connvertex

REFERENCE ARCHITECTURE · AI + SECURITY · v1.0

The Secure AI Application Reference

A CISO's reference for approving, shipping, and running AI applications. Threat models, control boundaries, and compliance alignment for single-LLM, RAG, and agentic systems.

Single LLM | RAG | Agentic | OWASP LLM Top 10 | NIST AI RMF

Prepared by Connvertex · Practitioner-led cybersecurity and AI engineering.

Released April 2026 · connvertex.com

SECTION 01

Document control

Scope, references, versioning, and the terms of use for this document.

Purpose

This reference describes the architecture, threat models, and control boundaries we use on Connvertex engagements to build and ship AI applications into regulated environments. It is written for a security leader evaluating, approving, or operating an AI application. It is not a product specification, a model benchmark, or a vendor comparison.

Framework references

Framework	Version referenced
OWASP LLM Top 10	2025 release
MITRE ATLAS	v4.x
NIST AI RMF	AI 100-1 (Jan 2023) + AI 600-1 Generative AI Profile
NIST SP 800-53	Rev 5
HIPAA Security Rule	45 CFR 164.300
SOC 2	AICPA TSC 2017 with 2022 points of focus
FedRAMP	Rev 5 baselines
EO 14110 / OMB M-24-10	Federal high-impact AI use minimum practices

SECTION 02

Contents

SECTION 03

Executive summary

The shortest read-out a security leader needs before approving an AI application.

AI security is not a research problem. It is a control-boundary problem.

The question we are most often asked is whether an AI application can be shipped safely into a regulated environment. The answer is yes, and the path there is less exotic than the vendor marketing suggests. AI applications are ordinary applications with three additional surfaces a CISO needs to bound: a probabilistic model, a retrieval or tool layer, and a prompt-driven control plane.

This reference describes the seven-layer architecture we deploy on Connvertex engagements. Every AI application we have shipped into FedRAMP-aligned, HIPAA-covered, or SOC 2 Type II-audited environments has used the same seven layers. The patterns differ. The layers do not.

Sections three through five walk through the layers and then apply them to the three AI patterns a security leader sees in practice: a single-turn LLM application, a retrieval-augmented application, and an agentic application with tools. Sections six through eight map the architecture to the OWASP LLM Top 10, the NIST AI Risk Management Framework, and the control catalog you already manage (NIST SP 800-53, the HIPAA Security Rule, SOC 2 Trust Services Criteria, FedRAMP baselines). Section nine is the decision framework for when to build, when to buy, and when to hold.

The five things this document argues

- An AI application is an ordinary application with three extra surfaces: a probabilistic model, a retrieval or tool layer, and a prompt control plane.
- Seven layers bound those surfaces: trust and tenant, input, retrieval and context, model, orchestration and tool, output, observability and governance.
- Three patterns (single LLM, RAG, agentic) reuse the same seven layers with different threat emphases. Agentic systems carry the highest control cost.
- Existing control catalogs (NIST 800-53, HIPAA, SOC 2, FedRAMP) cover roughly seventy percent of the surface. The remaining thirty percent is AI-specific and addressed by the OWASP LLM Top 10 and the NIST AI RMF.
- If your program does not have a red team for AI, a structured eval harness, and an incident playbook that includes a model rollback, you are not yet ready to ship a high-impact AI application.

WHO THIS IS FOR

A security leader (CISO, vCISO, security architect, or compliance officer) who needs to evaluate, approve, or operate an AI application in a regulated environment. If you are a builder looking for implementation details, the companion interactive explorer at connvertex.com/tools/ai-architecture-explorer is the better starting point.

SECTION 04

The seven-layer architecture

Seven layers bound every AI application we deploy. The patterns in Section 5 reuse these layers with different emphasis.

Architecture overview

LAYER 01	Trust and tenant
LAYER 02	Input
LAYER 03	Retrieval and context
LAYER 04	Model
LAYER 05	Orchestration and tool
LAYER 06	Output
LAYER 07	Observability and governance

Layers above. Each carries a defined purpose, a known set of components, a catalog of risks we watch for, and a minimum control set we deploy. The rest of this section is the detailed card for each layer.

