Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Attack-Path-Aware Enterprise Risk Quantification for Cloud-Native Financial Data Platforms Using AI and Cybersecurity Graphs

Gopinathan Rathinavelu

gopinathan.r@gmail.com

Data Architect

Research Article

Keywords: Enterprise Risk Quantification, Attack Graphs, Cloud-Native Security, Financial Data Platforms, FAIR, MITRE ATT&CK, Graph Neural Networks, AI Governance

Posted Date: November 12th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11405581/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Abstract

Enterprise risk quantification in banking has historically treated vulnerability management and financial-impact analysis as separate disciplines: scanners rank findings by base severity scores, while risk functions estimate loss with coarse, qualitative heat maps. Neither accounts for how an adversary actually chains weaknesses across a modern cloud-native data platform to reach regulated financial data. This paper presents an attack-path-aware risk-quantification architecture that unifies the two. The system ingests posture and identity telemetry from cloud-security, identity, and configuration-management sources, constructs a directed attack graph whose nodes are asset and privilege states and whose edges are adversary transitions tagged by MITRE ATT&CK tactic, and uses this graph to propagate compromise likelihood toward designated crown-jewel assets. An AI layer scores edge-transition likelihoods from heterogeneous signals using a graph neural network with a transparent fallback, and large language models are used only in a constrained, retrieval-grounded role to normalize unstructured control evidence into the graph schema. Risk is expressed in monetary terms through a Factor Analysis of Information Risk (FAIR) loss model driven by graph-derived frequencies, producing an annualized loss distribution rather than a point estimate. A budget-constrained optimizer then selects controls by their marginal effect on crown-jewel loss, making remediation path-aware rather than severity-ordered. We describe the end-to-end technical architecture, formalize the quantification method, and demonstrate it on a synthetic but representative cloud data platform. In the illustrative study, path-aware control selection reduced expected annual loss by 85% at a fixed budget, versus 41% for conventional severity-ordered selection, and correctly identified a third-party integration path—low in individual CVSS severity—as the dominant driver of crown-jewel risk. We position the method as a decision-support instrument for model-risk and operational-risk functions, requiring human validation and governance rather than autonomous action.

1 Introduction

Banking technology estates have moved decisively to cloud-native data platforms: streaming ingestion of transaction events, object-storage data lakes organized into raw and curated zones, cloud data warehouses, and machine-learning feature stores that serve fraud, credit, and marketing models. These platforms concentrate exactly the data that regulation and attackers both care about—cardholder data, account records, and the derived features that drive automated financial decisions. At the same time, the control surface has fragmented across identity and access management, Kubernetes workloads, infrastructure-as-code, secret stores, and a long tail of third-party integrations.

Two established disciplines attempt to manage the resulting risk, and they do not meet. Vulnerability and posture management enumerates weaknesses and ranks them, overwhelmingly by the Common Vulnerability Scoring System (CVSS) base score [6]. Enterprise and operational risk management, in turn, estimates potential loss, frequently through qualitative likelihood–impact matrices that supervisors have criticized as imprecise and difficult to aggregate [5, 8]. A severity-ordered remediation queue answers the question “which finding is individually worst?” but not “which weakness most increases the chance that

an adversary reaches our regulated data?”—and it is the latter question that determines financial exposure.

The gap is consequential because real intrusions are paths, not points. An adversary composes individually unremarkable steps—an exposed edge service, an over-privileged workload identity, a reachable secret, a trusted vendor connection—into a chain that terminates at a crown-jewel asset. Attack-graph analysis has modelled such chains in the academic literature for over two decades [9, 10, 12], and the MITRE ATT&CK knowledge base [13] has given industry a shared vocabulary for adversary behaviour. What has been missing in the banking data-platform context is a method that (i) builds such a graph automatically from the telemetry an enterprise already collects, (ii) propagates likelihood through it with defensible, auditable scoring, (iii) converts the result into a monetary loss distribution that risk committees can act on, and (iv) uses that distribution to prioritize controls by their effect on loss, not on nominal severity.

This paper presents such a method and its technical architecture. Our contributions are:

- An **attack-path-aware risk-quantification architecture** for cloud-native financial data platforms that unifies posture telemetry, attack-graph reasoning, and monetary loss modelling into a single decision-support pipeline (Section 4).
- A **quantification method** that propagates ATT&CK-tactic-tagged transition likelihoods to crown-jewel assets and feeds a FAIR-based Monte Carlo loss model, yielding an annualized loss distribution and per-edge criticality (Section 5).
- A **constrained use of AI**: a graph neural network with a transparent logistic fallback for edge-likelihood scoring, and retrieval-grounded large-language-model normalization of unstructured control evidence—deliberately scoped to keep the risk computation inspectable (Section 5.3).
- A **path-aware control optimizer** that selects remediations by marginal reduction in crown-jewel loss under a budget, and an **illustrative evaluation** on a synthetic platform showing substantially better loss reduction than severity-ordered selection (Section 6).

