OpenEoX: Closing the AI/ML Lifecycle Visibility Gap

How Machine-Readable End-of-Life Data Transforms AI Infrastructure Governance

Unofficial AI-assisted Research

Cloud Security Alliance AI Safety Initiative

2026-03-16

Key Takeaways

- OpenEoX is an emerging OASIS Open standard that delivers machine-readable, JSON-structured lifecycle data — general availability, end of sales, end of security support, and end of life — for software, hardware, services, and AI models, closing a gap that SBOMs and vulnerability feeds alone cannot address.
- AI/ML infrastructure presents a distinct visibility problem: knowing which components are present does not reveal whether those components still receive security patches, and no standardized machine-readable feed has historically conveyed that information.
- CISA's February 2026 OpenEoX endorsement and Binding Operational Directive 26-02 signal regulatory momentum toward mandatory lifecycle tracking, with direct implications for AI infrastructure dependent on components such as TorchServe (archived August 2025) and JupyterLab 3 (upstream project maintenance ended May 2024).
- CVE-2025-32434, a CVSS 9.8 remote code execution flaw in PyTorch ≤ 2.5.1, illustrates the concrete risk: organizations unable to patch due to EOL version pinning face an exposure for which no vendor-supplied patch exists through the model loading pathway.
- Security teams should treat OpenEoX adoption as a near-term planning priority while immediately auditing AI/ML stacks against known end-of-security-support milestones using currently available community resources.

Background

The modern AI/ML infrastructure stack is a layered system of dependencies — GPU drivers, CUDA toolkits, ML frameworks, model serving platforms, notebook environments, and data pipeline tools — each with its own support lifecycle and its own vendor policies for issuing security patches. Unlike the enterprise software segment, where lifecycle dates for operating systems and commercial databases are routinely published, tracked, and integrated into vulnerability management workflows, AI/ML infrastructure has historically lacked any standardized mechanism for communicating when a component's security support ends. The result is a visibility gap that conventional security tooling cannot

close: a production AI system may be fully inventoried, fully vulnerability-scanned, and still leave security teams without an answer to the operationally decisive question of whether patches will arrive for the CVEs they are tracking.

OpenEoX is an OASIS Open international standard conceived precisely to address this gap. The Technical Committee was launched on December 14, 2023, co-chaired by Justin Murphy of CISA and Omar Santos of Cisco Systems. [1] Members include Cisco, Dell Technologies, Huawei, Microsoft, Qualys, Red Hat, and Sophos — a consortium spanning both infrastructure producers and security tooling vendors, reflecting the standard's intent to bridge the gap between lifecycle data at its source and security workflows that need to consume it. [2] The TC received the OASIS Open Cup Award in 2024, and published its Standardization Framework Technical Report on April 24, 2025. [2]

The standard defines a lightweight JSON schema conveying four lifecycle milestones per product: General Availability (the date support obligations commence), End of Sales (the final date a product can be purchased), End of Security Support (EoSSec — the threshold beyond which no security patches will be issued), and End of Life (the date all vendor support terminates). [2] Of these, EoSSec is the operationally critical milestone from a vulnerability management perspective. A component past its EoSSec date is one for which no future CVE will receive a vendor-supplied patch, regardless of severity. OpenEoX explicitly scopes AI models as a covered product category, meaning that the standard is designed to address lifecycle visibility not just for the infrastructure that runs AI systems but for the AI models themselves.

CISA publicly endorsed OpenEoX on February 13, 2026, framing end-of-support assets as a critical attack surface routinely exploited by nation-state adversaries and calling on technology producers to publish OpenEoX documents freely, without paywalls or access barriers. [3] This endorsement is closely coupled with Binding Operational Directive 26-02, issued February 5, 2026, which mandates that Federal Civilian Executive Branch agencies inventory all end-of-support edge devices within three months, decommission devices on CISA's formal EOS list by February 2027, and eliminate all remaining EOS edge devices from production by August 2027 — with continuous discovery processes required to be in place by February 2028. [4] While BOD 26-02 is scoped to network edge devices and binds federal agencies directly, the directive articulates a policy rationale that applies broadly: end-of-support components represent persistent, structurally unmitigable risk because no patch remediation path exists. Security teams in the private sector and cloud operators supporting federal customers should read BOD 26-02 as a signal of the direction regulatory expectations are moving.

