International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Enterprise AI Supply Chain Security Governance: A Framework for Secure Open-Source AI Model Adoption Using an Artificial Intelligence Bill of Materials (AI-BOM)

Sandeep Kumar Anuguthala[1], Harish Namani[2]

[1,2]Independent Researcher
anuguthalasandeepkumar@gmail.com, namaniharish35@gmail.com

Abstract:
The rapid availability of open-source AI models accelerates enterprise innovation but introduces supply-chain risks — malicious serialized artifacts, vulnerable dependencies, and licensing violations — that existing SBOM frameworks and emerging AI-specific schema standards (CycloneDX ML-BOM, SPDX 3.0 AI profile) address only at the data-format level, not at the operational governance level. This paper introduces the Artificial Intelligence Bill of Materials (AI-BOM), an eight-category schema, integrated within the Enterprise AI Supply Chain Security Architecture (E-AISCSA): a ten-layer governance framework with a semi-quantitative Weighted Risk Score (WRS = 0.35×AR + 0.30×SR + 0.25×DR + 0.10×PR). Evaluation across ten Hugging Face NLP models confirmed framework feasibility: all models used pickle-serialized weights, all carried CVE-2024-3568 (CVSS 9.6 Critical) in their transformers dependency, one model was rejected as High Risk (WRS 5.60), and nine received conditional Medium-Risk approval. The framework aligns with NIST AI RMF 1.0, ISO/IEC 42001:2023, and EU AI Act requirements. Raw experimental data are available from the authors upon request.

Keywords: AI supply chain security; AI-BOM; model governance; MLOps; enterprise AI security; open-source AI models; serialization vulnerabilities; provenance risk; weighted risk scoring.

1. Introduction

Hugging Face now hosts over two million publicly available models [4], reducing AI time-to-deployment from months to days. However, externally sourced models introduce supply-chain risks distinct from traditional software: serialized weight files can carry executable payloads [5][6], weaponized PyTorch models have been documented on public repositories [7][8], and framework dependencies may carry critical CVEs [9]. SBOM frameworks improve component transparency [12] but were not designed for ML-specific artifacts such as model weights or training dataset provenance.

CycloneDX ML-BOM [23] and SPDX 3.0 AI profile [24] extend SBOM to ML documentation but prescribe only what data to record, not how an enterprise should operationalize that data within security infrastructure — no existing standard addresses sandbox procurement, automated scanning pipelines, semi-quantitative risk scoring, or governance approval workflows. SLSA [25] and in-toto [26] address build-time software integrity but have not been adapted for ML procurement. This gap motivates the present work.

1.1 Contributions

1. **AI-BOM Specification:** *Eight-category schema extending SBOM to ML artifacts with SHA-256 hashing, aligned with CycloneDX ML-BOM and SPDX 3.0 AI profile, adding three security-operational fields absent from both standards.*

2. **E-AISCSA Architecture:** *Ten-layer governance framework integrating sandbox validation, artifact scanning, dependency analysis, risk scoring, and continuous monitoring.*
3. **Secure Procurement Workflow:** *End-to-end process with role-based controls and regulatory audit trails from model request to repository publication.*
4. **WRS Framework:** *Semi-quantitative weighted scoring model (WRS = 0.35×AR + 0.30×SR + 0.25×DR + 0.10×PR) with operationalized rubrics and governance thresholds.*
5. **Experimental Validation:** *Ten Hugging Face models evaluated across four risk dimensions with reproducible WRS computation and provenance assessment.*
6. **Regulatory Alignment:** *Systematic mapping to NIST AI RMF 1.0, ISO/IEC 42001:2023, and EU AI Act.*

2. Background and Related Work

SBOM frameworks — standardized by NTIA [12] and mandated for US federal software by EO 14028 [11] — provide component-level transparency but were not designed for ML artifacts. NIST SP 800-218 [16] recommends SBOM as foundational security practice. Model cards [13] improve transparency on intent and limitations but omit artifact security; MLOps frameworks [14] manage deployment efficiency rather than supply-chain security; open-source supply chain vulnerabilities are pervasive across enterprise software [15]; dataset provenance work [21] establishes traceability records without cryptographic verification or CVE monitoring. The MITRE ATLAS framework [17] catalogues ML-specific attacks: pickle deserialization enabling arbitrary code execution [5][6] (≈60% of Hugging Face models use pickle [7]); data poisoning creating behavioral backdoors [18]; transitive dependency CVEs in deep learning frameworks [8][9][19]; and prompt injection [10][20].

CycloneDX ML-BOM [23] and SPDX 3.0 [24] address what fields to record for ML models. This paper addresses the orthogonal problem of how those records are generated, acted upon, and maintained within a security governance architecture. Table 1 shows the three security-operational fields unique to AI-BOM not present in either standard.

