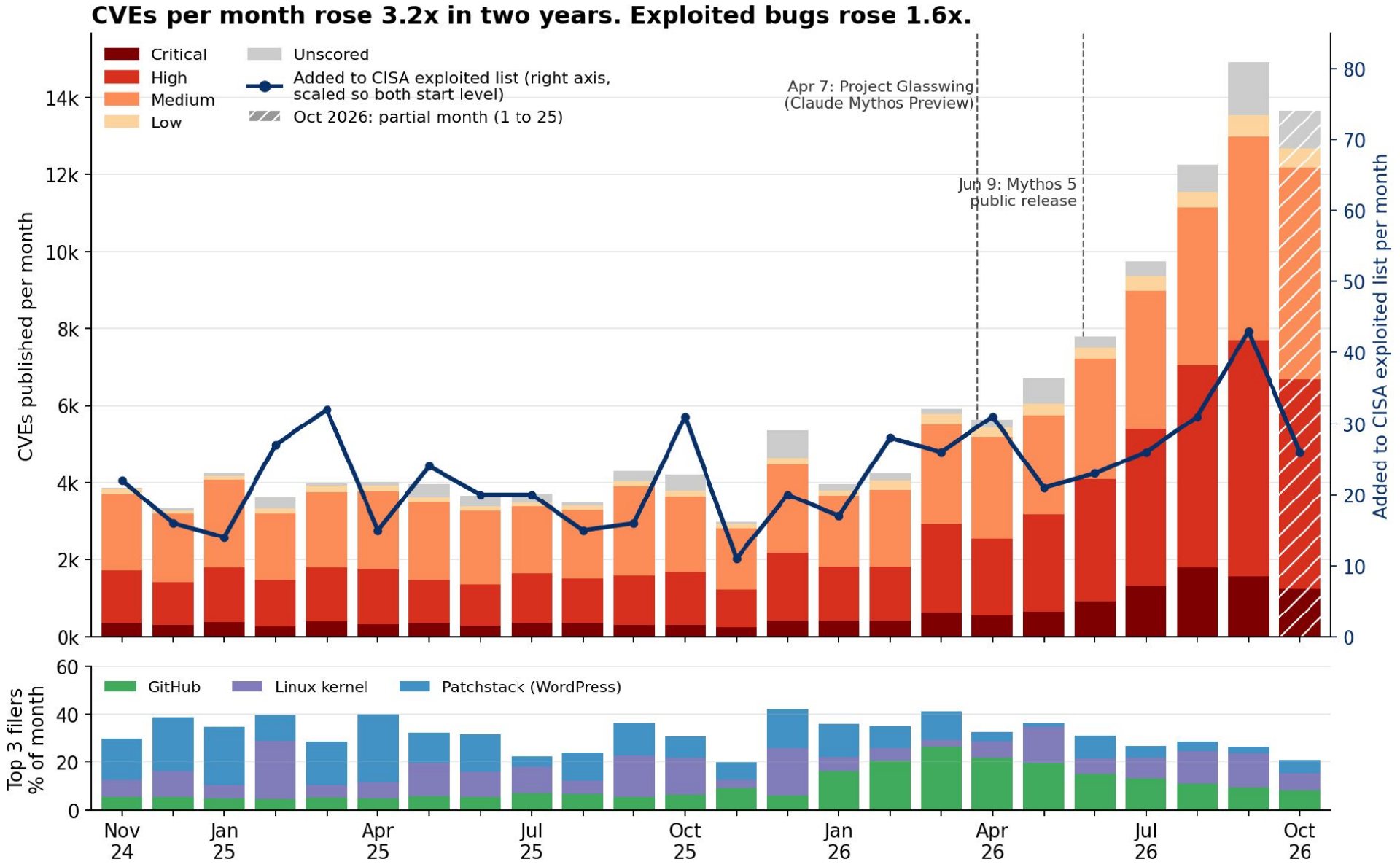
2024 XZ Utils (CVE-2024-3094):

A backdoor in a compression library targeted SSH access.

Transitive Dependency Risk

Your libraries rely on hundreds of others, bringing risks into your stack that are hard to track manually and can slip past perimeter defences.

You inherit the security posture of every unpaid maintainer in that dependency tree.



Source: CVE Program cvelistV5 (CVE records by publish date), CISA KEV catalog. Data through Oct 25, 2026. Severity: CNA score (CVSS 4.0 or 3.x), else CISA ADP score.

What Regulators Require

The regulatory floor is rising rapidly for dual-listed organizations, explicitly targeting open source software governance. Relying on the community to "fix it eventually" is no longer an acceptable defense during an audit.