The European regulatory environment reinforces this trajectory. ENISA published its NIS2 Technical Implementation Guidance on June 26, 2025, translating the NIS2 Implementing Regulation (EU) 2024/2690 into operational requirements across 13 thematic areas. [5] The guidance's patch management provisions require that patches be sourced from trusted origins, applied within a

reasonable time, and that deviations be documented with justification. Supply chain security requirements mandate that organizations maintain a current registry of direct suppliers and service providers, and that contracts specify vulnerability handling and secure decommissioning procedures. [5] Open-source software components, which form a substantial portion of many AI/ML stacks, must be assessed for risk, with organizations expected to evaluate whether OSS components remain actively maintained. For AI stacks running EOL frameworks, this provision creates a potential compliance challenge that may be difficult to satisfy, as the requirement to assess OSS maintenance status is structurally unmet by components whose maintenance has ceased. Organizations subject to NIS2 should consult legal counsel on their specific obligations and the acceptability of compensating controls in their jurisdiction.

Security Analysis

The Structural Visibility Problem

The AI/ML lifecycle visibility gap is a structural consequence of how the sector developed. Enterprise software vendors established lifecycle management conventions — extended support windows, public EOL announcements, and eventually machine-readable feeds — in response to regulatory pressure and enterprise procurement demands over decades. The AI/ML tooling ecosystem, much of it originating in academic research and open-source communities, developed without those procurement pressures. Lifecycle information for ML frameworks, model servers, and notebook environments exists, but it is scattered across GitHub release notes, project blog posts, and vendor support portal documents in inconsistent formats and without any standardized publication mechanism.

The practical consequence is that an SBOM for an AI system accurately answers what is present and what known CVEs affect those components, but leaves unanswered whether those CVEs will ever be patched. Trend Micro's TrendAI State of AI Security Report documented 2,130 AI-related CVEs in 2025 — a 34.6% year-over-year increase, and the highest annual share of total CVE volume on record. [6] ML framework CVEs grew 80.4% year-over-year. [6] Against this backdrop, TensorFlow alone had accumulated approximately 435 CVEs across its version history as of July 2024. [7] The question of which of those CVEs will receive patches — and which are permanently open because the affected version has crossed EoSSec — cannot be answered from SBOM data alone.

EOL Milestones Reached in 2024–2025

Several widely deployed AI/ML components crossed end-of-support thresholds in the 2024–2025 period without any machine-readable notification reaching the security tooling that might act on that information. TorchServe, the PyTorch model serving framework widely used in production ML deployments, was archived on August 7, 2025, transitioning to "Limited Maintenance" with no further security patches. [8] Organizations running production inference infrastructure on TorchServe now operate on an unsupported platform for which CVEs will accumulate without a vendor patch path. The JupyterLab Project ended maintenance for JupyterLab 3 on May 15, 2024, a transition that subsequently propagated through managed cloud platforms — including Amazon SageMaker, which set its own JupyterLab 1 and 3 platform EOL dates extending through mid-2025. [9] The breadth of JupyterLab's deployment in research and production notebook environments where data scientists interact with models and data makes this a consequential exposure regardless of which EOL date applies to a given organization's deployment context.