We are explicit that the system is a decision-support instrument. Its outputs are estimates built on expert-elicited and telemetry-derived inputs; they are intended to inform, and be validated by, model-risk and operational-risk functions under existing supervisory expectations [2, 4], not to trigger autonomous change.

2 Related work

2.1 Attack Graphs and Path Analysis

Model-based attack graphs were introduced to reason about how atomic exploits compose into multi-step intrusions. Phillips and Swiler [10] framed network vulnerability as a graph of attacker states; Sheyner et al. [12] generated and analysed attack graphs via model checking; and Noel and Jajodia [9]

advanced scalable topological analysis and visualization. Later work introduced Bayesian attack graphs to reason about uncertainty in exploitation [11], and security metrics derived from graph structure [15]. Our work inherits this lineage but differs in target and output: we model a cloud-native data platform at the level of asset and privilege states rather than host-and-CVE pairs, tag transitions by ATT&CK tactic for auditability, and terminate the analysis in a monetary loss distribution rather than a reachability or structural metric.

2.2 Risk Quantification and the FAIR Model

Quantitative risk analysis argues for expressing cyber risk in the same monetary, probabilistic language as other enterprise risks [8]. The Open Group's Factor Analysis of Information Risk (FAIR) [14] decomposes risk into loss event frequency and loss magnitude, each estimated as a distribution and combined by Monte Carlo simulation. FAIR has seen banking adoption but is typically driven by analyst-elicited frequencies that are disconnected from the actual technical attack surface. We retain FAIR's loss-magnitude discipline while replacing its frequency input with graph-derived, path-aware likelihoods.

2.3 AI and Graph Learning for Security

Graph neural networks (GNNs) have been applied to intrusion and fraud detection and to vulnerability and threat analysis over graph-structured security data [1, 16]. Separately, large language models have been explored for security operations, from alert triage to control-evidence summarization, with well-documented hallucination and provenance concerns [7]. Our design uses a GNN for edge-likelihood scoring where labelled transition outcomes and rich node features are available, and confines LLMs to a retrieval-grounded normalization task with human validation, so that the quantitative core of the system remains inspectable and reproducible.

2.4 Cloud-Native and Supervisory Context

Cloud-native security posture management (CSPM) and cloud-native application protection platforms (CNAPP) provide the configuration, identity, and workload telemetry our system consumes [3]. On the governance side, model-risk guidance [2] and emerging AI-specific regulation [4] shape how an AI-assisted risk instrument must be validated, documented, and overseen. Our contribution is to connect these operational and governance layers through an explicit, quantitative, path-aware model.

3 Threat model and problem formulation

3.1 Scope and Assumptions

We consider an enterprise cloud-native data platform operated by a financial institution. The defender seeks to protect a set of crown-jewel assets J—in our setting a cloud data warehouse, an ML feature store, and a backup vault—against an external adversary who may also leverage a compromised third-party integration. We model the adversary as rational and multi-step, composing transitions across the estate to reach J. We deliberately abstract away operational exploitation detail: edges denote that a

transition between asset/privilege states is feasible and with what likelihood, tagged by ATT&CK tactic, not how it would be performed. This abstraction is sufficient for risk quantification and avoids encoding an operational playbook.

3.2 Formal Model

Let G = (V, E) be a directed attack graph. Each v ∈ V is an asset or privilege state; a distinguished node s ∈ V is the attacker's initial foothold and J ⊆ V are crown jewels. Each edge e = (u, v) ∈ E carries an annual transition likelihood p$_e$ ∈ [0, 1] and an ATT&CK tactic label τ$_e$. For a subset of live edges L ⊆ E, a node is compromised if it is reachable from s using only edges in L. Treating each edge as an independent Bernoulli variable with parameter p$_e$ induces a distribution over compromise states. The quantity of interest for asset j is its compromise probability

π$_j$ = Pr[j reachable from s in (V, L)], L$_e$ ~ Bernoulli(p$_e$). (1)

Because exact computation is #P-hard in general for such network-reliability problems, we estimate π$_j$ by Monte Carlo sampling of L and reachability propagation.

4 System architecture

Figure 1 End-to-end architecture. Posture, identity, and configuration telemetry (telemetry plane) feed a graph-construction service that builds an ATT&CK-tagged attack graph over the cloud-native data platform (platform plane). The risk-quantification plane scores edge likelihoods, propagates crown-jewel compromise probability, converts it to a monetary loss distribution via a FAIR engine, and selects controls by marginal loss reduction. The AI components (dashed) are bounded: a GNN scores edges and an LLM normalizes unstructured evidence under retrieval grounding and human validation.

