AI GOVERNANCE RESOURCE

# AI Agent Audit Program

## Companion Guide for CAEs & Audit Leadership

An executive companion to the AI Agent Audit Program & Risk Checklist — built for board reporting, engagement planning, resourcing decisions, and committee conversations.

This document is part of AI Agent Audit Program series. Use alongside AI Agent Audit Program & Risk Checklist.

Prepared by: Ishan Jain | ishan@jainishan.com | jainishan.com

Version 1.0 | June 2026

# 1. Why This Matters Now

Agentic AI adoption has outpaced the governance controls most organizations built for conventional software and even for earlier-generation chatbot/copilot tools. Two structural shifts make this an audit priority now, not a future-roadmap item:

## The risk has changed shape

Traditional software risk is largely about what a system fails to do correctly. Agentic AI risk is increasingly about what a system does that it was never explicitly told to do — taking an action, calling a tool, or making a decision that falls inside its technical permissions but outside its intended scope. This is a category Internal Audit has limited precedent for testing, which is exactly why a dedicated program (rather than folding it into general IT audit) is warranted.

## Two illustrative failure patterns

- **Excessive-agency incidents:** Multiple vendors and enterprises have reported cases where an AI coding or support agent, given broad tool access, deleted production data, issued unauthorized refunds, or modified records outside its intended task — not through malicious intent, but because nothing in its design prevented the action once it was technically possible. The common thread across public post-mortems is the same: permissions were broader than the task required, and no approval gate caught the action before it executed.
- **Indirect prompt injection:** Security researchers have repeatedly demonstrated that AI agents which read external content — emails, web pages, shared documents — can be manipulated by instructions hidden in that content, causing the agent to take actions the legitimate user never requested. This class of attack does not require breaching any traditional perimeter; it rides in on data the agent was already trusted to read.

## The regulatory and assurance bar is rising in parallel

Regulators and standards bodies have moved from guidance to binding obligation faster than most internal control environments have adapted (see Regulatory Horizon, next section). Boards are increasingly asking whether Internal Audit has independently tested AI agent controls — not just whether the business says it has.

### The budget/headcount argument

Every additional Tier 1 agent deployed without a corresponding increase in test capacity widens the gap between what the organization is exposed to and what Internal Audit can independently verify. The ask is not 'more audit for its own sake' — it is matching test coverage to a risk surface that is growing faster than most other parts of the control environment.

# 2. Maturity Model

*Turn the audit program from a pass/fail checklist into a multi-year progress narrative*

Rate each of the seven risk domains on a five-level maturity scale, independent of whether individual findings are open or closed. Maturity describes the strength and consistency of the control environment itself; findings describe specific instances where that environment fell short in a sampled test. The two are related but distinct — a domain can be rated 'Defined' even with one open medium finding, if the control design and consistency are otherwise sound.

## Maturity Scale

| | |
|---|---|
| **1 — Initial** | Ad hoc or absent. Controls exist only informally, inconsistently, or not at all. Risk is largely unmanaged and undocumented. |
| **2 — Developing** | Controls are emerging but undocumented or inconsistently applied across agents. Reliant on individual effort rather than process. |
| **3 — Defined** | Controls are documented, approved, and consistently applied across in-scope agents. Repeatable but not yet systematically measured. |
| **4 — Managed** | Controls are measured against defined metrics/thresholds; performance is monitored and exceptions are tracked to closure. |
| **5 — Optimized** | Controls are continuously improved using monitoring data, incident learnings, and emerging threat intelligence; proactive, not reactive. |

# 3. Regulatory Horizon

## EU AI Act — Phased Applicability

| Date | Milestone | What It Means for Deployers |
|---|---|---|
| 2024 (in force) | **EU AI Act enters into force** | Regulation (EU) 2024/1689 entered into force August 1, 2024, establishing a phased implementation timeline rather than a single effective date. |
| Feb 2025 | **Prohibited practices & AI literacy duties** | Bans on prohibited AI practices (e.g., certain manipulative or social-scoring systems) and baseline AI literacy obligations for providers/deployers became applicable. |
| Aug 2025 | **GPAI obligations applicable** | General-Purpose AI model providers became subject to transparency, technical documentation, and (for high-impact models) systemic-risk obligations under Title V. |
| Aug 2026* | **Core high-risk system obligations applicable** | The bulk of Title III high-risk system obligations (risk management, data governance, human oversight, logging, conformity assessment) become applicable to in-scope systems — the most consequential date for most enterprise deployers. |
| Aug 2027* | **Extended transition for certain high-risk categories** | Additional high-risk system categories embedded in regulated products (e.g., certain safety components) reach full applicability under extended transition provisions. |

