The Collective Fragility Paradox

Rethinking Third-Party Risk in Software Supply Chains

Trevor Kavanaugh

Vice President, Third-Party Risk Management

January 2026

This paper represents independent analysis and is not an official position of any organization.

Table of Contents

Executive Summary: Managing Dependencies, Not Parties

The Multi-Billion Dollar Wake-Up Call

Between December 2020 and July 2024, the financial services industry experienced a series of technology incidents with combined direct costs conservatively estimated at $30-40 billion, with some analyses suggesting total economic impact—including indirect costs, opportunity losses, and long-tail remediation—exceeding $200 billion.[1] These incidents exposed a fundamental flaw in how we approach third-party risk management. The SolarWinds supply chain attack, Log4j vulnerability, Kaseya VSA ransomware campaign, 3CX supply chain compromise, MOVEit mass exploitation, xz Utils infiltration attempt, and CrowdStrike global outage were not failures of vendor oversight—they were failures of dependency management.

Traditional Third-Party Risk Management (TPRM) frameworks organize risk around legal entities: direct vendors (third parties), their vendors (fourth parties), and in sophisticated programs, fifth parties. But modern technology incidents do not respect organizational boundaries. They propagate through technical dependencies that cut across vendor relationships, regulatory jurisdictions, and risk management silos.

The industry manages "parties" but should be managing "dependencies."

Three Dimensions of Hidden Risk

This paper proposes that modern vendor dependencies create risk across three distinct but interconnected dimensions:

1. Supply Chain Risk: What's In The Product

When a critical vulnerability was discovered in Log4j in December 2021, it affected the majority of Java applications globally—with industry analyses estimating 60-64% of Java applications were vulnerable.[2] The vulnerability was not in products that financial institutions directly procured—it was embedded 5, 10, or even 20 levels deep in dependency chains.

The xz Utils backdoor attempt of 2024 revealed a qualitatively different threat: a nation-state-level actor spent 2.5 years patiently building trust as a maintainer of a widely-used compression library before inserting a backdoor targeting SSH authentication.[3] The compromise was discovered accidentally by a Microsoft engineer investigating performance anomalies—not through any systematic monitoring. Had it reached production, it would have affected every major Linux distribution. The Open Source Security Foundation warned this "may not be an isolated incident," suggesting a reusable attack pattern against volunteer-maintained critical infrastructure.[3]

Key Statistics:

- Apache Log4j appears in over 17,000 packages on Maven Central alone.[4]
- Analysis of modern applications shows approximately 60% of vulnerabilities come from transitive dependencies (dependencies of dependencies).[5]
- Most organizations lack comprehensive software composition visibility.[6]

The Challenge: Traditional vendor assessments focus on what vendors do with data and how they secure it. But they rarely examine what components vendors embed in their software, creating blind spots that span the entire industry.

2. Software Delivery Risk: How It Gets Updated

The SolarWinds attack in December 2020 demonstrated how software update mechanisms can become attack vectors. Nation-state actors compromised the build process for SolarWinds Orion software, inserting malicious code into legitimate updates distributed to approximately 18,000 organizations, including 9 U.S. federal agencies.[7]

The CrowdStrike incident of July 19, 2024, showed that even security vendors can create systemic risk through faulty updates. A defective update to CrowdStrike's Falcon Sensor crashed approximately 8.5 million Windows systems globally,[8] causing what Microsoft called "the largest outage in the history of information technology."[9]

Key Statistics:

- SolarWinds: ~18,000+ organizations installed compromised software.[10]
- SolarWinds: 9 federal agencies confirmed compromised.[11]
- CrowdStrike: 8.5 million devices affected.[12]
- CrowdStrike: Commercial banking identified as one of most impacted sectors.[13]
- Estimated $10+ billion in financial losses from CrowdStrike incident.[14]

The Challenge: TPRM programs extensively evaluate vendors' security controls but often overlook the mechanisms by which vendors distribute updates. Yet software delivery systems represent critical trust relationships that, when compromised, can impact thousands of organizations simultaneously.