4.1 Platform Plane

The protected platform follows a conventional lakehouse pattern: streaming ingestion of transaction events, a raw data-lake zone, a governed curated zone, a cloud data warehouse, and an ML feature store, with an application tier exposing customer APIs. Identities include human developers, workload identities bound to compute, and a CI/CD deployment identity. This plane is described by the system, not modified by it.

4.2 Telemetry Plane

Four classes of signal are ingested through connectors: (i) cloud security posture and workload findings from CSPM and CNAPP tooling; (ii) identity and entitlement data from the identity provider and cloud IAM, used to infer privilege-escalation and lateral-movement edges; (iii) configuration and relationship data from the configuration-management database and infrastructure-as-code; and (iv) vulnerability and exposure data, mapped to ATT&CK tactics. A normalization service reconciles these into a common asset and identity model.

4.3 Risk-Quantification Plane

This plane is the contribution of the paper and comprises five services:

- **Graph Construction** assembles G = (V, E) from the normalized model, instantiating edges where telemetry indicates a feasible transition and attaching tactic labels.
- **Edge-Likelihood Scoring** assigns each p□ (Section 5.3).
- **Propagation Engine** estimates crown-jewel compromise probabilities π□ by Monte Carlo reachability (Section 5.1).
- **FAIR Loss Engine** combines π□ with asset loss-magnitude distributions into an annualized loss distribution (Section 5.2).
- **Control Optimizer** selects remediations under budget by marginal loss reduction (Section 5.4).

A governance and audit store records inputs, model versions, and outputs for every run, supporting validation and supervisory review.

5 Risk quantification methodology

5.1 Compromise-Probability Propagation

Given scored edges, the propagation engine draws N independent realizations of the live-edge set $L^{(□)}$, k = 1, …, N, and computes reachability from s by iterating edge firing to a fixpoint (the graph is shallow, so a bounded number of passes suffices). The compromise probability of each node is the sample mean of its reachability indicator, and the crown-jewel event is the union over J:

$\hat{\pi}_J = (1/N) \Sigma□ \, 1[\exists j \in J : j \text{ reachable in } L^{(□)}]$. (2)

Monte Carlo error is reported as a standard error over N; we use N = 2×10⁵ in the evaluation.

5.2 Monetary Loss Model (FAIR)

For each crown jewel j we specify a loss-magnitude distribution combining a primary loss (lognormal, parameters elicited from business impact analysis) and a secondary loss (regulatory and legal). On each Monte Carlo draw, the realized loss is the sum over crown jewels actually compromised in that draw:

$\ell^{(□)} = \Sigma_{j \in J} 1[j \text{ compromised in } L^{(□)}] \cdot (M□^{(□)} + S□)$, (3)

where $M□^{(□)} \sim$ Lognormal(μ□, σ□) and S□ is the secondary loss. The engine reports the annualized loss expectation (ALE) together with the 95th and 99th percentiles of the loss distribution, giving risk committees both a central estimate and a tail view rather than a single point.

5.3 AI Components, Bounded by Design

Edge likelihoods p□ are the only learned quantities in the pipeline, and they are produced conservatively. Where sufficient labelled transition outcomes and node features exist, a graph neural network scores edges from node embeddings (asset type, exposure, privilege, segmentation) and edge context (tactic, telemetry confidence). Where data are sparse, the system falls back to a transparent logistic model over the same features and to expert-elicited priors, and the active path is recorded per edge. Large language models are used only to normalize unstructured control evidence—policy text, ticket notes, architecture documents—into the graph's structured schema, and only under retrieval grounding with mandatory human validation before any value enters the graph. This boundary is deliberate: the quantitative core (propagation, loss, optimization) is fully inspectable and reproducible, so that the system meets model-risk expectations for explainability [2, 4].

5.4 Path-Aware Control Selection

Each candidate control c mitigates a set of edges, multiplying their likelihoods by a residual factor f_c < 1, at annual cost cost_c. Given budget B, the optimizer selects C to minimize expected crown-jewel loss:

$$\min_C E[\ell \mid C] \text{ s.t. } \Sigma_{c \in C} \text{cost}_c \le B. \quad (4)$$

We solve this greedily by marginal return on investment: at each step the control with the greatest reduction in expected loss per unit cost is added, re-evaluating expected loss through the full propagation and FAIR pipeline so that interactions between controls on shared paths are captured. This makes prioritization path-aware: a control is valuable to the extent it severs or attenuates paths to crown jewels, regardless of the nominal severity of the weaknesses it addresses.

6 Illustrative evaluation

6.1 Setup

We instantiate the method on a synthetic but representative cloud-native financial data platform with 13 asset/privilege states and 18 ATT&CK-tactic-tagged transitions; all inputs are synthetic expert estimates and are reported in full so the study is reproducible. Three crown jewels are designated (warehouse, feature store, backup vault). Edge likelihoods, control costs, and residual factors are given in Table 1. Loss-magnitude parameters per crown jewel are lognormal with medians of US$14M, US$5M, and US$9M and secondary losses of US$6M, US$2M, and US$4M respectively. The propagation uses $N = 2\times10^5$ Monte Carlo draws. This is an illustrative study designed to exercise and explain the method, not an empirical validation on production data.

