Security and governance checklist

Govern before you approve, know what the system can do.

A practical checklist for reviewing AI agents, models, integrations, deployment boundaries, and governance controls.

> **Axeron builds governance into the workflow architecture before implementation — so the controls the CISO needs exist by design, not by retrofit.**

ASSET PROMISE

elps security teams ask the right questions before approving autonomous or s emi-autonomous AI workflows.

BEST AUDIENCE

CISOs, security architects, risk officers, IT governance teams

RECOMMENDED USE

Use before or after an enterprise briefing to help the buyer align internal stakeholders and define a practical next step.

01 Why the CISO matters early

AI programs fail late when security is treated as an approval step instead of a design constraint. Axeron's position is simple: if the system cannot be controlled, logged, stopped, and explained, it is not ready for high-trust environments.

02 Governance questions

Security teams should evaluate what the AI system can do, not only what the model can say.

- What systems can the agent access?
- What actions can it take?
- Which actions require human approval?
- Can permissions differ by role, workflow, and environment?
- Can self-improvement be gated?
- Can the organization stop or roll back behavior?

03 Deployment questions

The deployment model should match the sensitivity of the data and the operating environment.

- SaaS, private cloud, data center, on-prem, or fully dark
- Data movement and residency
- Identity and access integration
- Network boundary
- Logging location
- Model and tool provider exposure

04 Audit and observability questions

Governance is weak if it only exists in policy documents. The system must generate operational evidence.

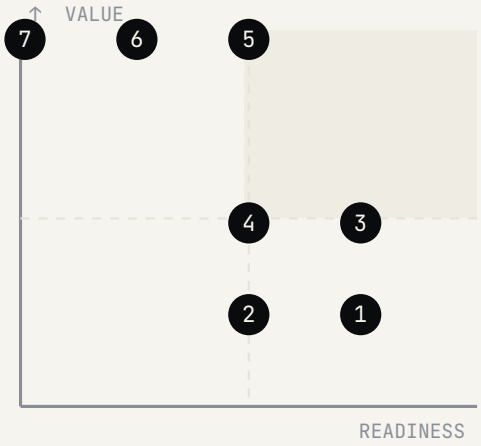
- Action logs
- Source traces
- Approval records
- Version history
- Exception handling
- Risk events
- Change history

05 ow Axeron supports the review

Continuum provides the governance layer for monitored, gated, and auditable agent behavior. AxeStudio helps define the operating controls before implementation so security is part of the workflow architecture from the beginning.

AI-OPPORTUNITY MATRIX

This matrix positions candidate AI workflows on two axes a CISO controls: how sensitive the workflow is to failure (Operational Risk) and how mature the governance controls are for that workflow today (Governance Readiness). Placement is directional — the purpose is to force a structured conversation before any approval decision, not to replace one.



- Audit evidence preparation
- Policy document Q&A
- Case triage and intake routing
- Procurement document review
- KYC and AML workflow support
- Credit and risk review preparation
- Autonomous action or execution

Approve with gates

High risk, high readiness. The workflow carries real operational or regulatory consequences, but the controls — approvals, logs, access boundaries, rollback — are in place. Approve conditionally: define the gate conditions, confirm audit coverage, and schedule a post-launch review.

Do not approve yet

High risk, low readiness. The failure impact is significant and the governance controls are incomplete. Do not deploy. Use the readiness scorecard to identify which dimensions are Weak and resolve them before returning for a security review.

Fast-track approval

Lower risk, high readiness. The workflow is well-governed and the failure impact is contained. Approve with standard logging and a scheduled review. Use this category to build organizational confidence in the governance model.

Remediate first

Lower risk, low readiness. The stakes are modest but the governance gaps are real. Remediate the access boundary, approval design, or audit coverage before deployment — even low-risk agents establish precedents.

WORKFLOW-READINESS SCORECARD

Run this rubric against one candidate AI workflow before the security review closes. Score each dimension as Weak, Workable, or Strong. A mix of Workable and Strong across all eight means the workflow is ready for a conditional approval. A single Weak on Access Boundary or Kill Switch is a blocker — resolve it before any production deployment.