3. Cloud/Infrastructure Concentration Risk: Where It Runs

Perhaps the most insidious form of hidden dependency risk is infrastructure concentration. Financial institutions may have diversified vendor portfolios with hundreds of third-party relationships, appearing to reduce concentration risk. But when examining where those vendors' applications actually run, apparent diversity collapses into concentration.

The Collective Fragility Paradox: Each individual decision to outsource to specialized providers is rational—leveraging economies of scale, accessing expertise, reducing costs. But when every organization makes the same rational decision, the industry collectively converges on a handful of critical infrastructure providers: AWS, Microsoft Azure, Google Cloud for compute; Cloudflare and Akamai for CDN; specialized payment processors; major cybersecurity vendors.

Key Insight from Recent Incidents:

- Kaseya VSA ransomware attack (July 2021): 50-60 managed service providers compromised, affecting 800-1,500 downstream businesses—demonstrating how MSP relationships create a 15-25x attack force multiplier invisible to traditional TPRM.[15]
- MOVEit Transfer vulnerability (May 2023): 2,773+ organizations affected, 95.8 million individuals' data exposed, estimated $15.8 billion in costs.[16]
- 3CX supply chain attack (March 2023): 600,000+ organizations affected, first documented "supply chain of a supply chain" attack with a five-level dependency chain.[17]
- UK FCA data (2022-2023): Third-party incidents identified as leading cause of operational incidents.[18]

Industry Statistics:

- Approximately 29% of all breaches attributable to third-party attacks.[19]

- 97% of major US banks experienced exposure through third- or fourth-party relationships in 2024—though only a handful of vendors were actually attacked, illustrating the multiplicative effect of shared dependencies.[20]

The Challenge: Traditional concentration risk management focuses on avoiding over-reliance on individual vendors. But it does not address the reality that hundreds of different vendors may all depend on the same underlying infrastructure, creating systemic concentration that vendor diversification strategies fail to mitigate.

Regulatory Frameworks Are Catching Up—Slowly

Financial regulators have evolved TPRM guidance substantially over the past decade, with major milestones including:

- **2013**: OCC Bulletin 2013-29 distinguished "critical activities" requiring heightened oversight and introduced comprehensive lifecycle management.[21]
- **2021**: Basel Committee's Principles for Operational Resilience included first explicit regulatory recognition of fourth-party risk management.[22]
- **2023**: Interagency Guidance (OCC Bulletin 2023-17 / Federal Reserve SR 23-4 / FDIC FIL-23-2023) harmonized federal banking regulator expectations and emphasized board accountability.[23]
- **2025**: EU Digital Operational Resilience Act (DORA) entered into application (adopted December 2022), creating direct oversight framework for critical ICT service providers and mandatory concentration risk management.[24]

However, regulatory guidance still exhibits a fundamental gap: It addresses third-party relationships (and increasingly, fourth-party relationships) but does not provide practical frameworks for managing the nth-party dependencies that characterize modern software supply chains, software delivery systems, and cloud infrastructure concentration.

Notable Quote from UK FCA: Following the CrowdStrike incident, the UK Financial Conduct Authority observed that "even organisations that comply with ISO 27001, SOC 2, and NIST CSF were still caught unawares."[25] Compliance with traditional frameworks did not prevent impact from dependency risks.

The Compliance Framework Problem

Financial institutions extensively rely on SOC 2 reports as a cornerstone of vendor assurance. But SOC 2 itself has limitations for addressing modern dependency risks:

What SOC 2 Addresses Well:

- Organizational security controls.
- Access management and monitoring.
- Change management processes.
- Incident response capabilities.
- Business continuity planning.

What SOC 2 Wasn't Designed to Address:

- Software composition and component dependencies.
- Security of software build and delivery mechanisms.

- Infrastructure provider dependencies (vendors' vendors).
- Open source component governance.
- Concentration risks from shared infrastructure.