Table 1
Candidate Controls for the Illustrative Platform

Control	CVSS of weakness	Cost ($k/yr)	Residual factor
Edge-service hardening	9.1	120	0.35
Vendor/integration isolation	7.4	90	0.40
Workload least-privilege	8.6	160	0.45
Secret rotation & scoping	8.2	70	0.40
Pipeline artifact signing	7.8	110	0.45
Data-zone segmentation	6.9	130	0.50
Backup isolation	9.3	60	0.30

6.2 Risk Baseline

With no additional controls, the estimated crown-jewel any-compromise probability is $\hat{\pi}_J = 9.6\%$ per year. The resulting annualized loss distribution has an expectation of US$1.9M, a 95th percentile of US$16.0M, and a 99th percentile of US$39.0M (Fig. 2). The long tail—an order of magnitude above the mean—illustrates why a point estimate is inadequate for capital and risk-appetite decisions.

6.3 Edge Criticality

Removing each edge in turn and re-estimating $\hat{\pi}_J$ gives a path-aware criticality ranking (Fig. 3). The two highest-criticality transitions both lie on the third-party integration path (vendor integration → pipeline identity, and foothold → vendor integration), each reducing crown-jewel compromise probability by more than 6 percentage points if neutralized. Notably, these transitions correspond to individually moderate weaknesses; a severity-ordered view that fixates on the single highest-CVSS item (backup isolation, CVSS 9.3) misranks them, because that item sits on a lower-probability tail path.

6.4 Path-Aware vs. Severity-Ordered Remediation

We compare two control-selection strategies under a fixed annual budget of US$300k (Table 2, Fig. 4). The severity-ordered strategy selects the highest-CVSS controls that fit the budget, as a conventional vulnerability-management queue would. The path-aware strategy uses the marginal-loss optimizer of Section 5.4. Severity-ordering spends US$250k and reduces expected annual loss by 41.5%. Path-aware selection spends US$280k and reduces expected annual loss by 85.2%—roughly double the risk reduction for a comparable outlay—by prioritizing vendor isolation and secret-rotation controls that sever the dominant paths, controls that the severity queue defers in favour of the single highest-scoring finding.

Table 2
Path-Aware vs. Severity-Ordered Remediation (B = $300k)

Strategy	Spend ($k)	Expected loss ($M)	Loss reduction
Baseline (no spend)	0	1.89	—
Severity-ordered (CVSS)	250	1.11	41.5%
Path-aware (ours)	280	0.28	85.2%

7 Discussion

7.1 Why Path-Awareness Changes the Decision

The evaluation reproduces, in quantitative form, a pattern long observed informally by defenders: the weakness that matters most is rarely the one with the highest intrinsic score, but the one that most increases an adversary's reach to critical assets. By carrying likelihood all the way to a monetary loss distribution and optimizing against it, the method turns that intuition into a defensible, auditable figure that a risk committee can compare against risk appetite and capital.

7.2 Governance, Validation, and Limitations

The method's outputs are only as good as its inputs. Edge likelihoods and loss magnitudes are estimates; the system is therefore designed as decision support, with every run logged for independent validation under model-risk governance [2] and with the AI components bounded so the quantitative core stays inspectable [4]. Key limitations include: the independence assumption on edge realizations, which ignores common-cause correlations and should be extended with correlated sampling; sensitivity of tail estimates to lognormal magnitude parameters, which argues for calibration against historical loss data and expert-elicitation discipline; and the synthetic nature of this illustrative study, which demonstrates the mechanism but does not validate it on production telemetry. We also treat the adversary as a likelihood process rather than a strategic optimizer; game-theoretic extensions are a natural direction.

7.3 Future Work

Priorities are: validation on anonymized production posture data with back-testing against realized incidents and control changes; correlated and time-varying edge models; calibration of the GNN likelihood scorer with uncertainty quantification; and integration of the loss distribution into enterprise capital and risk-appetite processes.

8 Conclusion

We presented an attack-path-aware risk-quantification architecture for cloud-native financial data platforms that unifies posture telemetry, ATT&CK-tagged attack-graph reasoning, and FAIR-based monetary loss modelling, with AI used in a deliberately bounded, inspectable role. On an illustrative

platform, propagating likelihood to a loss distribution and optimizing controls against it roughly doubled risk reduction at a comparable budget relative to severity-ordered remediation, and surfaced a low-severity third-party path as the dominant driver of crown-jewel exposure. The method is intended as a governed decision-support instrument for model-risk and operational-risk functions—bringing vulnerability management and financial risk quantification into a single, auditable loop—rather than as an autonomous control system.

Declarations

Conflict of interest

The author declares no conflict of interest. The views expressed are the author's own and do not necessarily represent those of the employer.