LAYER 01 — Trust and tenant layer

Purpose. Establishes who is calling, what they are allowed to see, and where their data lives.

COMPONENTS

- Authenticated caller identity, carried end-to-end as a signed token.
- Per-tenant and per-caller data partition (row-level, namespace, or dedicated index).
- Classification and clearance metadata attached to every request before a model is invoked.

COMMON RISKS

- Cross-tenant data exposure through a shared retrieval index.
- Silent privilege expansion when a caller identity is lost in the middle of an orchestration.
- Unsigned caller context that a downstream component chooses to trust.

CONTROLS

- Signed request context (JWT or equivalent) validated at every layer.
- Per-tenant index isolation; cross-tenant reads are impossible by construction, not policy.
- Explicit classification labels on every payload (PII, ePHI, CUI, FCI, Public).

LAYER 02 — Input layer

Purpose. Filters, classifies, and decorates the user input before anything model-facing sees it.

COMPONENTS

- PII and secrets detector that redacts sensitive fields at the perimeter.
- Prompt-injection classifier that scores an input for adversarial intent before routing.
- Context classifier (topic, risk tier) that determines which downstream model and toolset is allowed.

COMMON RISKS

- Direct prompt injection overriding the system prompt.
- Indirect prompt injection arriving through a retrieved document.
- PII or secrets copied into a vendor-hosted model's training telemetry.

CONTROLS

- Deterministic redaction before the prompt crosses a trust boundary.
- Input classifier run server-side; never expose the score to the caller.
- Hard ceiling on input length; reject anything past the limit rather than truncate silently.

LAYER 03

Retrieval and context layer

Purpose. Fetches the documents and facts the model will see, scoped to the caller's authorization.

COMPONENTS

- Vector index with per-tenant and per-document ACLs enforced at query time.
- Document provenance metadata preserved through the retrieval pipeline.
- Time-bound cache with invalidation tied to source-document revocation.

COMMON RISKS

- Retrieval returns a document the caller is not cleared to see.
- Poisoned document injects instructions the model treats as authoritative.
- Stale cache serves content the source system has already revoked.

CONTROLS

- ACL filter evaluated inside the retrieval query, not post-filter.
- Provenance-aware prompt assembly that tags every retrieved span with its source.
- Revocation hooks from the authoritative source systems into the vector index.

LAYER 04

Model layer

Purpose. Runs the inference. The only layer where the probabilistic nature of AI is unavoidable.

COMPONENTS

- Model choice: hosted frontier model, hosted open-weights model, or self-hosted.
- Data-use terms (zero-retention, no-training) encoded in the vendor contract.
- Version pinning and rollback capability tied to the incident response plan.

COMMON RISKS

- Vendor retains prompts or completions for training, exposing regulated data.
- Silent model version change alters behavior without notice.
- Self-hosted model running on infrastructure without FIPS-validated crypto modules.

CONTROLS

- Contractual zero-retention, or self-hosting, whenever regulated data is in play.
- Pinned version with a documented upgrade cadence; rollback path tested at least annually.
- FIPS-validated crypto for model weights at rest and in transit; attested runtime where the compliance context requires it.

LAYER 05

Orchestration and tool layer

Purpose. Routes the model's output to internal or external systems when the application calls for tool use.

COMPONENTS

- Tool allowlist with per-tool authorization rules tied to the caller's identity.
- Typed argument schemas validated before any tool call is dispatched.
- Human-in-the-loop checkpoints for high-impact actions (money movement, destructive writes, cross-tenant reads).

COMMON RISKS

- Model-initiated tool calls execute with more authority than the caller holds.
- Argument smuggling: a tool called with values the caller could not set directly.
- Runaway loops where a model calls tools repeatedly and exhausts budget or causes downstream load.

CONTROLS

- Tool authorization re-evaluated at call time against the caller's identity, not the model's context.
- Argument schema validation with a rejection on any unknown or coerced value.
- Per-request step budget and per-caller rate limit on tool invocations.