EU DORA

ICT Risk Mgmt

In Force: Jan 2025

Mandate: Third-party ICT risk management explicitly extends to open source components embedded in the stack, not just paid commercial vendors.

EU CRA

Product Liability

Scope: Digital Elements

Mandate: Introduces product-liability-style obligations for software with digital elements, heavily impacting commercialized open source.

NYDFS 500

CISO & Patching

Region: US / New York

Mandate: Mandates named CISO accountability and enforces strict, non-negotiable patching cadences across all core architecture.

SEC RULES

4-Day Disclosure

Timeline: 4 Days

Mandate: Demands material incident disclosures within 4 days - inherently assuming real-time visibility into dependency provenance.

Regulatory Issues with Community OSS

mariaDB



No SLA on CVE Patching

Remediation timelines depend entirely on volunteer maintainer bandwidth, creating unpredictable security exposure.



Lacking Liability

Offers zero legal indemnification and lacks an accountable corporate entity to absorb enterprise liability.



Unverified Provenance

Rarely provides guaranteed, cryptographically signed SBOMs or provenance attestations for running binaries.



Maintainer “Bus-Factor”

Critical infrastructure relied on by enterprise systems is often maintained by a dangerously small pool of unpaid individuals (e.g., XZ Utils vulnerability).

Fundamental Governance Mismatch: This is not a critique of community maintainers, who do incredible work.

It highlights the gap between what volunteer governance can realistically deliver and what regulated financial entities must legally attest to under strict regulatory frameworks like DORA.

What “Curated” Open Source Actually Means

To bridge the gap between community innovation and regulatory compliance, organizations must adopt “Curated” Open Source. This model introduces five essential, non-negotiable pillars of accountability.

1

Provenance

Cryptographically signed builds that provide a verifiable, unbroken chain of custody from the original source code commit to the final deployed binary.

2

Patch SLA

Contractually committed remediation windows for Common Vulnerabilities and Exposures (CVEs), replacing best-effort community timelines with legally binding guarantees.

3

Long-Term Support (LTS)

Maintained branches that extend security patching far beyond the community End-of-Life (EOL) cycles, ensuring compliance and stability for legacy systems.

4

Guaranteed SBOM

A comprehensive Software Bill of Materials delivered as a standard artifact alongside the software, rather than something reverse-engineered post-deployment.

5

Vendor Accountability

A named, contactable, and legally liable vendor standing behind the code.

Summary: Curated open source retains the open license, transparency, and rapid innovation velocity of the community, while injecting the strict accountability layer that modern regulation assumes is already in place.

MariaDB as an Example

Community GPL Model

Core Engine: MariaDB Server remains a fundamentally community-developed project licensed strictly under the GPL.

Ecosystem Role: Serves as the upstream foundation driving rapid open source innovation, broad public adoption, and active community contributions.

Community Features: Some features of the Community builds stay experimental and never make it to Enterprise. Community Features do not have SLAs or vendor accountability

Rapid Innovations: New versions out every quarter

Enterprise Curated Builds

Curated Assurance: MariaDB Corporation provides Enterprise Server builds to meet stringent commercial and regulatory requirements.

Enterprise Features: Introduces defined patch SLAs, extended maintenance lifecycles, and dedicated enterprise technical support.

Vendor accountability: All features in the Enterprise binaries are maintained and curated. No community maintained features

Long time cycle: All release series maintained for 5+3 years

Alignment with FINOS Governance

This approach aligns perfectly with FINOS's own open source governance goals. It directly supports the "Open Source Readiness" mindset by ensuring that the consumption of open source software within financial institutions is done securely, sustainably, and in full compliance with regulatory mandates.

Evaluation Process for OSPOs

Five critical vendor assessment questions before open source enters a regulated financial stack

01. CVE Accountability

02. Binding Patch SLA

03. SBOM

04. “Bus-Factor” Governance

05. Risk Register Tracking

EVALUATION OUTCOME

If the answer to any of these is unclear, the component represents an unmitigated regulatory risk.

Regulation Is Forcing Accountability, Not Restricting Open Source.

mariaDB

Modern regulation is forcing the financial industry to be rigorously honest about which open source components possess a genuine accountability layer.

The era of implicitly trusting the community to handle enterprise risk management is over. Regulators expect critical infrastructure to have explicit provenance tracking and contractual bindings equal to proprietary software vendors.

For more information on how curated open source operates in practice, please visit the MariaDB Enterprise Server support policy page.

MariaDB Confidential

Continue the conversation

IMMEDIATE CALL TO ACTION

Audit your stack this week

Apply the five questions to your most critical data platforms.

Find where you’re relying on hope, not a contract.

Take the checklist with you:

Booth T10, Hall 2



mariaDB

Thank you

Max Mether, MariaDB

mariadb.com

