AUSTIN EDWARDS CONSULTING GROUP

Open-Source AI Is Not Risk-Free:

What Every Organization Needs to Know About LLM Security

Luther Johnson, Ph.D.
James Wilson, CCDA, MPC

Executive Summary

Open-source large language models (LLMs) are rapidly becoming embedded in enterprise workflows. Organizations adopt them for cost efficiency, customization, and perceived data control. Many assume that local deployment reduces risk. **In practice, that assumption is often incorrect.** This white paper demonstrates that **enterprise risk exposure is not determined by whether an LLM is open-source or proprietary**, but by how it is governed, integrated, and used.

Key Findings

- Sensitive data is exposed through **routine employee workflows**, not just malicious activity
- Open-source model ecosystems introduce **software supply-chain vulnerabilities**
- LLM-integrated systems create **new attack surfaces through prompt injection**
- Local deployment **redistributes risk rather than eliminating it**

Core Insight

LLM risk is fundamentally sociotechnical. **While vulnerabilities often appear technical, they are typically the result of misalignment between system design, user behavior, and organizational governance.**

Strategic Implication

Organizations that treat AI as a productivity tool create unmanaged risk.

Organizations that treat AI as **governed.**

The Misconception: Open-Source Means Safe

The rapid adoption of open-source LLMs has been driven by a compelling narrative:

- Greater transparency
- Increased control
- Reduced dependency on external vendors

From this, a conclusion often follows:

If the model is local, the data is secure. **This conclusion often breaks down under real-world operational conditions. LLMs amplify and reshape existing categories of operational vulnerability:**

- Prompts function as unregulated data transfer channels
- Outputs may encode or reveal sensitive information
- External model artifacts behave as executable dependencies
- System boundaries between input, processing, and execution collapse

Research has demonstrated that LLMs can expose sensitive data under specific query conditions, and real-world incidents show that such exposure frequently occurs during routine use rather than adversarial attack.

Three Enterprise Risk Pathways

Across industries, three recurring patterns define how organizations experience AI-related exposure.

1. Routine Use Becomes Data Leakage

Employees interact with LLMs as productivity tools:

- Debugging code
- Drafting communications
- Summarizing internal documents

In doing so, they often submit proprietary data into systems that are not governed as data infrastructure. What appears as normal usage becomes an unmonitored data egress channel.

2. Model Ecosystems Introduce Supply-Chain Risk

Open-source LLM deployment depends on external artifacts:

- Model weights
- Tokenizers
- Inference scripts
- Supporting libraries

These components can contain:

- Malicious payloads
- Unsafe execution logic
- Hidden backdoors

Models are not static assets. They are executable supply-chain components.

3. Prompt Injection Redefines Attack Surfaces

LLM systems blur the line between:

- Data
- Instructions
- Execution

Adversarial content embedded in documents can:

- Override system prompts
- Trigger unintended actions
- Expose internal information

This is not a traditional software vulnerability. It is a failure of system boundaries.

How This Analysis Was Conducted

This analysis draws on cross-case patterns observed in:

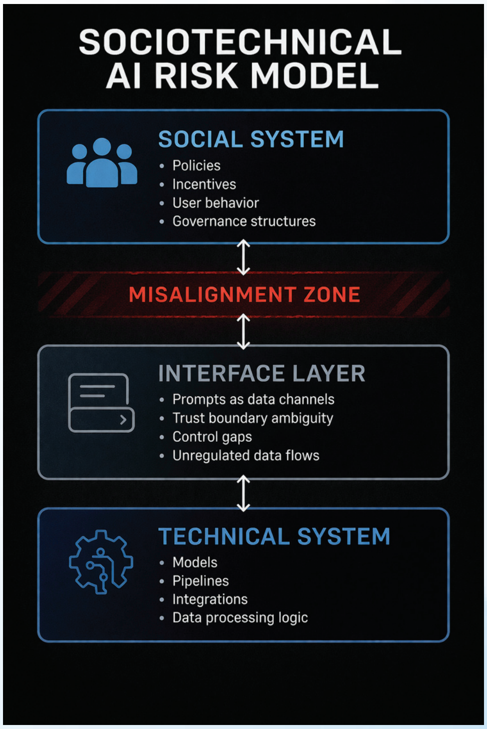
- Peer-reviewed research
- Documented enterprise incidents
- Security disclosures
- AI governance frameworks, including NIST AI RMF and OWASP