NVIDIA's CUDA support transitions illustrate a compound lifecycle problem specific to AI infrastructure. CUDA Toolkit 13.0 dropped support for Maxwell, Pascal, and Volta GPU architectures — hardware generations widely deployed in enterprise data centers during the mid-to-late 2010s. [10] An organization running Volta-class GPUs cannot access current CUDA security updates without hardware replacement. This creates a dependency chain where the EOL status of a GPU architecture forces downstream EOL status on the ML frameworks and drivers that require current CUDA for supported operation. Current security tooling does not widely provide automated cross-vendor lifecycle dependency tracing for AI infrastructure stacks; this typically requires manual analysis of interconnected lifecycle documents from multiple vendors — a capability gap that OpenEoX is designed to close once vendor adoption is sufficient.

CVE-2025-32434: The EOL Risk Made Concrete

CVE-2025-32434, published April 18, 2025, illustrates the specific risk that EOL AI components pose. The flaw carries a CVSS 3.1 score of 9.8 (Critical) and affects all versions of PyTorch through 2.5.1. [11] The vulnerability enables remote code execution through the `torch.load()` function even when the `weights_only=True` safety parameter is enabled — a parameter that had been widely recommended as a mitigation against unsafe deserialization. The practical effect is that any system loading AI models from external sources using affected PyTorch versions is exposed to arbitrary code execution at the point of model ingestion.

The remediation path requires upgrading to PyTorch 2.6.0 or later. For organizations whose AI stacks are pinned to earlier PyTorch versions due to compatibility constraints with deprecated CUDA versions or archived model serving frameworks, this may not be a simple patch operation. In these cases, upgrading the ML framework may require upgrading the CUDA toolkit, which may require GPU hardware replacement, which may require infrastructure capital expenditure that cannot be authorized on a security patch timeline. This scenario represents one end of the complexity spectrum: organizations running current CUDA on supported hardware, or using containerized environments with version flexibility, may be able to patch by updating the PyTorch package directly. But organizations at the constrained end of that spectrum face an unpatched CVSS 9.8 vulnerability that cannot be remediated through the normal patch process — precisely because the affected component sits at the intersection of multiple EOL dependency chains.

What OpenEoX Changes

OpenEoX's contribution is best understood as completing a four-stage vulnerability management pipeline that has been missing its final link. Asset inventory answers what hardware and systems exist. SBOM answers what software components are present. CSAF and VEX feeds answer what known vulnerabilities affect those components. OpenEoX answers whether security patches for those vulnerabilities will ever arrive from the vendor. [2] Embedded in or linked from an SBOM, an OpenEoX record for a component makes the EoSSec date machine-readable by vulnerability scanners, asset management platforms, and security dashboards that can then automatically classify components as patching-ineligible rather than merely unpatched.

Microsoft, a founding TC member, articulated the AI-specific dimension directly: "Standardizing how end of life, end of security support, and end of sales is handled for hardware and software makes the software supply chain more secure and efficient. This is especially important for developing and deploying AI systems securely." [2] The explicit inclusion of AI models in the standard's scope reflects recognition that the lifecycle visibility problem extends from the infrastructure layer — frameworks, serving platforms, GPU drivers — into the model layer itself. A model trained on a deprecated framework version and served by an archived serving platform may itself carry implicit lifecycle risk that is not visible from the model artifact alone.

CISA's guidance specifies that OpenEoX documents should be published publicly and without access barriers, enabling automated consumption by security tooling. [3] Major vendor feed availability is projected for Q4 2026, with full OASIS standard ratification anticipated in 2027. [2] This timeline gives organizations a defined window to build the internal processes needed to act on OpenEoX data before it becomes broadly automated.

Recommendations

Immediate Actions

Security teams should immediately audit AI/ML infrastructure for components with known EOL dates, starting with the milestones reached in recent years. TorchServe (EoSSec August 7, 2025), JupyterLab 3 (upstream project maintenance ended May 15, 2024), and NVIDIA CUDA support for Maxwell, Pascal, and Volta GPU architectures are confirmed EOL components that may be present in production AI environments without security team awareness. CVE-2025-32434 (CVSS 9.8) warrants the most urgent attention: any production workload running PyTorch 2.5.1 or earlier should be treated as a high-priority remediation target given the active exploitability of the `torch.load()` vulnerability. EOL infrastructure components without active CVEs should also be flagged for remediation planning, because the absence of known CVEs today does not prevent discovery of critical CVEs tomorrow against a platform that will not receive patches.

