AI SECURITY ENGINEERING FIELD GUIDE · 2026

Chapter 14 · Secure AI Architecture Design

Standalone learning module for LMS delivery and required reading.

FORMAT	USE	SCOPE	AUDIENCE
Standalone PDF	LMS reading module	Single chapter	Learners

CHAPTER 14

Secure AI Architecture Design

SECURE AI ARCHITECTURE

Put enforcement where the model can be wrong and the system still stays safe.

> Put enforcement where the model can be wrong and the system still stays safe.
>
> FIELD GUIDE

> **FIELD GUIDE: ARCHITECTURE REVIEW**
>
> Inspect trust placement, data-plane controls, model-provider boundaries, retrieval paths, tool authority, fallback routes, telemetry, and release gates.

This domain covers secure architecture for LLM applications, RAG systems, agents, workflows, model chains, provider boundaries, tool integrations, telemetry, and governance evidence. It matters when one design decision can determine whether prompt injection, retrieval leakage, excessive agency, privacy exposure, or missing evidence becomes a systemic failure.

Architecture review is where local fixes become a system design. Prompts, filters, evals, retrieval controls, tool policy, provider boundaries, approvals, and telemetry should not all fail for the same reason or rely on the same model judgment.

> **FIELD USE**
>
> Draw the architecture with three overlays: data flow, authority flow, and evidence flow. A design is not review-ready until all three are visible.