The goal is not statistical generalization, but **identifying repeatable failure patterns in enterprise AI systems.**

Sociotechnical Nature of AI Risk

AI risk does not originate solely in code. It emerges from the interaction between technical systems and human systems.

Framework Diagram:



SOCIOTECHNICAL AI RISK MODEL

SOCIAL SYSTEM
- Policies
- Incentives
- User behavior
- Governance structures

MISALIGNMENT ZONE

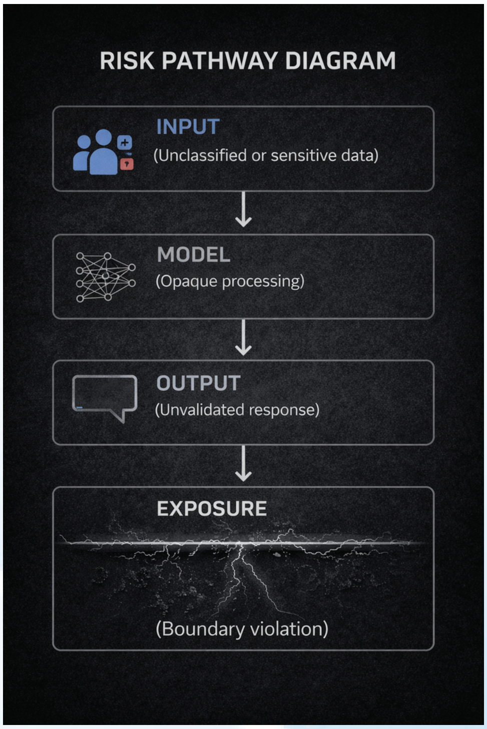
INTERFACE LAYER
- Prompts as data channels
- Trust boundary ambiguity
- Control gaps
- Unregulated data flows

TECHNICAL SYSTEM
- Models
- Pipelines
- Integrations
- Data processing logic

Key Insight

Risk emerges at the interface where governance fails to constrain technical capability.

How Exposure Actually Happens



RISK PATHWAY DIAGRAM

INPUT
(Unclassified or sensitive data)

MODEL
(Opaque processing)

OUTPUT
(Unvalidated response)

EXPOSURE
(Boundary violation)

Failure Conditions

- No data classification at input
- No control over model behavior
- No validation of outputs
- No monitoring of system boundaries

Case Patterns: Where Organizations Fail

Case 1: Routine Use, Unexpected Exposure

In a widely reported 2023 incident, employees at a major technology firm entered proprietary source code into a generative AI system during routine debugging tasks. There was no malicious intent.

What Failed

- Prompts were not governed as data assets
- AI tools were treated as informal utilitie
- Monitoring mechanisms were absent

Outcome

Sensitive intellectual property was exposed.

Case 2: Compromised Model Supply Chain

Security researchers identified malicious artifacts embedded in publicly available machine learning models. These artifacts executed code during model loading.

What Failed

- No verification of model provenance
- No sandboxing of external components
- Assumption that model files were inert

Outcome

Organizations introduced vulnerabilities into internal systems through trusted dependencies.

Case 3: Prompt Injection in Enterprise Systems

LLM-integrated applications retrieved external content and executed it as part of system workflows.

Adversarial instructions embedded in that content:

- Overrode system constraints
- Triggered unintended actions

What Failed

- No separation between content and control
- Over-privileged system access
- Weak architectural boundaries

Outcome

Systems acted outside organizational intent.