Table 1: AI-BOM Differentiation from Existing Standards. ✓ = capability present; — = absent; fields marked (new) are unique to AI-BOM and absent from both CycloneDX ML-BOM v1.5 and SPDX 3.0 AI Profile.

Capability / Field	CycloneDX ML-BOM v1.5	SPDX 3.0 AI Profile	AI-BOM (this work)
Training dataset metadata	✓	✓	✓
Model card / documentation	✓	✓	✓
SHA-256 artifact hashing	✓	Partial	✓
Serialization format classification	—	—	✓ (new)
Sandbox validation status	—	—	✓ (new)
Weighted Risk Score (WRS)	—	—	✓ (new)
Governance approval record	—	—	✓ (new)
Prescribes governance workflow	—	—	✓ (new)

3. Artificial Intelligence Bill of Materials (AI-BOM)

The AI-BOM extends SBOM to ML systems by documenting the complete composition of an AI model with cryptographic verification and security-operational metadata. Records are generated through automated artifact extraction — models downloaded in isolated sandboxes, SHA-256 hashes computed, dependencies extracted, and security scans executed — then stored as authoritative compliance records. The schema is specified as a versioned JSON document (ai_bom_version 1.1) compatible with CycloneDX ML-BOM component extensions and SPDX AI profile fields, adding the three security-operational fields shown in Table 1. Table 2 describes the eight schema categories.

Table 2: AI-BOM Schema — Eight Categories and Governance Purpose

Category	Key Elements	Governance Purpose
Model Metadata	Name, version, author, repository URL, license, framework, task type	Identity and license compliance tracking
Architecture Details	Model type, parameter count, layer configuration, config file hash	Integrity verification of model configuration
Model Weights	File paths, sizes, SHA-256 hashes, serialization format classification	Artifact integrity and serialization risk detection
Tokenizer Artifacts	Tokenizer type, vocabulary file, special tokens, cryptographic hashes	Integrity of inference components
Dataset Provenance	Training dataset names, source URLs, license, documentation completeness score	Provenance risk assessment and regulatory traceability
Software Dependencies	Pinned dependencies, system libraries, SBOM cross-references	Dependency CVE scanning and tracking
Security Metadata	Malware scan result, CVEs with CVSS, serialization classification, WRS	Risk scoring inputs and governance decision record
Governance Record	Request ID, approver, approval date, classification tier, monitoring schedule	Audit trail for regulatory compliance

4. Enterprise AI Supply Chain Security Architecture (E-AISCSA)

E-AISCSA operationalizes AI-BOM governance through ten sequential security layers between external model sources and enterprise consumption. Unlike standard MLOps pipelines, E-AISCSA (1) mandates acquisition and inspection in a network-isolated sandbox before any execution; (2) gates registry promotion on a complete AI-BOM record and computed WRS; and (3) produces auditable evidence aligned with NIST AI RMF, ISO/IEC 42001, and EU AI Act expectations. Table 3 summarizes the layers; Figure 1 illustrates the full flow.

Table 3: E-AISCSA — Ten Layers, Descriptions, and Responsible Teams

#	Layer	Description	Team
1	**External Source**	Open repositories: Hugging Face, GitHub, model hubs	External
2	**Model Request**	Formal request with business justification and intended environment	Data Scientists
3	**Governance Intake**	Policy review; registry check for prior approval	AI Governance
4	**Sandbox Download**	Network-isolated ephemeral environment; full audit logging	Cybersecurity

5	**Artifact Scanning**	Malware detection, serialization opcode analysis, hash integrity	Cybersecurity
6	**Dependency Analysis**	CVE cross-reference (NVD+OSV), license review, transitive deps	Cyber + Legal
7	**Risk Scoring**	WRS computation from AI-BOM security metadata	AI Governance
8	**Governance Approval**	Final classification; compensating controls if Medium risk	AI Governance
9	**Repository Publication**	Publish with AI-BOM, access controls, version tag	MLOps Platform
10	**Continuous Monitoring**	Ongoing CVE scanning, signature updates, periodic re-review	Cybersecurity

