From Policy to Enforcement: An Architectural Pattern for Governed AI Execution

Warren Stockdale MSc.

United Kingdom

Abstract

As artificial intelligence systems transition from passive decision-support tools to active, agentic systems capable of initiating real-world actions, existing approaches to AI governance exhibit structural limitations. Regulatory frameworks such as the EU Artificial Intelligence Act (European Commission, 2024), the NIST AI Risk Management Framework (NIST, 2023), and UK National Cyber Security Centre guidance (NCSC, 2024) require auditability, human oversight, risk management, and control, yet deliberately avoid prescribing technical architectures for achieving these outcomes.

This paper identifies five structural failure modes in policy-only governance regimes—detection lag, provenance gaps, human review bottlenecks, adversarial evasion, and compliance theatre—and proposes an architectural pattern, the AI Execution Control Plane, that enforces governance at the point of execution rather than relying on retrospective assurance. The primary contribution is the pattern itself: a composable set of design primitives grounded in capability-based security, reference monitor design, and mandatory access control. AERIE (Agent Execution and RAPTOR-Governed Intent Envelope) is presented as a reference architecture instantiating this pattern, demonstrating how structured human intent, scoped execution authority, non-bypassable policy enforcement, and immutable decision artefacts can translate regulatory expectations into enforced technical controls.

Keywords: AI governance, agentic AI, execution control plane, capability-based security, policy enforcement, human oversight, regulatory compliance

Contents

1 Introduction

AI governance has become a central concern for regulators, organisations, and system designers. Recent years have seen a convergence of international frameworks emphasising trustworthy, controllable, and auditable AI systems (European Commission, 2024; NIST, 2023; NCSC, 2024). Despite differences in jurisdiction and legal force, these frameworks share a common characteristic: they specify outcomes, not architectures.

Organisations are therefore left to determine how requirements such as human oversight, traceability, proportionality, and accountability should be implemented in practice. In many cases, governance has been addressed through policy documents, risk registers, approval workflows, and post-deployment monitoring. While these measures provide organisational assurance, they do not prevent unauthorised or unsafe AI actions from occurring at runtime.

This paper contends that the governance problem is fundamentally architectural. As AI systems gain the ability to act autonomously—invoking tools, modifying systems, or initiating transactions—governance must move from observation to enforcement. The central thesis is that governance should be embedded directly into the execution path of AI systems, rather than layered on as an external control. The primary contribution is an architectural pattern—the AI Execution Control Plane—together with a reference architecture (AERIE) that instantiates its design primitives. The paper proceeds by examining the structural limitations of retrospective governance (Section 2), analysing the gap between regulatory intent and technical implementation (Section 3), defining the execution control plane and its primitives (Sections 4–6), demonstrating regulatory alignment by construction (Section 7), comparing the approach with contemporary alternatives (Section 8), and discussing limitations and future work (Section 9).

2 Limits of Policy-Based and Post-Hoc AI Governance

2.1 Structural weaknesses of retrospective governance

Policy-based governance relies on documented rules, acceptable-use statements, contractual assurances, and periodic audits to constrain AI behaviour. Post-hoc governance supplements this with logging, telemetry, and retrospective human review. These approaches align with established compliance practices and impose relatively low upfront technical cost. However, practitioner postmortems, regulatory enforcement actions, and a growing body of empirical research identify five structural failure modes that render this model inadequate for agentic AI systems.

Detection lag and irreparable harm. AI systems operate at machine speed; governance processes operate at human speed. In agentic contexts—where a single prompt can trigger chains of tool invocations, API calls, and state mutations—the window between action and detection may be measured in milliseconds, yet consequences may include regulatory fines, data breaches, or reputational damage. The OWASP Top 10 for LLM Applications identifies this temporal mismatch as a core risk factor (OWASP, 2025), and industry analyses of agentic deployments corroborate the problem (Zenity, 2025).

Provenance gaps and forensic insufficiency. Logs may be incomplete, lack turn-level granularity, or fail to preserve causal linkage between model reasoning and downstream actions. In healthcare, research indicates that AI audit trails are frequently not integrated into clinical incident processes, undermining retrospective accountability (Murdoch, 2021). In financial services, vendor opacity and trade-secret claims can prevent meaningful internal audits (Langer et al., 2025).