Case Comparison

Risk Pathway	Trigger	Core Failure	Enterprise Impact
Routine Use	Employee prompts	Governance failure	Data leakage
Supply Chain	External model artifacts	Trust boundary failure	System compromise
Prompt Injection	Adversarial content	Architectural failure	Unauthorized actions

What This Costs You If You Get It Wrong

AI risk is not theoretical. It translates directly into business consequences:

- **Intellectual property loss**
- **Regulatory exposure** (GDPR, HIPAA, industry-specific compliance)
- **Reputational damage**
- **Erosion of competitive advantage**

In many cases, exposure is not immediately

Strategic Implications for Leaders

Misconception:
Deployment Determines Risk

Reality:
Governance determines risk

The Cloud vs Local Tradeoff

Model	Strength	Risk Shift
Cloud (e.g., Gemini, Chat GPT-5)	Performance, scalability	External data exposure
Local (e.g., Llama 3, Mistral)	Data control	Internal security burden

There is no inherently secure option. Each model redistributes responsibility.

Defining Insight

AI systems are not just tools. They are decision engines connected to your data.

Organizations that fail to recognize this create unmanaged exposure.

The Austin Edwards AI Risk Control Framework™

Three Layers. One Principle: Control the Flow of Intelligence.

Layer	Focus	Our Perspective
Governance	Define allowed intelligence flows	Prompts are regulated data events
Architecture	Constrain system behavior	Separate content from control
Operations	Detect and respond to risk	Monitor AI as a live system

How the Framework Works

1. Governance establishes boundaries
2. Architecture enforces boundaries
3. Operations continuously validate and adapt

This is not optional. It is the minimum requirement for enterprise AI deployment.

Austin Edwards Perspective

Most organizations are not failing because AI is inherently dangerous. They are failing because they are **treating AI as a tool instead of infrastructure.**

AI systems:

- Move data
- Transform data
- Act on data

Without governance, they introduce:

- Unmonitored data flows
- Uncontrolled execution pathways
- Invisible enterprise risk

Conclusion

Open-source LLMs do not eliminate risk. They shift it. Organizations that succeed will not be those that adopt AI fastest, but those that **align governance, architecture, and operations around it.**

Call to Action

Organizations that treat LLMs as governed infrastructure rather than productivity tools are better positioned to mitigate emerging risks.

Austin Edwards Consulting Group helps organizations design, implement, and operationalize AI governance frameworks that reduce enterprise risk while enabling innovation.

References

Bommasani, R., et al. (2022). On the opportunities and risks of foundation models. arXiv. https://doi.org/10.48550/arXiv.2108.07258

Carlini, N., et al. (2021). Extracting training data from large language models. Proceedings of the 30th USENIX Security Symposium. https://doi.org/10.48550/arXiv.2012.07805

Floridi, L., & Cowls, J. (2022). A unified framework of five principles for AI in society. Harvard Data Science Review. https://doi.org/10.1162/99608f92.8cd550d1

Greshake, K., Abdelnabi, S., Gawlikowski, J., et al. (2023). Not what you've signed up for: Compromising real-world LLM-integrated applications with indirect prompt injection. arXiv. https://doi.org/10.48550/arXiv.2302.12173

National Institute of Standards and Technology. (2023). Artificial Intelligence Risk Management Framework (AI RMF 1.0) (NIST AI 100-1). https://nvlpubs.nist.gov/nistpubs/ai/nist.ai.100-1.pdf

OWASP. (2024). OWASP Top 10 for Large Language Model Applications. https://owasp.org/www-project-top-10-for-large-language-model-applications/

Zimmermann, T., et al. (2020). Software supply chain security. IEEE Security & Privacy, 18(5), 44–52. https://doi.org/10.1109/MSEC.2020.3007895

AUSTIN EDWARDS
CONSULTING GROUP

http://AustinEdwards.net
info@AustinEdwards.net
(210) 704-7021