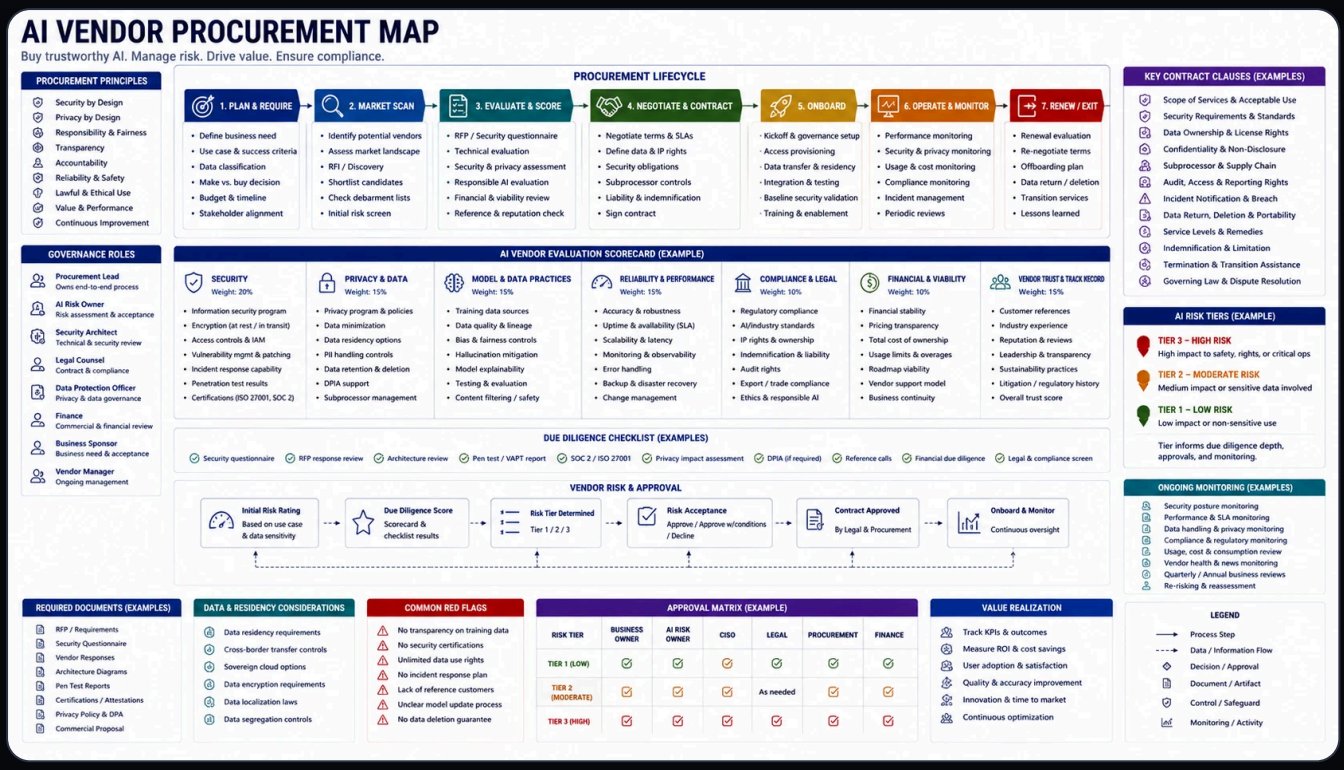

FIGURE 1: FIGURE 15: SECURE AI ARCHITECTURE REFERENCE SHOWING DATA FLOW, AUTHORITY FLOW, AND EVIDENCE FLOW OVERLAYS ACROSS PROMPT, RETRIEVAL, TOOL, PROVIDER, AND LOGGING LAYERS

WHAT BREAKS HERE?

- The model becomes the sole enforcement point for authorization, policy, or safety.
- Retrieved content, tool output, and memory share one trust level.
- Fallback paths bypass security controls.
- Multi-model chains lose provenance and accountability.
- Architecture diagrams omit logs, approvals, exception paths, and control owners.
- Defense layers all depend on the same compromised context or model judgment.

WHAT TO INSPECT

- System architecture, data flows, trust boundaries, and authority boundaries.
- Prompt, retrieval, model, tool, memory, provider, and logging layers.
- Fallback routes, cached outputs, disabled-control modes, and emergency workflows.
- Approval gates, policy engines, rate limits, sandboxes, and kill switches.
- Evidence paths for control decisions, tests, releases, and incidents.
- Multi-model chains, handoff points, cached context, and external integration paths.

WHAT TO ASK

- Which controls enforce policy outside the model?
- What untrusted content can influence high-privilege action?
- Which boundaries protect confidentiality before context assembly?
- What happens when the model, provider, retrieval system, or tool fails?
- Which fallback paths preserve authorization, logging, and approvals?
- Can the architecture produce evidence for customers, auditors, and incidents?
- Which layers fail independently, and which share the same failure mode?

WHAT TO TEST

- Model-enforced authorization failure.
- Prompt injection across retrieval, tools, memory, and multi-model chains.
- Fallback bypass of logging, authorization, rate limits, or approvals.
- Agent blast-radius scenarios.
- Cross-tenant, cross-role, and cross-provider data flows.
- Evidence reconstruction from architecture trace points.
- Control failure combinations across prompt, retrieval, tool, provider, and approval layers.

CONTROLS AND GUARDRAILS

- Explicit trust model for users, prompts, context, retrieval, tools, memory, providers, and logs.
- Deterministic controls for authorization, data access, tool policy, approvals, and output handling.
- Defense layers with independent failure modes.
- Fallback security requirements and control-preserving degradation paths.
- Architecture decision records for provider, retrieval, agent, data, and telemetry choices.
- Release gates tied to architecture risks and evidence production.
- Independent layers for data access, action authorization, output handling, telemetry, and approvals.

EVIDENCE TO COLLECT

- Secure AI architecture diagram.
- Trust-boundary map.
- Data-flow and authority-flow records.
- Architecture decision records.
- Fallback control matrix.
- Approval and telemetry evidence.
- Architecture review findings.
- Data-flow, authority-flow, and evidence-flow overlays.

OUTPUT ARTIFACTS

- Secure AI architecture review.
- Threat/trust map.
- Authority-flow diagram.
- Fallback control matrix.
- Architecture remediation backlog.

RELATED SERVICES AND WORKBENCH TOOLS

TYPE	RELATED PATHS
Workbench	Threat Canvas, Authority Graph, Runtime Proxy, AI Control Crosswalk
Services	AI Product Security Assessment, Product Security Baseline, Agentic Workflow Hardening
Handbook	Architecture and Trust Boundaries, Threat Modeling, Governance Evidence and Customer Trust

DECISION - CONDITIONAL: ARCHITECTURE READINESS DECISION

Approve the design only when enforcement, evidence, fallback behavior, and ownership remain clear even if model behavior is unreliable.