*Note: this table reflects the headline phased-applicability dates published at the time of this guide's preparation. The European Commission and AI Office continue to issue implementing and delegated acts that can adjust scope and timing for specific categories — reconfirm before using these dates.*

## Why these dates matter

- The August 2026* milestone is the most consequential near-term date for most enterprise deployers of high-risk AI systems — it converts what is currently leading practice (per the underlying audit program) into binding legal obligation for in-scope use cases.
- Sector-specific regulators (financial services, healthcare, insurance) are layering additional AI-specific supervisory expectations on top of the EU AI Act baseline — confirm with sector/compliance counsel which apply to this organization.
- ISO/IEC 42001 certification is increasingly referenced by regulators and customers as a credible way to demonstrate AI governance maturity ahead of binding deadlines — worth tracking as a voluntary readiness signal even where not mandatory.

### Action for the CAE

Use high-risk applicability dates as a forcing function in resourcing conversations. Framing test coverage gaps against a fixed regulatory date is materially more persuasive to a Committee than framing them against an internal risk appetite statement alone.

**Aug 2026 remains the legally binding date for high-risk system obligations under the text currently in force; however, EU negotiators reached an agreement to defer these obligations to Dec 2027 (with a parallel deferral for Annex I product-embedded systems from Aug 2027 to Aug 2028). These changes take legal effect only once formally adopted and published in the Official Journal — until then, Aug 2026 is the operative compliance date. Confirm current status against the Official Journal before using either date in a board deck.*

# 4. Resourcing & Skills Guidance

*Standards basis: ISO 42001 Clause 7.2 (competence); NIST AI RMF GOVERN 2.3; EU AI Act Art. 4 (AI literacy).*

Most of this program is executable by a competent generalist Internal Audit team using skills they already have — access review, vendor risk, policy testing, log analysis. Two domains are the exception and should be planned for deliberately.

| Domain | Skill Level Required | Detail |
|---|---|---|
| **Governance & Accountability** | General IA staff (no specialized AI skill required) | Standard policy/process audit skills: charter review, RACI testing, change-management walkthroughs. Any experienced auditor can execute this domain. |
| **Excessive Agency & Authorization** | IAM/access-control literacy required | Auditor must be able to read and interpret IAM policies, service-account scopes, and API permission grants — equivalent skill to a standard access-review or privileged-access audit. Does not require a security engineer, but does require comfort with technical access configurations. |
| **Prompt Injection & Adversarial Exposure** | Security engineering / AI red-team specialist required | This domain cannot be fully tested by a generalist auditor. Meaningfully assessing injection defenses requires someone who can read prompt/orchestration architecture and, ideally, design or interpret adversarial test cases. Co-source or use IT security/red-team resources for this domain if not available in-house. |
| **Human Oversight & Escalation** | General IA staff + interview skills | Primarily a process and behavioral-evidence test (review-time sampling, override-rate analysis, interviews). No specialized technical skill required, though familiarity with statistical sampling helps. |
| **Data Lineage, Quality & Privacy** | Data governance / privacy audit experience | Best executed by someone with data-lineage, data-classification, or privacy-audit background. Useful overlap with existing GDPR/data-protection audit skill sets already in most IA functions. |
| **Vendor, Model & Third-Party Risk** | Third-party risk / procurement audit skills | Standard vendor-risk-management audit competency. The AI-specific layer (model versioning, GPAI documentation) is learnable with a short briefing; does not require a technical specialist. |
| **Monitoring, Logging & Incident Response** | Technical log analysis capability | Auditor should be comfortable querying or reviewing raw system/application logs and reconstructing an event timeline — similar skill to IT general controls or SOC log-review work. Specialized AI knowledge is not required if this baseline technical comfort exists. |

## The Honest Resourcing Picture

- **Five of seven domains:** executable with your current team, possibly with a short internal briefing on AI-specific terminology. No new hires required.
- **Prompt Injection & Adversarial Exposure:** genuinely requires either (a) a security engineering background on the audit team, (b) co-sourcing with your information security function, or (c) a specialist third-party engaged for this domain specifically. Do not attempt to self-certify this domain with a generalist team and a checklist alone — the technical depth required to design or evaluate a real injection test exceeds what a control-walkthrough approach can validate.
- **Excessive Agency & Authorization:** not a specialist domain, but it does require an auditor who is comfortable reading IAM/permission configurations directly rather than relying solely on management's narrative description of access controls.