Ethics approval

Not applicable. The study uses only synthetic data and involves no human participants or personal data.

Funding

[Please state any funding received for this work, or write "No funding was received for conducting this study."]

Author Contribution

G.R. conceived the study and designed the methodology and system architecture. G.R. developed the attack-graph, Monte Carlo propagation and FAIR loss models, implemented the software, carried out the illustrative evaluation and formal analysis, and prepared Figures 1–4 and Tables 1–2. G.R. wrote the original draft and reviewed and edited the manuscript. As the sole author, G.R. takes full responsibility for the content of the manuscript.

Data availability

All data used in the illustrative study are synthetic and fully specified in Section 6 and Table 1; no proprietary, customer, or production data were used. The attack-graph model, Monte Carlo propagation, FAIR loss parameters, and control-selection procedure are described in sufficient detail to reproduce the reported figures.

References

1. Bilot, T., El Madhoun, N., Al Agha, K., Zouaoui, A.: Graph neural networks for intrusion detection: a survey. IEEE Access 11, 49114–49139 (2023)
2. Board of Governors of the Federal Reserve System, Office of the Comptroller of the Currency: Supervisory Guidance on Model Risk Management. SR 11 – 7 / OCC 2011-12 (2011)
3. Cloud Security Alliance: Cloud-Native Application Protection Platform (CNAPP): Reference Architecture. CSA (2023)
4. European Parliament and Council of the European Union: Regulation (EU) 2024/1689 laying down harmonised rules on artificial intelligence (Artificial Intelligence Act). Off. J. Eur. Union (2024)
5. Federal Financial Institutions Examination Council: FFIEC Information Technology Examination Handbook: Information Security. FFIEC (2016)
6. FIRST: Common Vulnerability Scoring System v3.1: Specification Document. Forum of Incident Response and Security Teams (2019)
7. Gupta, M., Akiri, C., Aryal, K., Parker, E., Praharaj, L.: From ChatGPT to ThreatGPT: impact of generative AI in cybersecurity and privacy. IEEE Access 11, 80218–80245 (2023)
8. Hubbard, D.W., Seiersen, R.: How to Measure Anything in Cybersecurity Risk, 2nd edn. Wiley, Hoboken (2023)
9. Noel, S., Jajodia, S.: Managing attack graph complexity through visual hierarchical aggregation. In: Proceedings of the ACM Workshop on Visualization and Data Mining for Computer Security, pp. 109–118 (2004)
10. Phillips, C., Swiler, L.P.: A graph-based system for network-vulnerability analysis. In: Proceedings of the New Security Paradigms Workshop (NSPW), pp. 71–79 (1998)
11. Poolsappasit, N., Dewri, R., Ray, I.: Dynamic security risk management using Bayesian attack graphs. IEEE Trans. Dependable Secure Comput. 9(1), 61–74 (2012)
12. Sheyner, O., Haines, J., Jha, S., Lippmann, R., Wing, J.M.: Automated generation and analysis of attack graphs. In: Proceedings of the IEEE Symposium on Security and Privacy, pp. 273–284 (2002)
13. Strom, B.E., Applebaum, A., Miller, D.P., Nickels, K.C., Pennington, A.G., Thomas, C.B.: MITRE ATT&CK: Design and Philosophy. Technical report, MITRE Corporation (2020)
14. The Open Group: Risk Analysis (O-RA), Version 2.0 (Factor Analysis of Information Risk, FAIR). Open Group Standard C20B (2021)
15. Wang, L., Jajodia, S., Singhal, A., Cheng, P., Noel, S.: k-zero day safety: a network security metric for measuring the risk of unknown vulnerabilities. IEEE Trans. Dependable Secure Comput. 11(1), 30–44 (2014)
16. Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., Yu, P.S.: A comprehensive survey on graph neural networks. IEEE Trans. Neural Netw. Learn. Syst. 32(1), 4–24 (2021)