The Unchanged Framework: The core Trust Services Criteria governing SOC 2 have remained largely unchanged since 2017,[26] despite the emergence of major supply chain and dependency risks. While the AICPA updated "points of focus" in 2022, the fundamental criteria do not require comprehensive dependency visibility, software supply chain security assessment, or infrastructure concentration analysis.

Historical Context: The SOC framework emerged from the AICPA's response to market misuse of SAS 70 (Statement on Auditing Standards No. 70). Companies began demanding "SAS 70 reports" from all vendors regardless of whether financial reporting was relevant, and the AICPA formalized this by creating SOC 2 in 2011.[27] This history is instructive: SOC 2 was designed to address organizational security controls, not to map complex dependency chains or assess systemic concentration risk.

The Technical SME Gap

A critical challenge facing TPRM teams is the gap between traditional vendor management expertise and the technical depth required to evaluate modern software supply chains, delivery mechanisms, and cloud architectures.

Traditional TPRM Staffing:

- Compliance professionals.
- Vendor contract specialists.
- Risk analysts.
- Business relationship managers.

Emerging Technical Requirements:

- Software composition analysis understanding.
- Cloud architecture and shared responsibility models.
- Software delivery pipeline security (CI/CD).
- Open source governance.
- Infrastructure-as-code evaluation.
- API security architecture.

Examiner Evolution: Regulatory examiners are increasingly asking technical questions about software delivery mechanisms, infrastructure dependencies, and supply chain visibility that traditional TPRM teams struggle to answer effectively. This creates examination findings and remediation expectations that require capabilities TPRM functions were not built to provide.

A Framework for Dependency Risk Management

Rather than replacing existing TPRM frameworks, this paper proposes augmenting them with three parallel dependency risk dimensions:

Risk Dimension	Traditional TPRM Question	Dependency Risk Question
Supply Chain	"Who is the vendor and what do they do?"	"What components are in the vendor's product?"
Delivery	"How does the vendor secure their environment?"	"How does the vendor distribute updates to my environment?"
Infrastructure	"What are our critical vendor relationships?"	"What infrastructure do our critical vendors depend on?"

The chapters that follow examine each dimension in detail, explore the structural limitations of current assurance frameworks, and consider what mature dependency risk management looks like in practice.

Call to Action

The tens of billions of dollars in incidents between 2020 and 2024 have demonstrated that traditional third-party risk management frameworks, while necessary, are insufficient for managing the complexity of modern technology dependencies.

This is a multi-stakeholder problem requiring a multi-stakeholder solution. Financial institutions cannot solve this alone. Progress requires coordinated evolution across:

- **Regulators**: Creating demand signals for dependency transparency (SBOM requirements, supply chain attestation).
- **Assurance frameworks**: Updating attestation scope to address software composition and delivery mechanisms.
- **Vendors and fintechs**: Generating SBOMs, adopting secure development practices, providing infrastructure transparency.
- **Financial institutions**: Building internal technical capability to consume and act on dependency information.

For individual institutions, this means augmenting traditional tools with:

- Software composition visibility.
- Software delivery risk assessment.
- Infrastructure dependency mapping.
- Technical SME integration into TPRM teams.

The good news: Many of the tools and techniques required already exist in application security, DevSecOps, and cloud architecture disciplines. The challenge is organizational—integrating technical depth into traditionally compliance-focused TPRM functions.

The question for financial institutions: Will you wait for the next major incident to expose a dependency risk you did not know you had, or will you proactively build the capability to see beyond vendor relationships to the dependencies that determine your actual risk exposure?

The chapters that follow trace how we arrived at this structural mismatch—and what it will take to close the gap.