### Build vs. Co-Source Decision

If Prompt Injection testing capability does not exist in-house and the organization has multiple Tier 1 agents, co-sourcing this single domain is usually more defensible — and cheaper — than either skipping it or building a permanent specialist capability for what may currently be a small population of agents. Revisit this decision annually as the Tier 1 population grows.

# 5. Example - Executive Summary

*Designed to be lifted into a committee deck.*

## AI Agent Risk: Audit Coverage & Posture

Internal Audit has established a standards-anchored program to assess risk in autonomous AI agents — systems that take action, not just generate text, with limited human approval. Mapped to ISO/IEC 42001, NIST AI RMF, and the EU AI Act, so findings translate directly into language regulators recognize.

**The risk in plain terms:**

AI agents can take unauthorized or harmful action at machine speed, across many transactions, before a human notices. The same failure — too much autonomy, no injection defense, weak human override — can recur thousands of times before detection.

**Current coverage:**

Tier 1 (Critical) agents identified: **[#]** | Fully tested this cycle: **[#] ([%])**

Open High-severity findings: **[#]** | Domains at “Initial” maturity: **[#] of 7**

**What the Committee should take away:**

- [Biggest exposure this cycle — e.g., gap in human oversight design for Tier 1 agent X]
- [What improved since last reporting cycle]
- [Resourcing or budget ask, if applicable]

*Keep this page self-contained — it should make sense to someone who has not read the underlying audit program.*

# 6. Example - Engagement Planning Memo Template

*To be used by audit manager as starting point for engagement planning.*

## AI Agent Audit — Engagement Planning Memo

| | |
|---|---|
| **Agent/System Name** | [Name of agent or AI-enabled system] |
| **Business Owner** | [Name, title, business unit] |
| **Risk Tier** | [Tier 1 – Critical / Tier 2 – Elevated / Tier 3 – Limited — per Section 2.3 of the audit program] |
| **Tier Rationale** | [1–2 sentences: why this tier — e.g., autonomous transaction authority, data sensitivity, EU AI Act Annex III relevance] |
| **Domains In Scope** | [List which of the 7 domains apply — not all domains apply to every agent] |
| **Domains Excluded** | [List any domains excluded from this engagement, with rationale] |
| **Prior Coverage** | [Date of last test, if any; summary of prior open findings] |
| **Specialist Resource Needed?** | [Yes/No — Prompt Injection domain in scope? Co-sourced or in-house?] |
| **Estimated Hours** | [See planning guide below] |
| **Target Start / Completion** | [Dates] |
| **Engagement Lead** | [Name] |
| **Reporting Destination** | [Audit Committee / AI Governance Committee / Management only] |

## Indicative Hours Planning Guide

Use as a starting estimate only; adjust for system complexity, data access friction, and whether prior-period working papers exist to build from.

| Tier | Domains Tested | Indicative Hours | Notes |
|---|---|---|---|
| Tier 1 – Critical | All 7, full scope | 80–140 hrs | Add 20–40 hrs if co-sourcing Prompt Injection testing externally |
| Tier 2 – Elevated | Sections 5-8 targeted; | 40–70 hrs | Design-only review of remaining domains adds roughly 8–12 hrs |
| Tier 3 – Limited | Sample-test only | 15–25 hrs | Often combinable with adjacent IT general controls or vendor audits already planned |

### Planning Note

These ranges assume reasonably cooperative access to system owners and existing documentation. Add a contingency of 20–30% for first-time engagements on a given agent, where no prior working papers or established management contacts exist.

# 7. Example - Domain Maturity Tracker

Update once per audit cycle. Plot current vs. target maturity to demonstrate progress to the Committee.

| Domain | Current Level | Target Level | Target Date | Trend |
|---|---|---|---|---|
| Governance & Accountability | [1–5] | [1–5] | [Qtr/Yr] | — |
| Excessive Agency & Authorization | [1–5] | [1–5] | [Qtr/Yr] | — |
| Prompt Injection & Adversarial Exposure | [1–5] | [1–5] | [Qtr/Yr] | — |
| Human Oversight & Escalation | [1–5] | [1–5] | [Qtr/Yr] | — |
| Data Lineage, Quality & Privacy | [1–5] | [1–5] | [Qtr/Yr] | — |
| Vendor, Model & Third-Party Risk | [1–5] | [1–5] | [Qtr/Yr] | — |
| Monitoring, Logging & Incident Response | [1–5] | [1–5] | [Qtr/Yr] | — |

### How to use this with the Committee

Lead with the trajectory, not the score. "We are Developing on Prompt Injection, targeting ‘Defined’ by Q3" is a far stronger committee narrative than a static red/amber/green grid — it shows the program is moving and gives the board a checkpoint to hold management to.