01 **Access boundary**

Is there an explicit, documented list of the systems, data sources, and actions the agent is permitted to reach? Strong: permissions are defined at the role, workflow, and environment level and enforced by the platform — not assumed. Workable: the boundary is described but not yet enforced by technical controls; a design exists. Weak: no one has enumerated what the agent can access or do in production.

02 **uman approval gates**

Are the actions that require human authorization identified and enforced before deployment? Strong: every high-risk action has a named approver, an approval record, and a platform-level gate that cannot be bypassed. Workable: approval points are identified and will be enforced; the implementation is not yet complete. Weak: approval requirements are informal or assumed — no platform gate exists.

03 **Kill switch and rollback**

Can the organization stop or revert agent behavior without a full system shutdown? Strong: a documented stop and rollback path exists, has been tested, and can be executed by the operations team without vendor assistance. Workable: a stop path exists but has not been tested; rollback relies on vendor support. Weak: no documented method exists to halt or reverse agent behavior once it is in production.

04 **Audit and observability coverage**

Does the system generate structured, durable evidence of agent actions, approvals, source references, and exceptions? Strong: action logs, source traces, approval records, and exception events are captured as structured data and retained in a queryable location outside the agent runtime. Workable: some events are logged but coverage is incomplete or logs are held only in the agent runtime. Weak: the only record of what the agent did is the output itself — no structured log, no source trace, no approval record.

05 **Data residency and movement**

Is it clear where data goes when the agent processes it — including any model API calls, tool invocations, or temporary storage? Strong: data movement is documented, residency requirements are met for every processing step, and no data leaves an approved boundary. Workable: residency is met for storage but model or tool calls move data through a provider not yet reviewed. Weak: data movement is not mapped; it is unclear what the model API or tools receive.

06 **Identity and access integration**

Does the agent authenticate through the organization's identity and access management system, and do its permissions reflect the principle of least privilege? Strong: the agent authenticates via the organization's IAM; permissions are scoped to the minimum required for the workflow and differ by environment. Workable: authentication is integrated but permissions are broader than the minimum; scoping work is planned. Weak: the agent operates with service-account or static credentials not connected to the organization's IAM.

07 **Self-improvement gating**

Can the agent propose changes to its own behavior — prompts, routing logic, playbooks — and if so, are those changes reviewed before they reach production? Strong: self-improvement is governed by a policy gate; proposed changes are logged, reviewed, and approved before adoption; high-risk changes require human sign-off. Workable: a gating mechanism is planned or partially implemented; some change paths are ungated. Weak: the agent can modify its own behavior without any review or approval step.

08 **Incident response path**

Is there a documented process for what happens when the agent produces a wrong output, takes an unauthorized action, or exposes data it should not have accessed? Strong: an incident path is documented, ownership is assigned, the security team has been briefed, and the path has been rehearsed at least once. Workable: an incident path exists on paper but has not been rehearsed; ownership is assigned. Weak: no incident path exists; the response would be improvised.

WHY AI PILOTS FAIL

Most AI governance failures follow a small set of recognizable patterns. Each one is visible before deployment — if the review happens early enough and the right questions are asked. The following are the conditions this checklist is designed to surface before they reach production.

WHY AI PILOTS FAIL

- × Do not treat security review as a final approval step after the system is built. When the agent's permissions, data movement, and action scope are already implemented, the cost of making them narrower is high and the pressure to accept the design as-is is significant. Security review belongs at the architecture stage, not the launch stage.
- × Do not accept a verbal description of access controls in place of technical enforcement. An agent that is described as having limited access but whose permissions have not been scoped, tested, or enforced by the platform is an ungated agent. The description is not the control.
- × Do not approve a workflow because the demo output looks safe. A demo uses a constrained scenario. Production introduces volume, exception cases, unexpected inputs, and edge conditions the demo did not encounter. The governance questions are about what the system can do, not what it did in the demo.