For all PyTorch deployments, organizations should verify immediately that instances have been upgraded to version 2.6.0 or later to mitigate CVE-2025-32434. [11] Version pinning constraints that hold AI stacks back from this upgrade should be documented as accepted risk with compensating controls in place, not left as undiscovered debt.

Short-Term Mitigations

Until OpenEoX feeds are broadly available from major AI/ML vendors, security teams should extend their SBOM practice to include lifecycle status as a manually tracked attribute. The endoflife.date community project currently tracks approximately 445 products with a publicly accessible REST API, and provides lifecycle data for Python, NVIDIA drivers, and other AI-adjacent infrastructure components. [12] For components not covered by endoflife.date, lifecycle status should be verified against vendor documentation and GitHub repository status as part of the SBOM review cycle, with findings recorded in the asset management system.

Organizations subject to NIS2 obligations should document their approach to OSS risk assessment in a manner that addresses component maintenance status explicitly. The ENISA guidance's requirement for current supplier registries and contract provisions covering decommissioning applies to AI/ML tool vendors and should be reflected in existing vendor risk management processes. [5] AI platform vendor assessments should include questions about published lifecycle dates and the vendor's notification practices for EOL transitions.

Network isolation and access controls for systems running EOL AI components serve as compensating controls during transition periods. Restricting which systems can initiate connections to model serving endpoints running archived TorchServe meaningfully limits the blast radius of exploitation, even when the underlying vulnerability cannot be patched. These controls should be documented as time-limited mitigations with explicit review dates tied to remediation milestones.

Strategic Considerations

Organizations should begin planning now for OpenEoX integration into their vulnerability management and asset management tooling. CISA's guidance urging vulnerability scanner vendors to support OpenEoX consumption [3] suggests that tool-level integration will arrive before the 2027 ratification date, and organizations that have not built the internal processes for lifecycle-aware vulnerability triage will be unprepared to act on the data when it becomes available. The investment required is primarily process design — defining how EoSSec dates surface in risk dashboards, who owns remediation decisions for EOL AI components, and what the default escalation path is when EOL infrastructure cannot be immediately replaced.

Supply chain risk management programs should explicitly scope AI/ML infrastructure components alongside traditional enterprise software in vendor lifecycle tracking. AI platform providers — including cloud-based ML services, model hosting platforms, and MLOps tooling vendors — should be expected to publish lifecycle dates for their products and to provide advance notice of EoSSec transitions. Organizations that make this expectation explicit in vendor contracts and procurement criteria will be better positioned as OpenEoX adoption creates market differentiation between vendors who publish machine-readable lifecycle data and those who do not.

CSA Resource Alignment

This research note connects to multiple CSA frameworks addressing the governance and risk dimensions of AI/ML infrastructure lifecycle management. The CSA AI Model Risk Management Framework establishes a four-pillar structure encompassing model cards, data sheets, risk cards, and scenario planning, with explicit attention to model lifecycle documentation, MLOps practices, and NIST AI RMF alignment. [13] The EOL visibility gap described here is a specific realization of infrastructure lifecycle risk within that broader model risk domain: security decisions about model operation cannot be made independently of lifecycle decisions about the infrastructure layer on which the model runs. When the serving platform is EOL, the model's risk profile changes in ways that a model card alone does not capture.

CSA's AI Organizational Responsibilities guidance on AI Tools and Applications addresses third-party AI supply chain management, including the integration of SBOM and SLSA frameworks into AI tool vendor assessment processes. [14] OpenEoX fits naturally within this framework as an additional data element in vendor qualification — lifecycle date publication should be a standard expectation in AI platform vendor questionnaires alongside existing SBOM and vulnerability disclosure requirements. The guidance's emphasis on Operational Qualification and Performance Qualification for AI tool deployments should be extended to include lifecycle qualification: documenting that a tool's current version is within its security support window.