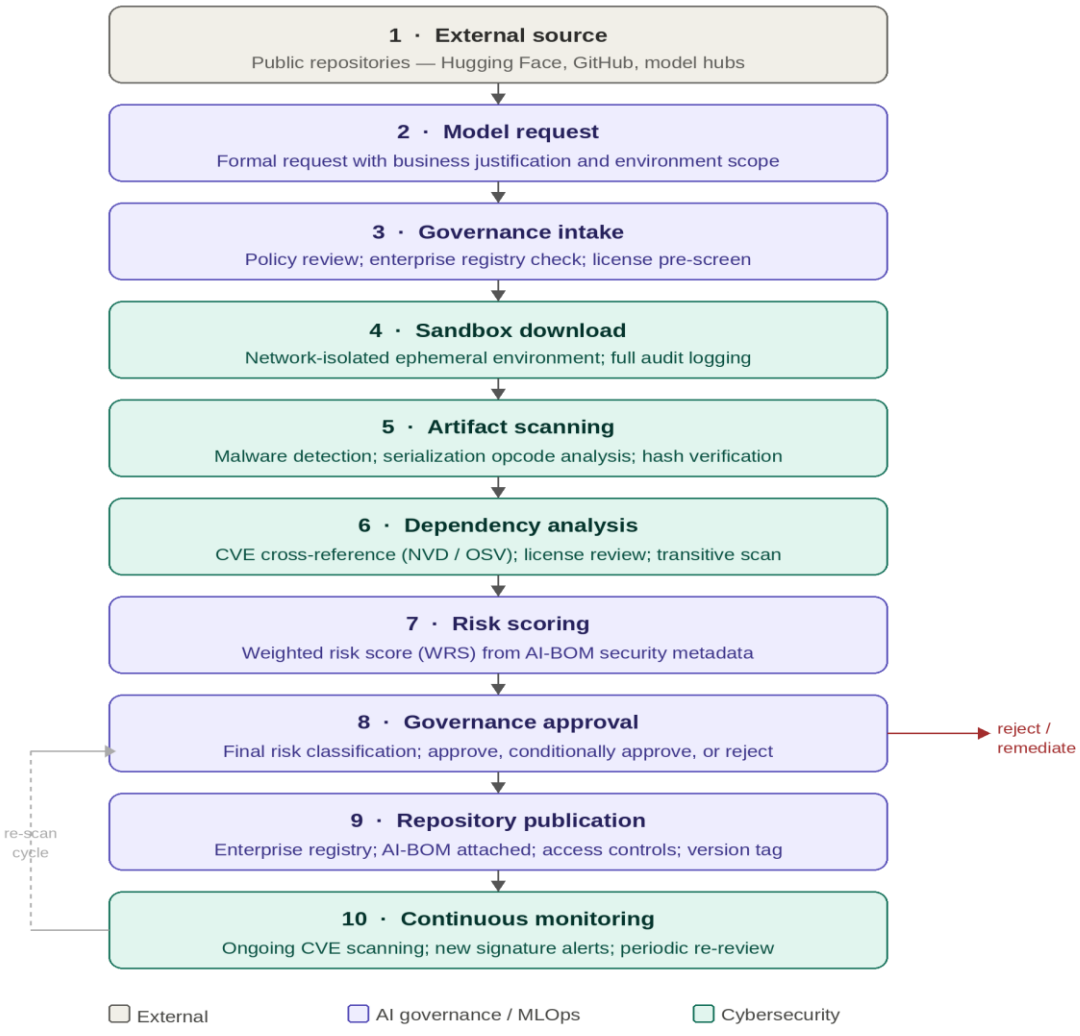

Figure 1: E-AISCSA ten-layer defense-in-depth architecture.

4.1 Regulatory Alignment

E-AISCSA is designed to support key expectations from NIST AI RMF 1.0, ISO/IEC 42001:2023, and EU AI Act for managing third-party and supply-chain risks [1][2][3].

Table 4: E-AISCSA Regulatory Alignment Matrix

Framework	Obligation (Simplified)	Supporting E-AISCSA Layers
NIST AI RMF (MAP)	Enumerate and evaluate third-party AI risks	Layers 3, 5, 6, 7
NIST AI RMF (MANAGE)	Monitor and respond to AI risk changes over time	Layer 10; AI-BOM update triggers
ISO/IEC 42001 (Clause 8)	AI-specific supply chain risk assessment	Layers 4, 5, 6, 7
ISO/IEC 42001 (Clause 9)	Documented controls and role responsibilities	Roles defined across all ten layers; records Layers 7–9
EU AI Act (Arts. 11, 18)	Technical documentation and traceability for high-risk AI	AI-BOM records; Layer 9; Layer 8 audit trails
EU AI Act (Art. 72)	Post-market monitoring obligations	Layer 10 monitoring, alerting, model revocation workflows

5. Secure Model Procurement and Governance Workflow

The E-AISCSA workflow begins with a formal model request (Layer 2) capturing identity, business justification, intended environment, and timeline. The AI Governance Team reviews compliance and registry for prior approval (Layer 3). New models proceed to a network-isolated sandbox for download with full logging (Layer 4). All artifacts then undergo four-stage inspection (Layer 5): (1) ClamAV malware signature scanning; (2) serialization analysis via PickleScan opcode scanning and Fickling AST decompilation; (3) dependency CVE scanning via pip-audit against NVD and OSV; and (4) SHA-256 hash integrity verification. A pip-licenses audit and legal review follow (Layer 6). The WRS is computed (Layer 7) and drives governance approval (Layer 8): Low-risk approved; Medium-risk receive conditional approval with documented compensating controls and 90-day re-review; High-risk rejected. Approved models are published with attached AI-BOM records and immutable version tags (Layer 9), then subjected to continuous CVE monitoring (Layer 10).