Human review bottlenecks and automation bias. Human reviewers under time pressure are subject to overload and automation bias. Research demonstrates that decision-makers frequently defer to AI recommendations even when incorrect, a phenomenon that intensifies under cognitive load (Schemmer et al., 2022). Post-hoc review further suffers from *authority drift*: the nominal power to override AI decisions erodes in practice through organisational incentives and workflow design.

Adversarial evasion. Adversarial techniques erode the effectiveness of post-hoc controls. Indirect prompt injection can manipulate large language models via untrusted external content, bypassing intended guardrails (Greshake et al., 2023). Adversarial prompting can reliably subvert model instructions and safety constraints through crafted inputs (Perez and Ribeiro, 2022). These techniques operate beneath the detection threshold of output-filtering and log-review mechanisms.

Compliance theatre. Perhaps the most consequential failure mode is the presence of governance artefacts—policies, registers, checklists—that satisfy procedural requirements without constraining system behaviour. Practitioners and sociotechnical researchers describe this variously as ‘governance theatre’ or ‘compliance theatre’ (Guidepost Solutions, n.d.). The term is used here as a practitioner construct rather than a formal academic concept, but it captures a widely recognised pattern: governance that exists on paper but not in practice.

2.2 Documented governance failures in deployed systems

The insufficiency of policy-only governance is not merely theoretical. The following cases, illustrative rather than exhaustive, demonstrate that formal governance measures can fail to prevent serious harm.

Robodebt (Australia). An automated debt-recovery programme used data-matching and income-averaging algorithms to generate debt notices. The scheme operated with bureaucratic policies and ministerial approvals, yet produced unlawful determinations and severe distress among welfare recipients. The Royal Commission found systemic governance failures despite extensive documentation (Royal Commission into the Robodebt Scheme, 2023).

Volkswagen emissions software. Volkswagen maintained formal compliance functions and public environmental commitments, yet software was engineered to evade laboratory emissions tests—a widely cited case of software-enabled non-compliance despite an extensive governance apparatus (Ewing, 2017).

Recruitment algorithm bias. Amazon's early recruiting prototype reportedly exhibited systematic gender discrimination and was ultimately abandoned, illustrating limitations of upstream data governance despite internal policy frameworks (Dastin, 2018).

Clinical decision support. IBM Watson Health projects were promoted with clinical-validation claims, but evaluations revealed poor accuracy and overpromising of capabilities. While detailed peer-reviewed assessments of specific deployments remain limited, journalistic and practitioner analyses document a persistent gap between governance claims and clinical outcomes (Strickland, 2019; Ross and Swetlitz, 2018).

These cases share a common structural feature: governance artefacts existed but did not prevent harm because they operated outside the execution path of the systems they were intended to constrain.

2.3 The excessive agency problem in agentic systems

The emergence of tool-calling, multi-step AI agents intensifies these failure modes. The OWASP Top 10 for LLM Applications identifies excessive agency—granting more functionality, permissions, or autonomy than necessary—as a principal vulnerability (OWASP, 2025). Security researchers have documented reproducible attack classes exploiting over-privileged agents, including confused-deputy abuses, tool and plugin backdoors, context-poisoning attacks, and classical authorisation failures exposed through LLM tooling (Cobalt, 2025; Entro Security, 2025).

Industry reports describe incidents including malicious plugins exfiltrating email contents, AI assistants manipulated into approving fraudulent transactions, and large-scale campaigns exploiting exposed model endpoints (Protecto, 2025; Techzine, 2026). While many of these accounts originate from vendor security analyses rather than peer-reviewed research, the consistency of reported patterns across independent sources supports the underlying threat model. Conventional RBAC and synchronous monitoring assume predictable, human-driven flows; agentic systems break that assumption by operating multi-step reasoning chains, consulting mutable memory, invoking external tools, and making decisions without per-step human approval. Defences based solely on static permission reviews, network-level controls, or output filtering appear insufficient without fine-grained capability gating and runtime policy enforcement.