The MAESTRO framework for agentic AI threat modeling identifies the infrastructure execution environment as a distinct threat surface requiring explicit threat modeling. EOL AI/ML infrastructure components are a concrete realization of this concern: an agentic system running on an archived model serving framework or an EOL notebook environment is operating in an execution environment with permanently unmitigable vulnerabilities. MAESTRO-based threat assessments for agentic AI deployments should include infrastructure lifecycle status as a standard threat modeling input.

Within the Cloud Controls Matrix, the Supply Chain Management (STA) and Change Control and Configuration Management (CCC) control domains provide the most direct structural alignment. CCC controls governing configuration baseline management and vulnerability remediation tracking should explicitly address EOL component identification and define remediation timelines. STA controls governing third-party security requirements should include lifecycle data disclosure — specifically the publication of machine-readable EoSSec dates in formats compatible with OpenEoX — as a vendor security expectation that organizations can reference in supplier assessment and contracting.

References

[1] OASIS Open. "OASIS Launches OpenEoX Technical Committee." December 14, 2023. https://www.oasis-open.org/2023/12/14/oasis-launches-openeox/

[2] OASIS OpenEoX Technical Committee. "OpenEoX Standardization Framework Technical Report." April 24, 2025. https://docs.oasis-open.org/openeox/standardization-framework/openeox-standardization-framework-technical-report.pdf

[3] CISA. "The End is Just the Beginning of Better Security: Enhanced Vulnerability Management with OpenEoX." February 13, 2026. https://www.cisa.gov/news-events/news/end-just-beginning-better-security-enhanced-vulnerability-management-openeox

[4] CISA. "Binding Operational Directive 26-02: Mitigating Risk From End-of-Support Edge Devices." February 5, 2026. https://www.cisa.gov/news-events/directives/bod-26-02-mitigating-risk-end-support-edge-devices

[5] ENISA. "Technical Implementation Guidance on Cybersecurity Risk Management Measures, Version 1.0." June 26, 2025. https://www.enisa.europa.eu/publications/nis2-technical-implementation-guidance

[6] Trend Micro. "Fault Lines in the AI Ecosystem: TrendAI State of AI Security Report." 2025. https://www.trendmicro.com/vinfo/us/security/news/threat-landscape/fault-lines-in-the-ai-ecosystem-trendai-state-of-ai-security-report

[7] NIST National Vulnerability Database. "TensorFlow CVE Records." Queried July 2024. https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&query=tensorflow&search_type=all

[8] PyTorch. "pytorch/serve repository archived." GitHub. August 7, 2025. https://github.com/pytorch/serve

[9] Jupyter Blog. "JupyterLab 3 — End of Maintenance." May 15, 2024. https://blog.jupyter.org/jupyterlab-3-end-of-maintenance-879778927db2

[10] NVIDIA. "CUDA Toolkit Release Notes." Accessed March 2026; Maxwell, Pascal, and Volta architecture deprecation noted in CUDA Toolkit 13.0. https://docs.nvidia.com/cuda/cuda-toolkit-release-notes/

[11] NIST National Vulnerability Database. "CVE-2025-32434: PyTorch Remote Code Execution." Published April 18, 2025. https://nvd.nist.gov/vuln/detail/CVE-2025-32434

[12] endoflife.date. "Open Source Community EOL Tracking Project." https://endoflife.date/

[13] CSA AI Technology and Risk Working Group. "AI Model Risk Management Framework." 2024. https://cloudsecurityalliance.org/artifacts/ai-model-risk-management-framework

[14] CSA AI Organizational Responsibility Working Group. "AI Organizational Responsibilities: AI Tools and Applications." 2025. https://cloudsecurityalliance.org/artifacts/ai-organizational-responsibilities-ai-tools-and-applications