6. AI Model Risk Scoring Framework

6.1 Weighted Risk Score

Each model is assessed across four risk dimensions scored 0–10. Weights reflect relative threat severity and sum to 1.0 (WRS ∈ [0, 10]): artifact payloads are the most direct execution threat (0.35); unsafe serialization enables code execution at model-load time (0.30); dependency vulnerabilities require an additional exploitation step (0.25); provenance gaps introduce uncertainty rather than confirmed threats (0.10).Weights were selected heuristically based on enterprise security risk assessment experience and were not tuned on the evaluation dataset.

$$WRS = 0.35 \times AR + 0.30 \times SR + 0.25 \times DR + 0.10 \times PR$$

6.2 Component Scoring Rubrics

Table 5: Component Scoring Rubric for WRS Computation.

Score	AR — Artifact Risk	SR — Serialization Risk	DR — Dependency Risk	PR — Provenance Risk
0	No anomalies: hashes verified	Safe format (safetensors/ONNX)	No known CVEs	Full card (≥500 w); dataset named, licensed, verified authorship
1–3	Minor anomalies; no payload	Pickle; 0 GLOBAL imports; is_likely_safe=true	Low CVEs (CVSS < 4.0)	Partial card (100–499 w); dataset named but not linked
4–6	Suspicious patterns; embedded scripts	Pickle; non-standard torch imports; CAUTION	Medium CVEs (CVSS 4.0–6.9)	Sparse card (<100 w); vague dataset; unverified author
7–9	High-confidence malicious indicators	Pickle; GLOBAL/REDUCE → system modules	High CVEs (CVSS 7.0–8.9)	No card; no dataset named; anonymous authorship
10	Confirmed exploit code	Confirmed payload; demonstrated execution	Critical CVEs (CVSS ≥ 9.0)	No documentation; fraudulent provenance

6.3 Assessment Methodology

AR: ClamAV recursive scan (freshclam-updated) for malware; Python hashlib SHA-256 computed for all weight files and compared against author-published hashes. AR=0 when ClamAV returns Infected files: 0 and hashes match (or are unverifiable due to no published hash). **SR:** ModelScan classifies weight format (safetensors → SR=0; pickle → proceed). PickleScan records GLOBAL imports (torch.* expected; os/subprocess suspicious). Fickling [22] --json produces is_likely_safe, suspicious_nodes, high_severity_nodes, and SAFE/CAUTION/UNSAFE verdict mapping to SR per Table 5. Models with is_likely_safe=true and num_nodes=3 — indicating a standard long-integer opcode structure with zero GLOBAL imports — are assigned SR=2 within the 1–3 rubric range. **DR:** pip-audit queries NVD and OSV simultaneously; worst-case CVSS base score maps to DR per Table 5. pip-licenses audits copyleft obligations independently of DR score. **PR:** Structured five-criterion checklist applied to Hugging Face Hub API metadata, model card content, and DatasetCard.load() verification.

Table 6: Provenance Risk Scoring Checklist (C1–C5). PR = C1+C2+C3+C4+C5 ∈ [0,10]. Score 0 = documentation present (low risk); Score 2 = absent (high risk). Tiers: Good (0–2), Partial (3–5), Minimal (6–7), Absent (8–10).

Criterion	What is Assessed	Score 0 — Low Risk (documentation present)	Score 1 — Medium Risk (Partial)	Score 2 — High Risk (documentation absent)
C1	Model card completeness	≥500 words; usage, evaluation, limitations	50–499 words	<50 words or no card
C2	Training data named	Specifically named, linked, or HF datasets field populated	Vaguely described; no link	No dataset mentioned
C3	Dataset license	Explicit data license (CC-BY, Apache, research-only)	Model license only; no data license	No license information
C4	Verified authorship	Verified HF org or paper with institutional affiliation	Named individual; no verified org	Anonymous; no institutional link
C5	Limitations disclosed	Dedicated section with specific bias types and affected groups	Brief generic mention	No limitations section

6.4 Classification and Governance Actions

Table 7: WRS Classification Thresholds and Governance Actions

WRS Range	Classification	Governance Action
0.0–2.5	**Low Risk**	Approve for standard enterprise use; configure weights_only=True universally as compensating control for pickle-format weights
2.6–5.5	**Medium Risk**	Conditional approval: document compensating controls; weights_only=True; patch critical CVEs before production promotion; 90-day mandatory re-review
5.6–10.0	**High Risk**	Reject; require full remediation (dependency patching; format conversion to safetensors/ONNX) before re-evaluation