3 Regulatory Intent and Architectural Silence

3.1 Deliberate non-prescriptiveness across frameworks

Major AI governance frameworks deliberately avoid prescribing technical architectures. The EU AI Act imposes binding obligations on high-risk systems—including record-keeping, human oversight, and risk management—while leaving implementation choices to system designers (European Commission, 2024). The NIST AI RMF adopts an outcomes-based, voluntary approach structured around governance, mapping, measurement, and management functions (NIST, 2023). UK NCSC guidance emphasises secure-by-design principles without mandating specific enforcement mechanisms (NCSC, 2024).

This architectural silence is intentional, preserving technological neutrality. However, it creates what practitioners have termed the *design gap*: the recurring deficiency where organisations possess governance documents, committees, and checklists without mapping them to concrete sensing, decision, and actuation mechanisms in engineering workflows (Kavanagh, 2023). As characterised in practitioner literature, the design gap distinguishes governance as narrative from governance as control system. The absence of prescribed architectures does not imply the absence of architectural necessity.

3.2 Emerging signals toward architectural enforcement

Despite the non-prescriptive framing, evidence from recent years suggests movement from high-level principles toward obligations that presuppose—and in some cases require—technical controls as the primary means of compliance.

Management-system standards such as ISO/IEC 42001 and risk guidance such as ISO/IEC 23894 call for traceability, impact assessments, data governance, and lifecycle controls that are practically implementable only via metadata, lineage systems, and telemetry. Certification under these standards requires technical artefacts and demonstrable controls (ISO, 2023). NIST's cybersecurity overlays map AI-specific controls to SP 800-53 and emphasise continuous monitoring and integration of AI risk into security baselines—an architectural framing that implies policy enforcement points and runtime defences (NIST, 2023).

Jurisdictional regulators appear to be reinforcing this trajectory. The EU's risk-based regime and newly enacted state-level laws in the United States press for demonstrable controls rather than aspirational statements. Federal agency playbooks are explicitly charged with operationalising risk rubrics through inventories, metrics, and technical controls—moving compliance into pipelines, SRE processes, and procurement artefacts (GSA, 2025).

Across these materials, a consistent set of enforcement mechanisms is either explicitly recommended or directly implied: model and dataset attestations; provenance and lineage metadata; immutable telemetry and tamper-evident logging; runtime monitoring and anomaly detection; policy enforcement points and standardised control APIs; compute and access controls including sandboxing and gated deployment; and formalised interfaces for auditors and regulators. It is argued that these mechanisms are the operational predicates for demonstrating conformity, not optional best practices (ISO, 2023; NIST, 2023).

4 The AI Execution Control Plane

4.1 Conceptual foundation

In systems engineering, control planes determine whether actions may occur, while data planes carry them out. Applying this distinction to AI systems yields the concept of an AI Execution Control Plane: a non-bypassable architectural layer that mediates all AI-initiated actions.

Rather than observing AI behaviour after execution, the control plane evaluates proposed actions before they occur, enforcing authorisation, policy compliance, and evidence generation as prerequisites. The concept draws on mandatory access control, reference monitor design, and capability-based security, extending them to the specific requirements of agentic AI governance.

4.2 Absence of a canonical definition

A review of academic literature reveals no canonical definition of an 'AI execution control plane'. Relevant concepts appear across capability-based security, mandatory access control, MLOps orchestration, human-in-the-loop gating, and provenance systems (Hardy, 1985; Watson et al., 2010; Miller, Yee and Shapiro, 2003).

Industry usage of the term often conflates orchestration with enforcement, prioritising integration velocity over non-bypassable control. Some vendor implementations claim structural impossibility of disallowed actions—a design-time property—while simultaneously offering runtime policy checks, an execution-plane capability. This conflation requires scrutiny of what is actually enforced, where, and how (Salunkhe, n.d.; Hoop, n.d.). These industry sources, while not peer-reviewed, illustrate the terminological confusion motivating the definition of a unifying pattern grounded in established security principles.

5 RAPTOR as the Human Governance Interface

Human oversight is frequently interpreted as review or escalation. Research suggests that such interpretations fail to constrain AI behaviour at runtime (Langer et al., 2025). For oversight to function as a genuine control, human intent must be translated into a form that machines can enforce deterministically.