Figures

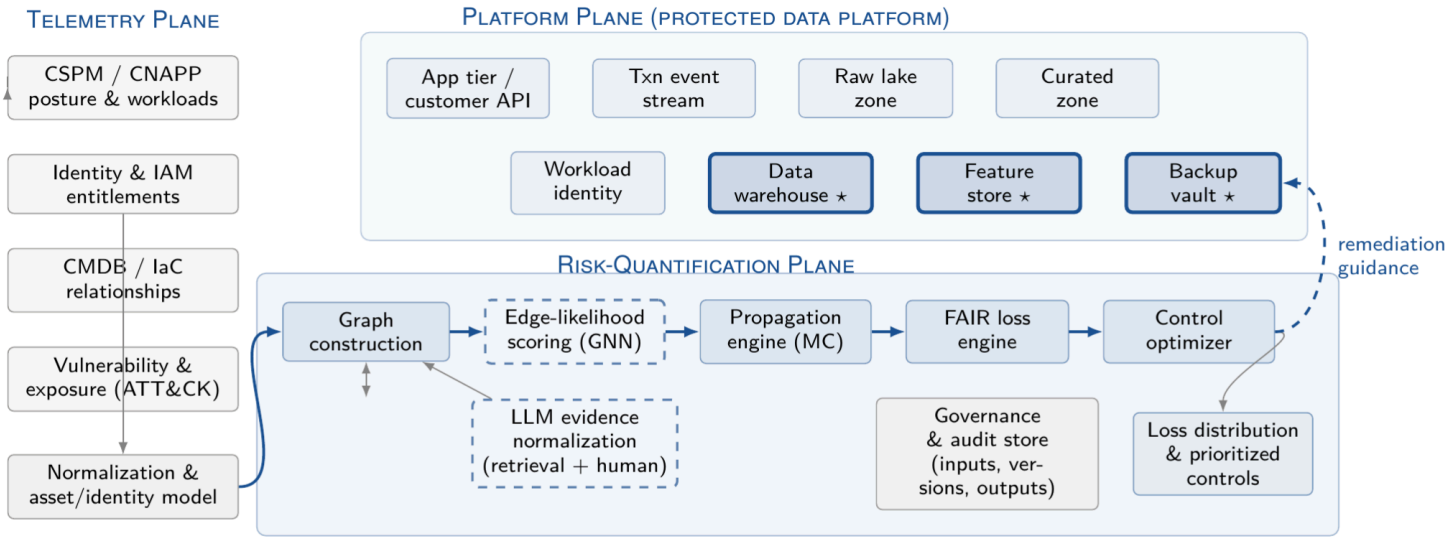


Figure 1

End-to-end architecture. Posture, identity, and configuration telemetry (telemetry plane) feed a graph-construction service that builds an ATT&CK-tagged attack graph over the cloud-native data platform (platform plane). The risk-quantification plane scores edge likelihoods, propagates crown-jewel compromise probability, converts it to a monetary loss distribution via a FAIR engine, and selects controls by marginal loss reduction. The AI components (dashed) are bounded: a GNN scores edges and an LLM normalizes unstructured evidence under retrieval grounding and human validation.