7. Experimental Evaluation

7.1 Setup

Ten open-source NLP models were evaluated from Hugging Face (April 4, 2026) in a sandbox (Python 3.12, Ubuntu 22.04; no outbound network after download). Tools: ClamAV v1.4.3, signature DB 27961 (AR); ModelScan v0.8.8, PickleScan v1.0.4, Fickling v1.16 (SR); pip-audit v2.10.0 with NVD+OSV, pip-licenses v5.5.5 (DR); HF Hub API v1.9.0 (PR). *Raw tool outputs are available from the authors upon request.*

Table 8: Evaluated Models — Task Type, Framework, Parameters, Weight Format, and File Size

Model	Task Type	Framework	Params	Format	Size
gpt2	NLP (Causal LM)	PyTorch	124M	pickle (.bin)	1.83 GB
facebook/bart-large-cnn	NLP (Seq2Seq)	PyTorch	400M	pickle (.bin)	3.03 GB
Helsinki-NLP/opus-mt-en-de	NLP (Translation)	PyTorch	74M	pickle (.bin)	0.28 GB
cardiffnlp/twitter-roberta-base-sentiment	NLP (Sentiment)	PyTorch	125M	pickle (.bin)	0.47 GB
nlptown/bert-base-multilingual-uncased-sentiment	NLP (Sentiment)	PyTorch	167M	pickle (.bin)	1.25 GB
cross-encoder/ms-marco-MiniLM-L-6-v2	NLP (Cross-encoder)	PyTorch	22M	pickle (.bin)	0.17 GB
ProsusAI/finbert	NLP (Financial)	PyTorch	110M	pickle (.bin)	0.41 GB
dslim/bert-base-NER	NLP (NER)	PyTorch	108M	pickle (.bin)	0.81 GB
yiyanghkust/finbert-tone	NLP (Financial)	PyTorch	110M	pickle (.bin)	0.41 GB
mrm8488/bert-tiny-finetuned-sms-spam-detection	NLP (Classification)	PyTorch	4.4M	pickle (.bin)	0.03 GB

7.2 AI-BOM Generation Results

AI-BOM generation was completed for all ten models. All models provided complete model metadata, weight file hashes, dependency declarations, and tokenizer artifacts (100% success rate per category). SHA-256 hashes computed for all 17 weight files were verified against author-published HF Hub hashes — all comparisons matched, confirming complete artifact integrity. Dataset provenance documentation achieved a 20% Good-tier rate: only facebook/bart-large-cnn (PR=0) and dslim/bert-base-NER (PR=1) achieved full provenance; six models had verifiable HF dataset cards but with incomplete license or author documentation; and four models (gpt2, nlptown, ProsusAI/finbert, yiyanghkust/finbert-tone) had no training dataset declared in any metadata field.

7.3 Security Findings

AR=0 for all models. ClamAV returned Infected files: 0 for all ten; artifact integrity was confirmed through hash verification in Section 7.2. The absence of malware is consistent with [7][8]: supply-chain threats in open-source ML repositories manifest primarily through structural and dependency risks, not embedded signatures.

SR varies by Fickling verdict. All ten models use pickle-format weights; ModelScan classified all as PyTorch/Pickle. Fickling produced three outcome groups. *SAFE (SR=2, 6 models):* gpt2, bart-large-cnn, Helsinki-NLP, nlptown, ProsusAI/finbert, dslim/bert-base-NER — is_likely_safe=true, num_nodes=3, high_severity_nodes=0. *CAUTION (SR=5, 3 models):* cardiffnlp, cross-encoder, yiyanghkust/finbert-tone — is_likely_safe=false; non-standard torch imports (torch._utils._rebuild_tensor_v2, torch.LongStorage, torch.FloatStorage); high_severity_nodes=0. *UNSAFE (SR=8, 1 model):* mrm8488/bert-tiny — is_likely_safe=false; 6 high_severity_nodes across pytorch_model.bin and training_args.bin importing from transformers.training_args and transformers.trainer_utils. Regardless of verdict, the pickle format unconditionally permits arbitrary code execution at deserialization; remediation for SAFE/CAUTION models is weights_only=True; mrm8488 additionally requires safetensors conversion. The CAUTION classification for the three models with non-standard imports reflects that torch._utils._rebuild_tensor_v2, torch.LongStorage, and torch.FloatStorage are standard PyTorch state-dict reconstruction routines with no executable payload. The UNSAFE verdict for mrm8488/bert-tiny maps to the 7–9 rubric row; the 6 high_severity_nodes arise from transformers framework imports (transformers.training_args,

transformers.trainer_utils) rather than os/subprocess modules, but the UNSAFE classification and rejection stand regardless of import source.