LAYER 06

Output layer

Purpose. Filters, validates, and attributes the model's response before a human or downstream system sees it.

COMPONENTS

- Schema validation on structured outputs; refusal on malformed responses.
- PII and secret redaction on the outbound path.
- Citation and provenance formatting tied to the retrieval layer.

COMMON RISKS

- Model exfiltrates training data or system prompt in an output.
- Hallucinated content delivered as authoritative in a regulated context.
- Unsanitized output rendered as HTML or Markdown, enabling XSS or prompt-injection in chained apps.

CONTROLS

- Output validator that rejects responses that fail schema or contain forbidden tokens.
- Provenance-required responses: if retrieval was used, citations are required or the response is rejected.
- Content-safety classifier on the outbound path; log scores, not the raw content.

LAYER 07

Observability and governance layer

Purpose. The audit trail, the evaluation harness, and the red-team cadence. This layer is how the program stays trustworthy.

COMPONENTS

- Structured logging of every request, retrieval, tool call, and response, with PII redacted.
- Offline evaluation harness with ground-truth test set and safety-specific tests.
- Online monitoring for drift, refusal rate, and adversarial input rate.
- Documented red-team cadence with findings tracked to closure in the POA&M.;
- Incident response plan that includes model rollback and retrieval-index rollback paths.

COMMON RISKS

- Logs contain regulated data because the redaction is at the wrong layer.
- Evaluation drift: the model gets worse but no one notices because the eval is a year old.
- An incident is declared but the team has no playbook for which component to roll back first.

CONTROLS

- Redaction before logging, not after. Logs never contain raw prompts or completions.
- Eval harness run on every material change; results entered into the release checklist.
- Quarterly red team with at least one novel prompt-injection scenario and one tool-abuse scenario.
- IR plan includes a one-page 'AI incident' addendum with model rollback, retrieval rollback, and public comms guidance.

SECTION 05.1

Pattern — Single-turn LLM application

A stateless application that sends a user prompt to a model and returns the completion. No retrieval, no tools. The lowest-risk pattern and the easiest to approve.

Typical use cases

- Drafting assistance and summarization of user-provided text.
- Classification, extraction, and translation inside a scoped workflow.
- Citizen-facing or employee-facing productivity features where the model sees only what the user provides.

Layer-by-layer emphasis

Layer	Emphasis in this pattern
Trust and tenant layer	Table stakes. Caller identity and tenant context carried through.
Input layer	Primary surface. Prompt-injection defense lives here. PII redaction matters most for vendor-hosted models.
Retrieval and context layer	Not present. Skip.
Model layer	Vendor selection, data-use terms, version pinning. FIPS for regulated workloads.
Orchestration and tool layer	Not present. Skip.
Output layer	Schema validation, PII redaction on outbound path, safety classifier.
Observability and governance	Full logging, eval harness, red team cadence.

Top risks (mapped to OWASP LLM Top 10)

Risk	How it shows up in this pattern
LLM01 Prompt Injection	Direct injection from the user input.
LLM02 Sensitive Information Disclosure	Vendor retains or logs regulated data.

LLM05 Improper Output Handling	Output rendered unsafely in a downstream surface.
LLM09 Misinformation	Hallucinated content delivered as authoritative.
LLM10 Unbounded Consumption	Cost / availability attack through long inputs or adversarial outputs.

Minimum control set

- Server-side prompt-injection classifier on every input.
- Zero-retention or self-hosted model for any regulated workload.
- Structured-output schema validation on the response path.
- Per-caller cost and rate limit enforced before the model call.
- Structured logging with PII redaction; no raw prompts or completions in logs.

SECTION 05.2

Pattern — Retrieval-augmented application (RAG)

An application that retrieves documents or facts from an indexed corpus and includes them in the prompt. Adds one major surface (the retrieval layer) and a meaningful category of indirect prompt-injection risk.

Typical use cases

- Policy and SOP assistants that answer from authoritative internal documentation.
- Case-lookup or record-lookup tools inside a bounded dataset.
- Knowledge-base question answering for citizen services or internal help.