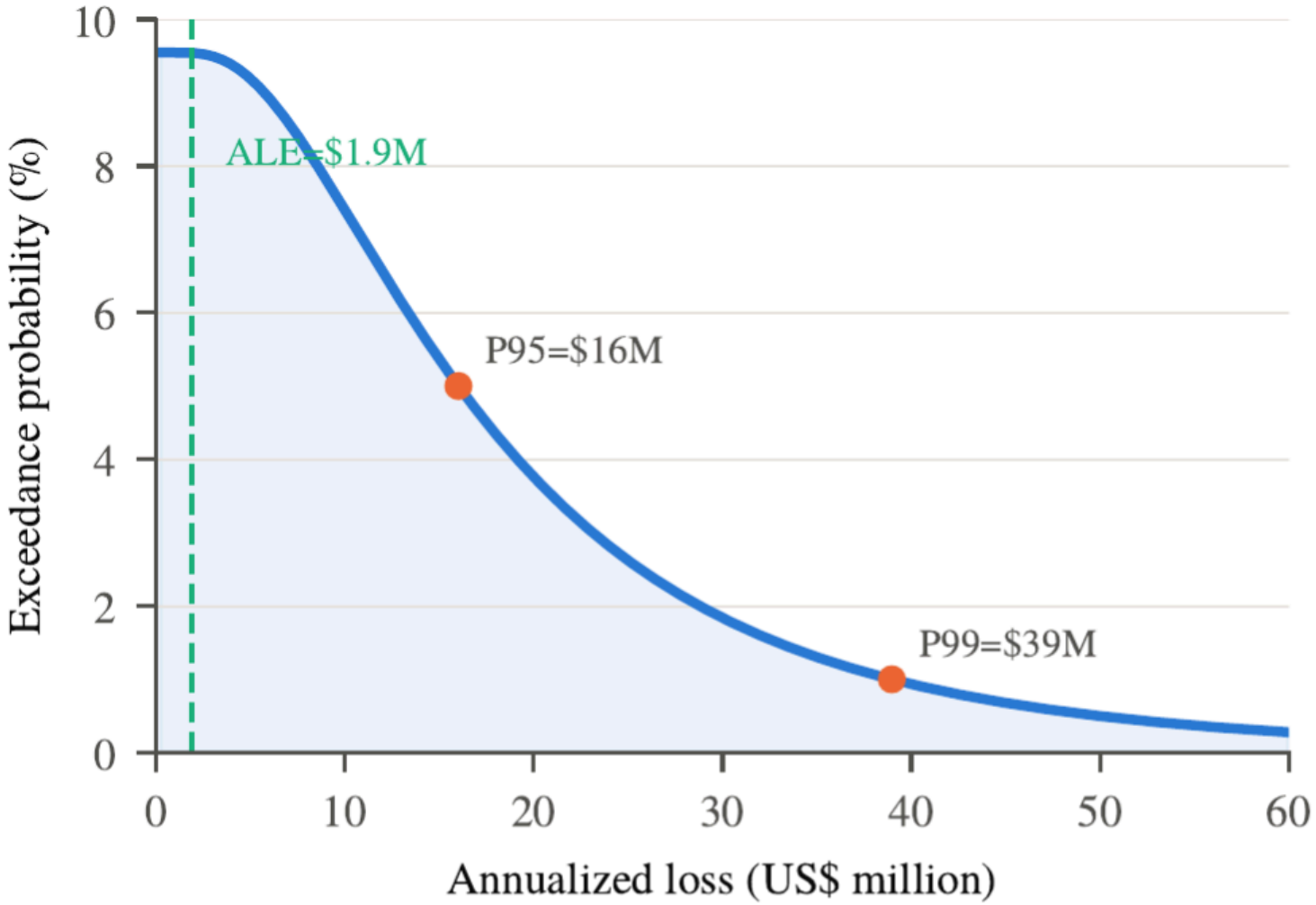


Figure 2

Baseline annualized loss-exceedance curve for the illustrative platform. The expected annual loss (ALE) is modest, but the distribution has a heavy tail: the 99th-percentile loss is roughly twenty times the mean, motivating distribution-based rather than point risk reporting.