DR=10 for all models. pip-audit identified CVE-bearing dependencies in all ten models (100%), driven entirely by pinned transformers versions (4.2.2–4.9.0); all other dependencies returned zero CVEs. After deduplicating NVD and OSV co-reports, 18 unique CVEs were confirmed per model: 1 CRITICAL — CVE-2024-3568 (CVSS 9.6, arbitrary code execution via pickle.load() in TFPreTrainedModel.load_repo_checkpoint(), fix: transformers ≥4.38.0); 9 HIGH (CVSS 7.5–8.8) including deserialization RCE CVEs CVE-2024-11392/11393/11394 (8.8 each) and ReDoS variants; 7 MEDIUM and 1 LOW. A pip-licenses audit (95 packages) confirmed no copyleft obligations in any model's runtime dependencies; the sole GPL-flagged package (fickling) belongs to the evaluation toolchain only.

PR varies by provenance tier. Four models received Minimal or Absent tiers (PR≥6): nlptown (PR=7, Minimal), ProsusAI/finbert (PR=7, Minimal), mrm8488/bert-tiny (PR=7, Minimal), yiyanghkust/finbert-tone (PR=8, Absent). Two models achieved Good provenance: facebook/bart-large-cnn (PR=0 — cnn_dailymail dataset card verified, license confirmed, 705-word card, Facebook verified org) and dslim/bert-base-NER (PR=1 — conll2003 card verified). Four models had no training dataset in HF metadata. Two raised legal compliance concerns: cardiffnlp (tweet_eval; Twitter ToS restricts derivative use) and mrm8488 (sms_spam dataset specifies research-use-only).

7.4 WRS Computation and Governance Classification

Table 9 presents per-model WRS scores. No model reached the Low-Risk threshold (WRS ≤ 2.5) — the universal DR=10 from outdated transformers pinning and the minimum SR=2 from pickle-format weights together ensure WRS ≥ 3.10 across all evaluated models. Nine models received conditional Medium-Risk approval; one was rejected as High Risk.

Table 9: Per-Model WRS Computation and Governance Classification. WRS = 0.35×AR + 0.30×SR + 0.25×DR + 0.10×PR. Nine models: Medium Risk; one model (mrm8488/bert-tiny): High Risk — rejected.

Model	AR (×0.35)	SR (×0.30)	DR (×0.25)	PR (×0.10)	WRS	Class.	Governance Action
gpt2	0	2	10	4	**3.50**	**Medium**	Patch transformers ≥4.38.0; weights_only=True; 90-day re-review
facebook/bart-large-cnn	0	2	10	0	**3.10**	**Medium**	Patch transformers ≥4.38.0; weights_only=True
Helsinki-NLP/opus-mt-en-de	0	2	10	3	**3.40**	**Medium**	Patch transformers ≥4.38.0; provenance gap documented (PR=3, Partial)
cardiffnlp/twitter-roberta-base-sentiment	0	5	10	5	**4.50**	**Medium**	Patch transformers ≥4.38.0; legal review of tweet_eval Twitter ToS required
nlptown/bert-base-multilingual-uncased-sentiment	0	2	10	7	**3.80**	**Medium**	Patch transformers ≥4.38.0; no training dataset declared (PR=7, Minimal)

cross-encoder/ms-marco-MiniLM-L-6-v2	0	5	10	5	**4.50**	**Medium**	Patch transformers ≥4.38.0; msmarco dataset card verified; no limitations section
ProsusAI/finbert	0	2	10	7	**3.80**	**Medium**	Patch transformers ≥4.38.0; no training dataset declared (PR=7, Minimal)
dslim/bert-base-NER	0	2	10	1	**3.20**	**Medium**	Patch transformers ≥4.38.0; conll2003 verified (Good provenance)
yiyanghkust/finbert-tone	0	5	10	8	**4.80**	**Medium**	Patch transformers ≥4.38.0; absent provenance documented (PR=8, Absent)
mrm8488/bert-tiny-finetuned-sms-spam-detection	0	8	10	7	**5.60**	**HIGH**	REJECT: Fickling UNSAFE (6 high-severity nodes); convert to safetensors; patch transformers ≥4.38.0 before re-evaluation

7.5 Worked WRS Example: ProsusAI/finbert

AR=0: ClamAV clean; SHA-256 hash matched HF Hub. **SR=2:** ModelScan: PyTorch/Pickle. PickleScan: 0 GLOBAL imports. Fickling: is_likely_safe=true, num_nodes=3, SAFE. Table 5, row 2 → SR=2. **DR=10:** pip-audit: 18 unique CVEs (CVE-2024-3568 CVSS 9.6 Critical; 9 HIGH; 7 MEDIUM; 1 LOW), all from transformers 4.6.0. Worst CVSS 9.6 ≥ 9.0 → DR=10. **PR=7:** C1=1 (151 words); C2=2 (hf_datasets=[]); C3=2 (no license); C4=0 (ProsusAI verified org); C5=2 (no limitations). PR=1+2+2+0+2=7, Minimal tier.