Layer-by-layer emphasis

Layer	Emphasis in this pattern
Trust and tenant layer	Critical. Retrieval must enforce ACLs at query time, not post-filter.
Input layer	Same as single-turn, plus rewriting of retrieval queries if the user input shapes them.
Retrieval and context layer	Primary surface. Per-tenant index isolation, provenance-preserving retrieval, time-bound caching.
Model layer	Same selection criteria as single-turn. Additional context-window budget for retrieved spans.
Orchestration and tool layer	Not present in classic RAG. Appears in agentic RAG.
Output layer	Citation enforcement. Responses without a valid citation are rejected.
Observability and governance	Additional: retrieval-quality metrics, stale-cache rate, cross-tenant access alerting.

Top risks (mapped to OWASP LLM Top 10)

Risk	How it shows up in this pattern
LLM01 Prompt Injection (indirect)	A poisoned document instructs the model through the retrieved context.

LLM02 Sensitive Information Disclosure	Retrieval returns documents the caller is not cleared to read.
LLM04 Data and Model Poisoning	An indexed document has been tampered with upstream.
LLM08 Vector and Embedding Weaknesses	Adversary inverts embeddings or floods the index with near-duplicates.
LLM09 Misinformation	Stale retrieval serves content that has since been revoked.

Minimum control set

- Per-tenant and per-document ACLs enforced inside the retrieval query.
- Provenance-aware prompt assembly; every retrieved span is tagged with its source.
- Citation required in the response; responses without citations are rejected.
- Indirect-injection detector on retrieved content before it reaches the model.
- Index revocation hook from the source systems with tested round-trip.

SECTION 05.3

Pattern — Agentic application with tools

An application that lets the model plan, call tools, and act on external systems. The highest-capability and highest-risk pattern. Most of the control work lives in the orchestration, tool, and governance layers.

Typical use cases

- Operations copilots that open tickets, run scripts, and update records on behalf of a human operator.
- Research or analyst agents that gather evidence across sources before producing a read-out.
- Back-office automation that executes known-shape workflows end-to-end under human sign-off.

Layer-by-layer emphasis

Layer	Emphasis in this pattern
Trust and tenant layer	Critical. Tool calls re-authorize against the caller's identity on every dispatch.
Input layer	Same defenses as single-turn. Higher weight on goal-shape classification and intent validation.
Retrieval and context layer	Same as RAG when retrieval is in play. Tool-call outputs treated as untrusted context.
Model layer	Same selection criteria. Version pinning is non-negotiable; agentic behavior drifts with model updates.
Orchestration and tool layer	Primary surface. Tool allowlist, argument validation, HITL checkpoints, step budgets.
Output layer	Side effects checked before they are committed. Dry-run mode for high-impact actions.
Observability and governance	Highest investment. Step-by-step tracing, agent eval harness, scenario-based red team.

Top risks (mapped to OWASP LLM Top 10)

Risk	How it shows up in this pattern

LLM06 Excessive Agency	Model initiates tool calls beyond the caller's authority or the task's intent.
LLM01 Prompt Injection	A tool output or retrieved document takes over the agent's plan.
LLM05 Improper Output Handling	A tool argument is rendered unsafely by the downstream system.
LLM10 Unbounded Consumption	Agent loops and exhausts budget, quota, or downstream system load.
LLM03 Supply Chain	Third-party tool or plugin is compromised and exfiltrates data the agent can reach.

Minimum control set

- Per-tool authorization evaluated against the caller's identity at dispatch time.
- Typed argument schemas; reject any argument the caller could not set by hand.
- Human-in-the-loop checkpoint on any destructive, financial, or cross-tenant action.
- Per-request step budget and per-caller tool-invocation rate limit.
- Full step trace in the audit log with PII redaction before persistence.
- Scenario-based red team that includes at least one tool-abuse and one runaway-loop scenario per quarter.

SECTION 06

OWASP LLM Top 10 reference

The industry-standard risk catalog, mapped to the layer that primarily addresses each item.