Avoid rating every domain the same level by default. Genuine variance across domains is normal and expected — a domain that has had executive sponsorship and dedicated budget (often Governance) should mature faster than one that requires specialized technical testing capability (often Prompt Injection).

# 8. Example - Findings

*Illustrative language for three of the highest-risk control areas.*

Each example follows condition / criteria / cause / effect / recommendation structure deliberately — this is the structure most audit report templates and most Audit Committees expect.

## Example 1 — Excessive Agency

**Finding: Agent Service Account Holds Broader Database Permissions Than Task Requires**

*Domain reference: Section 5 — Excessive Agency & Authorization* *Severity:*

**High**

Condition: The [agent name] service account was found to hold read/write access to 14 database tables, of which independent walkthrough confirmed only 4 are required to perform its documented task (customer support ticket triage and auto-categorization). The account additionally retains a database role with DELETE privileges on two tables outside its task scope. Criteria: Per the organization's AI governance policy and leading practice under ISO/IEC 42001 Annex A.6, AI agents should be provisioned least-privilege, task-scoped access. Cause: Service account provisioning followed a standard application-onboarding template that grants broad schema access by default, rather than a task-scoped access request specific to the agent's function. No subsequent access review was performed after go-live. Effect: In the event of a prompt injection, model error, or compromised orchestration logic, the agent has the technical ability to delete or modify data well beyond its intended task — materially increasing the potential blast radius of any single failure. Recommendation: Re-scope the service account to the minimum permissions required for the documented task; implement a quarterly access recertification specific to AI agent service accounts, consistent with existing privileged-access review processes.

## Example 2 — Human Oversight (Rubber-Stamp Approval)

**Finding: Human Review Step Does Not Provide Meaningful Oversight**

*Domain reference: Section 7 — Human Oversight & Escalation* *Severity:*

**High**

Condition: Review of [#] sampled approval decisions over the audit period showed a median reviewer decision time of [X] seconds and an approval rate of [98%]. Interviews with three reviewers confirmed they are evaluated on approval throughput and do not have access to the underlying source documents the agent used to generate its recommendation at the point of review. Criteria: The EU AI Act (Art. 14) and the organization's own AI governance policy require human oversight to be meaningful — i.e., the reviewer must have sufficient information, time, and authority to identify and act on an incorrect output. Cause: The review workflow UI surfaces only the agent's final recommendation, not its supporting reasoning or source data, and reviewer performance metrics are throughput-based rather than quality-based. Effect: The control as designed creates an appearance of human oversight without the substance of it, exposing the organization to undetected erroneous agent decisions at scale, and to regulatory criticism that oversight is nominal rather than genuine. Recommendation: Redesign the review interface to surface source data and agent reasoning alongside the recommendation; replace throughput-based reviewer metrics with a quality-sampling-based metric; introduce periodic independent re-review of a sample of approved decisions.

## Example 3 — Prompt Injection Exposure

### Finding: No Tested Defense Against Indirect Prompt Injection via Retrieved Documents

*Domain reference: Section 6 — Prompt Injection & Adversarial Exposure* *Severity:*

**High**

Condition: The [agent name] agent retrieves and summarizes externally sourced documents (e.g., vendor-submitted files, inbound emails) as part of its workflow, and is also capable of initiating a downstream action (e.g., drafting and sending a response, updating a record) based on that content. No input sanitization, source-trust tagging, or instruction-hierarchy enforcement was identified between the system prompt and externally retrieved content. No red-team or adversarial test of injection resistance has been performed. Criteria: NIST AI 600-1 (GenAI Profile) and leading practice under the OWASP LLM Top 10 identify indirect prompt injection as a top-tier risk for agents that act on externally sourced content, and recommend explicit defenses including content-source segregation and pre-execution validation of any resulting action. Cause: The agent's architecture concatenates retrieved content directly into the model context without structural separation from system instructions, and no adversarial testing has been incorporated into the agent's pre- or post-deployment assessment process. Effect: A malicious actor able to influence content the agent will read (e.g., a crafted vendor email or document) could potentially cause the agent to take an unauthorized action, exfiltrate information, or bypass intended workflow controls, without needing to breach any traditional network perimeter. Recommendation: Engage security engineering or a qualified third party to perform adversarial/injection testing prior to next deployment cycle; implement structural separation between system instructions and externally retrieved content; require human confirmation for any action triggered by externally sourced, unverified content.