WRS = 0.35×0 + 0.30×2 + 0.25×10 + 0.10×7 = 0 + 0.60 + 2.50 + 0.70 = 3.80 → Medium Risk (Conditional Approval)

Governance action: patch transformers ≥4.38.0; configure weights_only=True; document provenance gap (Minimal tier, no training dataset declared); 90-day re-review.

8. Discussion

Serialization and dependency risks are universally pervasive, yet structurally independent. All ten models exhibit both risks simultaneously, but they arise from independent mechanisms: serialization risk is a property of the weight file format, while dependency risk arises from outdated framework pinning. This independence means that neither risk can serve as a proxy for the other — a model can score SAFE on Fickling and still carry CVE-2024-3568 (CVSS 9.6), as demonstrated by six of the evaluated models. The implication for enterprise governance is direct: no single scanning tool is sufficient. AI-BOM-driven layered assessment is the minimum viable approach.

The transformers dependency is a systemic ecosystem risk. All ten models share the same critical CVE profile because they all pin to vulnerable transformers versions. CVE-2024-3568 specifically exploits pickle.load() in TFPreTrainedModel.load_repo_checkpoint() — an arbitrary code execution pathway that compounds serialization risk at the framework level. Upgrading to transformers ≥4.53.0 resolves all 18 identified CVEs. This finding argues for an enterprise-level dependency governance policy: models declaring transformers < 4.38.0 should be treated as conditionally approved until patched, regardless of other risk scores.

The WRS correctly discriminates within risk classes. Among nine Medium-Risk models, WRS ranged from 3.10 (bart-large-cnn, Good provenance, standard pickle) to 4.80 (yiyanghkust/finbert-tone, Absent provenance, CAUTION-level serialization). This within-class variance enables governance teams to prioritize remediation effort without requiring subjective judgment. The one High-Risk model (mrm8488/bert-tiny, WRS=5.60) was correctly distinguished by the compound signal of Fickling UNSAFE verdict combined with critical CVE and Minimal provenance — a combination that no single-dimension scan would surface.

Provenance gaps carry legal risk beyond documentation quality. Two models raised compliance concerns that no technical scanner can detect: cardiffnlp's training data is subject to Twitter Terms of Service restrictions on derivative use, and mrm8488's sms_spam dataset explicitly prohibits commercial deployment. The EU AI Act Article 11 requires training data documentation for high-risk AI systems. Enterprises deploying AI models in regulated contexts — financial services, healthcare, public sector — face direct compliance exposure from the 40% of evaluated models with Minimal or Absent provenance. The AI-BOM provenance record provides the structured evidence needed to satisfy these obligations.

Practical deployment overhead is manageable. The complete evaluation workflow — sandbox download, ClamAV scan, ModelScan+PickleScan+Fickling analysis, pip-audit scan, and HF API provenance assessment — required approximately 45–90 minutes per model. For enterprises onboarding large model volumes, automating the artifact scanning pipeline (Layers 4–7) within CI/CD infrastructure would reduce governance overhead to minutes per model, with human review reserved for Layer 8 governance approval decisions where business context and regulatory requirements must be integrated.

9. Continuous Vulnerability Monitoring

Layer 10 implements ongoing scanning of registry-approved models against updated CVE databases and malware signature feeds, tiered by risk: high-use or regulatory-critical models daily; standard models weekly; archived models monthly. Automated alerts trigger on newly published CVEs affecting any dependency declared in a model's AI-BOM. The governance team may revoke access, issue conditional notices, initiate emergency re-procurement, or document accepted risk — all actions recorded in the AI-BOM governance record to maintain the continuous audit trail required by ISO/IEC 42001 Clause 9 [2] and EU AI Act Article 72 [3].

10. Conclusion

Open-source AI models introduce supply-chain risks — pickle serialization, critical dependency CVEs, and provenance gaps — that SBOM frameworks were not designed to address. This paper presented AI-BOM integrated within E-AISCSA: a ten-layer governance architecture producing auditable evidence aligned with NIST AI RMF 1.0, ISO/IEC 42001:2023, and EU AI Act requirements.

Experimental evaluation confirmed three findings with direct enterprise policy implications. First, all ten models simultaneously exhibit serialization risk and critical CVE exposure (CVE-2024-3568, CVSS 9.6), demonstrating that no single scanning layer covers the full AI supply-chain attack surface. Second, the WRS framework produced reproducible, traceable governance decisions — nine conditional Medium-Risk approvals and one High-Risk rejection — with within-class variance enabling prioritized remediation without subjective judgment. Third, 40% of models had provenance documentation insufficient for EU AI Act compliance, including two with training data licenses that directly prohibit commercial use — risks undetectable by any technical scanner.