- × Do not assume the model provider's terms cover the organization's data residency requirements. Model API calls move data to an external processing environment. Whether that movement complies with regulatory obligations, contractual commitments, or internal policy requires a review — not an assumption.
- × Do not treat a working kill switch as optional for an initial deployment. The ability to stop, limit, or roll back agent behavior is not a recovery feature — it is a basic governance control. A system with no tested stop path has no meaningful upper bound on failure impact.

WHAT TO DO INSTEAD

- → Require a documented, enforced access boundary before the security review closes — not a description of intent, but a tested list of what the agent can reach and what it cannot.
- → Define the human approval gates at the design stage, before implementation begins. The approval architecture is harder to change once it is built around an agent's action model.
- → Confirm that audit logs are structured, durable, and held outside the agent runtime. Logs that live inside the system they are supposed to be auditing are not audit logs.
- → Test the kill switch before approving production deployment. The stop path should be executed by the operations team — not described by the vendor — at least once before the workflow goes live.
- → Use the readiness scorecard as a structured pre-approval conversation, not a post-hoc checklist. Each Weak dimension is a remediation task. Track them, assign owners, and confirm resolution before the workflow reaches production.

WHAT YOU WALK AWAY WITH

After running the CISO checklist against a candidate AI workflow, the security team has a structured set of outputs designed for three internal audiences: the security leadership closing the review, the implementation team resolving the gaps, and the executive sponsor making the deployment decision.

01 Governance readiness assessment completed against the eight scorecard dimensions — each dimension rated Weak, Workable, or Strong, with plain notes on what a Weak or Workable rating means for the deployment timeline and what remediation is required.

02 Access boundary definition reviewed and confirmed — a written record of which systems the agent can reach, which actions it can take, how permissions differ by role and environment, and who is responsible for maintaining that boundary as the workflow evolves.

03 Human approval gate design reviewed — a documented list of the actions that require human authorization, the named approver for each, the platform-level enforcement mechanism, and the audit event each approval generates.

04 Data movement and residency assessment — a documented trace of where data goes during agent processing, including model API calls and tool invocations, with a confirmation that each processing step meets the organization's residency and contractual requirements.

05 Audit and observability coverage confirmed — a record of which events are logged, where logs are held, how long they are retained, and whether the log set is sufficient to reconstruct agent behavior for an internal or external audit.

06 Incident response path documented and reviewed — a written process for what happens when the agent produces a wrong output, takes an unexpected action, or exposes data it should not have accessed, with ownership assigned and a rehearsal scheduled.

07 Deployment decision record — a plain written record of the security team's finding: approved with named conditions, deferred pending remediation of named dimensions, or rejected with the reasons stated — so the decision is traceable and can be revisited as the system evolves.

WHERE IT FEEDS

Continuum provides the governance layer that makes this checklist operational rather than documentary. Access boundaries, approval gates, audit events, kill switch controls, and self-improvement gates are configured in Continuum before the workflow goes live — so the controls the security team reviews exist as enforced platform behavior, not as policy statements. AxeStudio works with the implementation team during the design phase to embed governance requirements into the workflow architecture from the start. The security review is not a separate track that runs alongside the build — it shapes the build. That means when the CISO closes the review and issues a conditional approval, the named conditions are already reflected in the platform configuration, not queued as post-launch remediation items. The same Axeron team that maps the process, builds the agent, and configures the governance layer operates the system in production. There is no handoff to a separate managed-services vendor who was not in the security review. That continuity matters: the people who understand why a particular gate was designed the way it was are the people who will maintain it when the workflow evolves.

BEFORE YOU START

- [ ] Access boundary defined
- [ ] Audit fields defined
- [ ] Data residency confirmed
- [ ] Incident path documented
- [ ] Human approvals defined
- [ ] Kill switch identified
- [ ] Model/tool exposure reviewed
- [ ] Self-improvement gates configured

NEXT STEP

Run the CISO checklist before selecting the first AI workflow for production.

Axeron turns resource-level education into a scoped conversation: one process, one measurable outcome, one deployment model, and one governance path.