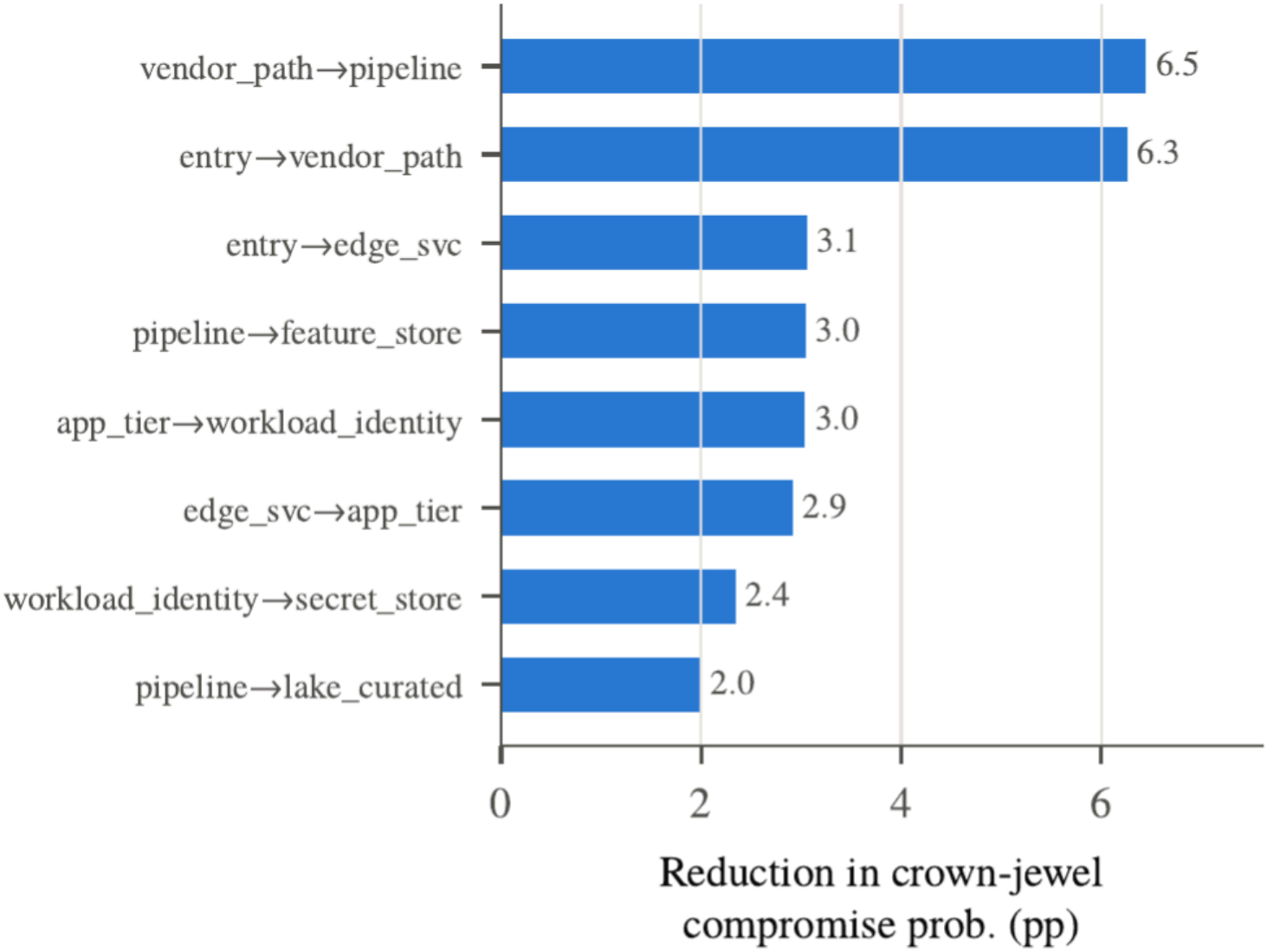


Figure 3

Top-ranked edge criticality: reduction in crown-jewel compromise probability (percentage points) if each transition is neutralized. The dominant transitions lie on the third-party integration path, which carries only moderate individual severity.

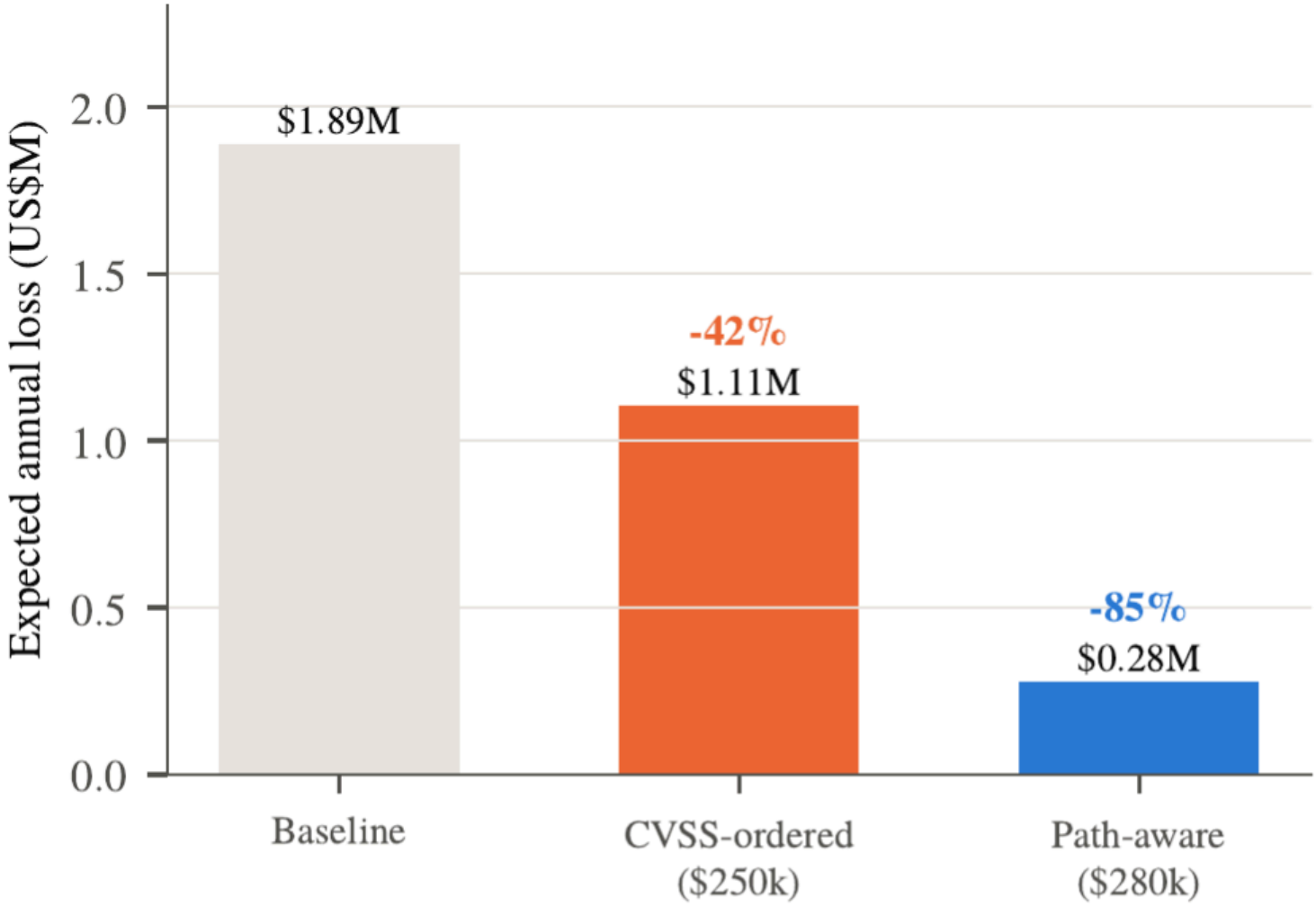


Figure 4

Expected annual loss under two remediation strategies at a comparable budget. Path-aware selection achieves roughly double the loss reduction of severity-ordered selection by targeting the controls that attenuate crown-jewel paths.