RAPTOR (Role, Aim, Parameters, Tone, Output, Review) serves this function by encoding human intent as a structured, reviewable artefact. Within the execution control plane, RAPTOR declarations are evaluated before execution authority is granted. Intent therefore becomes an authorisation primitive, binding human judgement directly to AI action.

This approach represents a broader class of governance mechanisms in which human intent is encoded as a structured, enforceable artefact rather than expressed as natural-language policy. The distinction is significant: advisory oversight relies on humans detecting and correcting problems after they manifest, whereas structured intent specification prevents unauthorised actions from executing in the first instance. RAPTOR operationalises this principle by making the separation between cognition and authority explicit: AI systems may reason, plan, and propose freely, but may not act without explicitly issued authority.

RAPTOR is one instantiation of structured intent specification; alternative formalisms for encoding human intent into machine-enforceable constraints may be equally viable. The architectural contribution of this paper lies in the principle of intent-as-authorisation, not in the specific schema.

6 Design Primitives of a Governed AI Execution Plane

The execution control plane is realised through a small set of composable primitives, each grounded in established security engineering:

Intent Envelope—an immutable representation of authorised purpose and constraints, derived from structured intent declarations. The intent envelope binds human-specified objectives and boundaries to a machine-evaluable format, serving as the authoritative source of permitted action scope.

Governance and Policy Engine—a deterministic evaluation mechanism assessing proposed actions against intent, organisational policy, regulatory constraints, and runtime context. This component functions as a reference monitor in the classical security sense, providing complete mediation of execution requests.

Capability Tokens—scoped, time-bound execution authority encoded as capabilities, drawing on capability-based security research (Hardy, 1985; Watson et al., 2010; Miller, Yee and Shapiro, 2003). Capability tokens embody least privilege in a form that is both enforceable and auditable.

Trust Boundary—a hard separation between AI cognition and real-world action, preventing unauthorised execution. This boundary ensures that reasoning and planning do not constitute action, and that only explicitly authorised operations cross into the execution domain.

Immutable Execution Ledger—an append-only, tamper-resistant record binding intent, policy decision, capability issuance, and execution outcome. The ledger provides the evidentiary basis for auditability and accountability requirements across regulatory frameworks.

Together, these primitives are designed to transform governance from a managerial assurance mechanism into an architecturally enforced property. The degree to which this transformation is achievable in practice is discussed in Section 9.

7 Regulatory Alignment by Construction

Mapping execution control plane properties to regulatory expectations suggests that compliance can emerge structurally from architecture rather than being applied retrospectively:

Human oversight is enforced through authority issuance rather than advisory review, addressing the EU AI Act's requirement for meaningful human control (Article 14) and NIST's Govern function expectation that override authority be defined with auditable records.

Auditability is intrinsic, with complete decision lineage preserved as immutable artefacts—fulfilling the Act's tamper-resistant logging requirements (Article 12) and ISO/IEC 42001's expectation of traceable records.

Least privilege is continuous, enforced through ephemeral, scoped capabilities that align with NCSC secure-by-design guidance and NIST's emphasis on proportionate access controls.

Risk proportionality is dynamic, applied per action rather than statically at system level, supporting the Act's proportionality language and NIST's profile-driven evidence model.

Accountability is explicit and attributable rather than inferred post-hoc, addressing GDPR accountability obligations and the Act's conformity assessment requirements (Article 43).

This alignment-by-construction approach also addresses concerns identified in recent work: that the measurement problem for oversight effectiveness and the new attack surfaces oversight mechanisms can introduce are both structurally mitigated when oversight is embedded in the execution path rather than layered externally (Langer et al., 2025; Ditz et al., 2025). It is acknowledged that demonstrating full alignment-by-construction in practice would require formal verification of the policy engine's completeness and correctness—a challenge discussed in Section 9.

8 Comparison with Contemporary Agent Frameworks and Governance Approaches

8.1 Agent framework security models

Contemporary agent frameworks—including LangChain, AutoGen, CrewAI, and similar orchestration platforms—manage behaviour through guardrails, monitoring, and orchestration. These approaches detect or discourage undesirable actions but cannot guarantee prevention; they remain probabilistic and reactive.