Endnotes - Executive Summary

[1] Aggregate cost estimate methodology: SolarWinds (~$100M total, combining $18-19M direct remediation and $90M+ in cumulative legal/regulatory/insurance costs per Arnold & Porter analysis); Log4j (estimates range from $12B to $100B+ depending on methodology and scope—lower bound based on conservative industry estimates of global enterprise response costs, upper bound includes indirect impacts across the estimated 64% of Java applications affected); MOVEit ($15.8B, per Emsisoft breach cost analysis covering 2,773+ organizations and 93M+ individuals); CrowdStrike ($10B+ for Fortune 500 companies alone, per Parametrix study; ~$5.4B for top 500 US companies excluding Microsoft). Conservative estimates placing emphasis on documented direct costs total approximately $30-40B; upper-bound estimates incorporating indirect impacts, long-tail remediation, and opportunity costs across all affected organizations exceed $200B.

[2] Impact estimates vary by methodology. CISA Cyber Safety Review Board, "Review of the December 2021 Log4j Event," July 2022, documented that 35,000+ Java packages (8% of Maven Central) were affected. Industry analyses from Contrast Security and others estimated 60-64% of Java applications contained the vulnerability. The exact percentage depends on methodology (analyzing packages vs. deployed applications vs. vulnerable code paths), but all sources agree the impact was extraordinarily widespread. Available at: https://www.cisa.gov/sites/default/files/publications/CSRB-Report-on-Log4-July-11-2022_508.pdf

[3] The xz Utils backdoor incident (CVE-2024-3094, disclosed March 2024) involved a sophisticated supply chain attack where a malicious actor gained maintainer access to the xz Utils compression library through years of social engineering, then inserted a backdoor targeting SSH authentication on Linux systems. The backdoor was discovered by a Microsoft engineer investigating performance anomalies before widespread deployment to production systems. See: CISA Alert, "Reported Supply Chain Compromise Affecting XZ Utils Data Compression Library," March 29, 2024; Ars Technica, "What we know about the xz Utils backdoor that almost infected the world," April 2024.

[4] Apache Log4j Maven Central package statistics cited in: "From SolarWinds to Log4j," Check Point Security blog. Available at: https://blog.checkpoint.com/security/from-solarwinds-to-log4j-the-global-impact-of-todays-cybersecurity-vulnerabilities/

[5] Endor Labs, "State of Dependency Management 2024," available at: https://www.endorlabs.com/state-of-dependency-management. Multiple industry analyses confirm that the majority of vulnerabilities in modern applications originate in transitive dependencies rather than direct dependencies. See also: Snyk, "State of Open Source Security Report 2024" for corroborating data on transitive dependency risk.

[6] Synopsys 2024 Open Source Security and Risk Analysis (OSSRA) Report: While 96% of codebases contain open source, comprehensive software composition visibility remains limited. See also: Endor Labs, "State of Dependency Management 2024" survey findings.

[7] "Lessons Learned from the SolarWinds Cyberattack," Arnold & Porter, June 2021. Available at: https://www.arnoldporter.com/en/perspectives/advisories/2021/06/lessons-learned-from-the-solarwinds-cyberattack

[8] "2024 CrowdStrike-related IT Outages," Wikipedia. Available at: https://en.wikipedia.org/wiki/2024_CrowdStri
related_IT_outages

[9] Microsoft statement on CrowdStrike incident, July 2024, cited in multiple sources including UK FCA analysis.

[10] SolarWinds incident statistics from Arnold & Porter analysis and CISA reporting.

[11] U.S. Government Accountability Office (GAO), “SolarWinds Cyberattack Demands Significant Federal and Private-Sector Response,” April 2021. GAO-21-501.

[12] Microsoft Azure Status blog post on CrowdStrike incident impact assessment, July 2024.

[13] UK Financial Conduct Authority, “CrowdStrike Outage: Lessons for Operational Resilience,” October 31, 2024. Available at: https://www.fca.org.uk/firms/operational-resilience/crowdstrike-outage-lessons-operational-resilience

[14] CrowdStrike incident cost estimates from: “How the 2024 CrowdStrike Outage Revealed Glaring Gaps in Risk and Incident Management,” OneClickComply. Available at: https://oneclickcomply.com/blog/how-the-2024-crowdstrike-outage-revealed-glaring-gaps-in-risk-and-incident-management