Limitations include reliance on static analysis (obfuscated payloads may evade Fickling), inability to verify training data content, heuristic WRS weight calibration, and absence of a standardized AI-BOM schema. Future AR evaluations should incorporate YARA pattern-based detection targeting ML-specific payloads as a complementary layer beyond signature scanning. Four directions warrant further

investigation: (1) dynamic sandbox analysis to detect obfuscated serialization attacks; (2) automation pipeline integration with CI/CD and MLOps platforms; (3) empirical calibration of WRS weights using confirmed AI supply-chain incidents; and (4) formal contribution of the AI-BOM governance record schema to CycloneDX and SPDX working groups.

Data Availability and Ethical Statements

Raw experimental data — including ClamAV scan logs, Fickling JSON outputs, pip-audit JSON files, SHA-256 hash verification records, and Hugging Face Hub provenance metadata for all ten evaluated models — are available from the corresponding author upon request.

The authors declare no competing financial interests. This research received no external funding. All models evaluated are publicly available on Hugging Face under their respective open-source licenses; no proprietary or restricted data were used. No human subjects research was conducted.

REFERENCES:

[1] NIST. (2023). Artificial Intelligence Risk Management Framework (AI RMF 1.0). NIST AI 100-1.
[2] ISO. (2023). ISO/IEC 42001:2023 — Artificial Intelligence — Management System. Geneva: ISO.
[3] European Parliament and Council. (2024). Regulation (EU) 2024/1689 of the European Parliament and of the Council of 13 June 2024 on Artificial Intelligence (AI Act). Official Journal of the European Union, OJ L, 2024/1689.
[4] Hugging Face. (2026). Hugging Face Model Hub. Retrieved April 4, 2026, from https://huggingface.co/models
[5] Carlini, N., et al. (2024). Poisoning web-scale training datasets is practical. arXiv:2302.10149.
[6] Trail of Bits. (2024). Exploiting ML models with pickle file attacks. https://blog.trailofbits.com/2024/06/11/exploiting-ml-models-with-pickle-file-attacks-part-1/
[7] Rapid7. (2025). From .pth to p0wned: Abuse of pickle files in AI model supply chains. https://www.rapid7.com/blog/post/from-pth-to-p0wned-abuse-of-pickle-files-in-ai-model-supply-chains/
[8] JFrog Security Research. (2024). Data scientists targeted by malicious Hugging Face ML models. https://jfrog.com/blog/data-scientists-targeted-by-malicious-hugging-face-ml-models-with-silent-backdoor/
[9] Sonatype. (2025). Exposing 4 critical vulnerabilities in Python PickleScan. https://www.sonatype.com/blog/bypassing-picklescan-sonatype-discovers-four-vulnerabilities
[10] OWASP Foundation. (2023). ML06:2023 — ML Supply Chain Attacks. OWASP Machine Learning Security Top 10.
[11] Executive Office of the President. (2021). Executive Order 14028: Improving the Nation's Cybersecurity. Federal Register, 86(93), 26633–26641.
[12] NTIA. (2021). The Minimum Elements for a Software Bill of Materials (SBOM). U.S. Department of Commerce.
[13] Mitchell, M., et al. (2019). Model cards for model reporting. Proceedings FAT*, pp. 220–229.
[14] Amershi, S., et al. (2019). Software engineering for machine learning: A case study. ICSE-SEIP, pp. 291–300.
[15] Synopsys. (2024). Open Source Security and Risk Analysis (OSSRA) Report 2024.
[16] NIST. (2022). Secure Software Development Framework (SSDF) v1.1. NIST SP 800-218.
[17] MITRE Corporation. (2023). MITRE ATLAS: Adversarial Threat Landscape for AI Systems. https://atlas.mitre.org/
[18] Steinhardt, J., Koh, P.W., & Liang, P.S. (2017). Certified defenses for data poisoning attacks. NeurIPS, 30, 3517–3529.
[19] JFrog Security Research. (2025). Unveiling 3 zero-day vulnerabilities in PickleScan. https://jfrog.com/blog/unveiling-3-zero-day-vulnerabilities-in-picklescan/
[20] OWASP Foundation. (2023). OWASP Top 10 for Large Language Model Applications v1.1.

[21] Gebru, T., et al. (2021). Datasheets for datasets. Communications of the ACM, 64(12), 86–92.
[22] Trail of Bits. (2021). Fickling: A Python pickling decompiler. https://github.com/trailofbits/fickling
[23] CycloneDX. (2023). CycloneDX ML-BOM Specification v1.5. https://cyclonedx.org/capabilities/mlbom/
[24] Linux Foundation. (2024). SPDX 3.0 AI Profile. https://spdx.github.io/spdx-spec/v3.0.1/model/AI/AI/
[25] Google LLC. (2021). SLSA Framework v0.1. https://slsa.dev/
[26] Torres-Arias, S., et al. (2019). in-toto: Farm-to-table guarantees for bits and bytes. USENIX Security, pp. 1393–1410.