Industry security analyses identify convergence around three authorisation models: RBAC for role grouping, attribute-based and policy-based access control (ABAC/PBAC) for context-aware decisions, and capability-style tokens for fine-grained tool invocation. Reported practices include per-agent scoping, just-in-time elevation, automated token rotation, and vaulted secret storage (Obsidian Security, 2025;

WorkOS, 2025; Composio, 2026). At runtime, enforcement typically mixes cryptographic and platform controls: short-lived tokens bind intent and actor, centralised policy engines make contextual decisions, and sandboxing constrains tool execution.

However, these controls operate at the framework level rather than as a non-bypassable architectural layer. Industry analyses suggest that default or persistent permissions concentrate risk: over-privilege expands blast radius; recursive delegation without scope attenuation enables cross-agent privilege escalation; and persistent tokens create footholds for supply-chain attacks. Standard authorisation checks do not verify whether the observed action matches the original intent (Lumos, 2025; Okta, 2025; Acuvity, 2025). These findings derive primarily from vendor analyses and practitioner reports and should be weighted accordingly.

8.2 Governance-by-design approaches

A parallel class of research and practitioner initiatives asserts governance by design: embedding governance requirements into architecture, data pipelines, and development lifecycles so that safety and compliance become structural properties. These approaches draw on high-reliability engineering and include model cards (Mitchell et al., 2019), dataset documentation, CI/CD release gates, lineage systems, and policy-as-code frameworks (Kavanagh, 2023).

Governance-by-design raises baseline assurance and can simplify audits by reducing the set of unsafe states a system can enter. However, it primarily addresses design-time and build-time governance. Emergent behaviour, long-tail interactions, and adversarial inputs at inference time cannot be fully anticipated at design time. Runtime enforcement therefore remains necessary as a complementary layer.

8.3 The execution control plane distinction

It is argued that the execution control plane differs from both categories in a fundamental respect: it manages authority, not behaviour. Actions that are not authorised cannot occur. Orchestration coordinates execution; control determines permission. Table 1 summarises the key differences.

Table 1: Comparison of governance enforcement approaches.

Property	**Agent Frameworks**	**Governance by Design**	**Execution Control Plane**
Enforcement point	Framework-level guardrails	Build/deploy gates	Runtime execution path
Bypass resistance	Probabilistic (can be evaded)	Pre-deployment only	Non-bypassable by construction
Authority model	Static RBAC or broad tokens	Policy-as-code at CI/CD	Per-action capability tokens
Human oversight	Escalation workflows	Approval gates at release	Authority issuance at intent
Audit evidence	Logs (mutable, optional)	Build artefacts, model cards	Immutable execution ledger
Scope of control	Model outputs	Development lifecycle	Every AI-initiated action

This comparison suggests that existing approaches address important parts of the governance problem but leave a structural gap at the point where AI systems take real-world actions. The execution control plane addresses this gap by operating at the boundary between cognition and consequence.

9 Limitations and Future Work

This paper proposes an architectural pattern rather than a fully standardised or empirically validated implementation. While the execution control plane is grounded in established security principles, several limitations must be acknowledged.

First, performance and latency impacts of fine-grained, per-action authorisation require empirical evaluation at scale. The operational burden of structured intent specification may prove significant in high-throughput environments. Prototype deployments in regulated settings are needed to quantify these trade-offs.

Second, formal verification of policy engines and ledger integrity is an open research problem. While tamper-evident logging and cryptographic anchoring are well-understood primitives, their composition within an AI execution context warrants formal analysis to establish the conditions under which the 'non-bypassable' property holds.

Third, human factors—including usability of structured intent specification, cognitive load of RAPTOR declarations, and design of governance workflows—warrant systematic user studies to avoid trading one form of governance burden for another. Recent work highlights shortfalls in standardised assessment procedures and interoperable attestations, where policy demands outpace available technical standards (Agarwal and Nene, 2025).

Fourth, cross-provider provenance in federated training, robust watermarking for generative outputs, secure attestations for models and datasets, and harmonised telemetry schemas remain open challenges affecting practical deployment of execution-level controls.