[15] Kaseya VSA ransomware attack (July 2, 2021): REvil ransomware gang exploited vulnerabilities in Kaseya’s VSA remote monitoring software used by managed service providers (MSPs). Approximately 50-60 MSPs were directly compromised, cascading to 800-1,500 downstream businesses. The attack demonstrated how MSP relationships create attack force multiplication invisible to traditional TPRM—a single MSP compromise affecting 15-25+ downstream organizations. Notable impact: Coop Sweden’s 800 grocery stores were closed when their POS system vendor’s MSP was compromised. Source: CISA, “Kaseya VSA Supply-Chain Ransomware Attack,” July 2021; Huntress Labs incident analysis.

[16] Emsisoft, “Unpacking the MOVEit Breach: Statistics and Analysis,” updated December 2023. Available at: https://www.emsisoft.com/en/blog/44123/unpacking-the-moveit-breach-statistics-and-analysis/

[17] 3CX supply chain attack (March 2023): First documented “supply chain of a supply chain” attack. The 3CX desktop application was compromised through a prior compromise of X_TRADER trading software used by a 3CX employee, creating a five-level dependency chain. Coalition Inc. concluded that “complete supply chain visibility is practically impossible” given such attack complexity. Source: SentinelOne, “SmoothOperator | Ongoing Campaign Trojanizes 3CXDesktopApp in Supply Chain Attack,” March 2023. Available at: https://www.sentinelone.com/blog/smoothoperator-ongoing-campaign-trojanizes-3cxdesktopapp-in-supply-chain-attack/

[18] UK FCA operational resilience data (2022-2023) cited in CrowdStrike lessons learned publication, October 2024.

[19] SecurityScorecard, “The State of Third-Party Risk Management,” 2024. Available at: https://securityscorecard.c
party-risk-management/

[20] SecurityScorecard and The Cyentia Institute, “Close Encounters of the Third (and Fourth) Party Kind” (2024). The report analyzed security incidents affecting financial services organizations and found that 97% of major US banks experienced exposure through third- or fourth-party relationships in 2024, despite the actual number of directly compromised vendors being relatively small. This demonstrates the multiplicative effect of shared dependencies. Note: This figure represents exposure through vendor relationships (having a connection to a breached vendor’s network), not necessarily confirmed data compromise. Vendor security rating methodologies vary.

[21] OCC Bulletin 2013-29: “Third-Party Relationships: Risk Management Guidance,” October 30, 2013. Available at: https://lenderscompliancegroup.com/wp-content/uploads/2020/02/OCCBulletin2013-292CThirdPartyRelationships28RiskManagement29.pdf

[22] Basel Committee on Banking Supervision, "Principles for Operational Resilience and Revised PSMOR," March 2021. BIS Newsletter on Third and Fourth-Party Risk. Available at: https://www.bis.org/publ/bcbs_nl28.htm

[23] OCC Bulletin 2023-17 / Federal Reserve SR 23-4 / FDIC FIL-23-2023: "Interagency Guidance on Third-Party Relationships: Risk Management," June 6, 2023. Available at: https://www.occ.gov/news-issuances/bulletins/2023/bulletin-2023-17.html

[24] EU Digital Operational Resilience Act (DORA), Regulation (EU) 2022/2554, adopted December 14, 2022, application date January 17, 2025. Available at: https://www.eiopa.europa.eu/digital-operational-resilience-act-dora_en

[25] UK Financial Conduct Authority, "CrowdStrike Outage: Lessons for Operational Resilience," October 31, 2024.

[26] AICPA, "2017 Trust Services Criteria for Security, Availability, Processing Integrity, Confidentiality, and Privacy" (with Revised Points of Focus – 2022). Core criteria unchanged since 2017; points of focus updated September 2022. Available at: https://www.aicpa-cima.com/resources/download/2017-trust-services-criteria-with-revised-points-of-focus-2022

[27] Secureframe, "The History of SOC 2," 2023. Available at: https://secureframe.com/hub/soc-2/history. See also: PKF AvantEdge, "A Brief History of SOC and SAS."