Fifth, this paper relies in several areas on industry reports, vendor security analyses, and practitioner commentary as evidence for threat models and failure modes. While these sources reflect operational experience, they have not undergone peer review and may carry vendor bias. As the academic literature on agentic AI security matures, stronger empirical grounding for these claims will be both possible and necessary.

Sixth, RAPTOR is presented as one instantiation of structured intent specification. Comparative evaluation against alternative intent-encoding formalisms—and empirical assessment of whether structured intent specification delivers measurable governance improvements—remains an open question.

Future work should include prototype deployments in regulated environments, formal security proofs for execution boundary enforcement, comparative evaluation against alternative governance mechanisms, user studies of structured intent specification, and engagement with standards bodies developing architectural requirements for AI governance. The convergence of ISO, NIST, and jurisdictional regulators toward architectural enforcement suggests that voluntary adoption of these patterns may precede mandatory requirements.

10 Conclusion

This paper has argued that AI governance must evolve from assurance to enforcement. As AI systems assume operational responsibility—initiating actions, invoking tools, and modifying external systems—architectures that embed governance into the execution path become a necessity rather than a preference.

The evidence assembled here—from documented governance failures in deployed systems, to the structural limitations of policy-only regimes, to the emerging regulatory trajectory toward architectural controls—supports a clear conclusion: the gap between governance intent and governance enforcement is an architectural problem requiring an architectural solution.

The AI Execution Control Plane, instantiated through the AERIE reference architecture, provides a coherent response grounded in established security principles. By making governance an unavoidable property of the execution path, it offers a foundation for deploying agentic AI systems that are not merely capable, but defensible. The degree to which this pattern can be validated empirically and adopted at scale remains the subject of future work.

References

Acuvity (2025) *The Agent Integrity Framework: the new standard for securing autonomous AI*. [Industry report] Available at: https://acuvity.ai/ (Accessed: 2026).

Agarwal, S. and Nene, M.J. (2025) 'Shortfalls in standardised assessment procedures for AI governance', preprint.

Cobalt (2025) 'LLM vulnerability: excessive agency', *Cobalt Blog*. [Industry analysis] Available at: https://www.cobalt.io/blog/llm-vulnerability-excessive-agency (Accessed: 2026).

Composio (2026) *Secure AI agent infrastructure guide*. [Practitioner guide] Available at: https://composio.dev/blog/secure-ai-agent-infrastructure-guide (Accessed: 2026).

Dastin, J. (2018) 'Amazon scraps secret AI recruiting tool that showed bias against women', *Reuters*, 10 October.

DigitalDefynd (2026) *AI governance failures: case studies and lessons*. [Practitioner case review] Available at: https://digitaldefynd.com/ (Accessed: 2026).

Ditz, J. et al. (2025) 'Attack surfaces of human oversight in AI systems', *arXiv preprint*.

Entro Security (2025) 'Agentic AI and OWASP: research findings', *Entro Security Blog*. [Industry analysis] Available at: https://entro.security/blog/agentic-ai-owasp-research/ (Accessed: 2026).

European Commission (2024) Regulation (EU) 2024/1689 of the European Parliament and of the Council of 13 June 2024 laying down harmonised rules on artificial intelligence (Artificial Intelligence Act). *Official Journal of the European Union*.

Ewing, J. (2017) *Faster, Higher, Farther: The Volkswagen Scandal*. New York: W.W. Norton.

Greshake, K., Abdelnabi, S., Mishra, S., Endres, C., Holz, T. and Fritz, M. (2023) 'Not what you've signed up for: compromising real-world LLM-integrated applications with indirect prompt injection', *arXiv preprint*, arXiv:2302.12173.

GSA (2025) *AI guidance and resources*. Washington, DC: U.S. General Services Administration. Available at: https://www.gsa.gov/technology/government-it-initiatives/artificial-intelligence/ (Accessed: 2026).

Guidepost Solutions (n.d.) *AI governance: bridging the gap between policy and practice*. [Practitioner analysis] Available at: https://www.guidpostsolutions.com/ (Accessed: 2026).

Hardy, N. (1985) 'The KeyKOS architecture', *ACM SIGOPS Operating Systems Review*, 19(4), pp. 8–25.

Hoop (n.d.) *Why action-level approvals matter for AI policy enforcement*. [Vendor documentation] Available at: https://hoop.dev/ (Accessed: 2026).

ISO (2023) *ISO/IEC 42001:2023 — Artificial intelligence management system*. Geneva: International Organization for Standardization.

Kavanagh, D. (2023) 'The design gap in AI governance', *AI Governance Career Pro*. [Practitioner commentary] Available at: https://governance.aicareer.pro/ (Accessed: 2026).

Langer, M. et al. (2025) 'Measurement and effectiveness challenges for human oversight of AI', *arXiv preprint*.

Lumos (2025) 'Agentic AI: identity governance and management', *Lumos Blog*. [Industry analysis] Available at: https://www.lumos.com/ (Accessed: 2026).

Miller, M.S., Yee, K.-P. and Shapiro, J. (2003) 'Capability myths demolished', *Technical Report SRL2003-02*. Baltimore, MD: Johns Hopkins University.

Mitchell, M. et al. (2019) 'Model cards for model reporting', in *Proceedings of the Conference on Fairness, Accountability, and Transparency (FAT* '19)*. New York: ACM, pp. 220–228.

Murdoch, B. (2021) 'Privacy and artificial intelligence: challenges for protecting health information in a new era of medicine', *BMC Medical Ethics*, 22, article 122.

NCSC (2024) *Guidelines for secure AI system development*. London: National Cyber Security Centre.

NIST (2023) *Artificial intelligence risk management framework (AI RMF 1.0). NIST AI 100-1*. Gaithersburg, MD: National Institute of Standards and Technology.

Obsidian Security (2025) 'Security for AI agents', *Obsidian Security Blog*. [Industry analysis] Available at: https://www.obsidiansecurity.com/ (Accessed: 2026).

Okta (2025) 'Agent security and the delegation chain', *Okta Blog*. [Vendor analysis] Available at: https://www.okta.com/blog/ai/agent-security-delegation-chain/ (Accessed: 2026).

OWASP (2025) *OWASP Top 10 for Large Language Model Applications: LLM08 — Excessive Agency*. Available at: https://genai.owasp.org/ (Accessed: 2026).

Perez, F. and Ribeiro, I. (2022) 'Ignore this title and HackAPrompt: exposing systemic weaknesses of LLMs through a global scale prompt hacking competition', *arXiv preprint*, arXiv:2211.09527.

Protecto (2025) 'AI agents excessive agency risks', *Protecto Blog*. [Industry analysis] Available at: https://www.protecto.ai/ (Accessed: 2026).

Ross, C. and Swetlitz, I. (2018) 'IBM's Watson supercomputer recommended unsafe and incorrect cancer treatments, internal documents show', *STAT News*, 25 July.

Royal Commission into the Robodebt Scheme (2023) *Report*. Canberra: Commonwealth of Australia.

Salunkhe, S. (n.d.) 'AI control plane: governance before models, around agents', *LinkedIn Pulse*. [Practitioner commentary] Available at: https://www.linkedin.com/ (Accessed: 2026).

Schemmer, M., Hemmer, P., Kühl, N., Benz, C. and Satzger, G. (2022) 'Should I follow AI-based advice? Measuring appropriate reliance in human-AI decision-making', *arXiv preprint*, arXiv:2204.06916.

Strickland, E. (2019) 'IBM Watson, heal thyself: how IBM overpromised and underdelivered on AI health care', *IEEE Spectrum*, 2 April.

Techzine (2026) 'First large-scale LLMjacking generates tens of thousands of attacks', *Techzine / Pillar Security*. [Industry report] Available at: https://www.techzine.eu/ (Accessed: 2026).

Watson, R.N.M., Anderson, J., Laurie, B. and Kennaway, K. (2010) 'Capsicum: practical capabilities for UNIX', in *Proceedings of the 19th USENIX Security Symposium*. Washington, DC: USENIX Association, pp. 29–45.

WorkOS (2025) 'AI agent access control', *WorkOS Blog*. [Industry analysis] Available at: https://workos.com/blog/ai-agent-access-control (Accessed: 2026).

Zenity (2025) 'Securing AI where it acts: why agents now define AI risk', *Zenity Blog*. [Industry analysis] Available at: https://zenity.io/ (Accessed: 2026).