ID	Risk	What it is	Primary control
LLM01	**Prompt Injection**	Direct or indirect adversarial input overriding the system prompt.	Input classifier, system-prompt segregation, provenance-tagged retrieval.
LLM02	**Sensitive Information Disclosure**	Model or system leaks regulated data in outputs, logs, or vendor telemetry.	Redaction at the perimeter, zero-retention vendor contracts, output filters.
LLM03	**Supply Chain**	Compromised model weights, datasets, plugins, or MCP servers.	Pinned model versions, signed weights, vendor SBOM, tool allowlist.
LLM04	**Data and Model Poisoning**	Training or retrieval data tampered with to induce adversary-chosen behavior.	Provenance controls, retrieval-time signing, anomaly detection on embeddings.
LLM05	**Improper Output Handling**	Downstream surface renders model output without escaping or schema validation.	Output schema validation, content-type safety, SSRF/XSS defenses downstream.
LLM06	**Excessive Agency**	Model is granted more capability than the task or caller warrants.	Least-privilege tool design, per-call re-authorization, HITL checkpoints.
LLM07	**System Prompt Leakage**	Operational secrets placed in the system prompt and later exfiltrated.	No secrets in prompts, role-split architecture, output filter on policy content.
LLM08	**Vector and Embedding Weaknesses**	Embedding inversion, near-duplicate flooding, or cross-tenant leakage in the index.	Per-tenant index isolation, query-time ACL, embedding-layer monitoring.
LLM09	**Misinformation**	Hallucinated or outdated content delivered as authoritative.	Citation requirement, eval harness, stale-cache detection.
LLM10	**Unbounded Consumption**	Cost, quota, or availability exhaustion via adversarial use.	Per-caller rate limits, input length caps, step budgets, circuit breakers.

SECTION 07

Compliance alignment

Where the AI-specific controls sit inside the control catalogs you already run.

Existing control catalogs cover roughly seventy percent of the AI control surface. The remaining thirty percent is addressed by AI-specific frameworks (the OWASP LLM Top 10 and the NIST AI RMF) and by the operational practices in Section 8. The table below maps common controls to their AI-specific interpretation.

Framework	Control reference	AI-specific interpretation
NIST SP 800-53 Rev 5	AC-3 / AC-6	Model and tool calls respect least-privilege at the caller's identity. Tool authorization is re-evaluated at dispatch; the model's in-context identity is not sufficient.
NIST SP 800-53 Rev 5	AU-2 / AU-6	Every model request, retrieval, and tool call produces a structured audit event. Logs are redacted before persistence.
NIST SP 800-53 Rev 5	SC-8 / SC-13 / SC-28	FIPS-validated crypto on the model path (weights at rest, prompts in transit, embeddings at rest) for any regulated workload.
NIST SP 800-53 Rev 5	SI-4	Monitoring covers drift, refusal rate, and adversarial input rate in addition to standard service health.
NIST SP 800-53 Rev 5	SA-9 / SR-3	Model and tooling vendors carry contractual zero-retention or self-hosting. Third-party tools are inventoried with SBOM.
HIPAA Security Rule	§164.308(a)(1)(ii)(A)	Risk analysis explicitly covers the AI components. Prompt injection and vendor retention are assessed as risks, not assumed as accepted.
HIPAA Security Rule	§164.312(a)(1)	Per-user access controls flow through the AI path. A vCISO may not delegate authorization to the model's context.
HIPAA Security Rule	§164.312(e)(1)	Transmission security between application, retrieval, and model layers is TLS 1.2+ with FIPS-validated modules where required.
SOC 2 TSC	CC6.1 / CC6.6	Per-tenant isolation is enforced by the retrieval layer, not by the model. Cross-tenant reads are impossible by construction.

SOC 2 TSC	CC7.1 / CC7.2	AI-specific monitoring (drift, refusal, adversarial input) is instrumented, reviewed on a cadence, and tied to incident response.
FedRAMP Moderate	Baseline + NIST AI RMF	FedRAMP Moderate controls apply plus NIST AI RMF Measure and Manage functions. High-impact AI uses require the additional controls from OMB M-24-10.
NIST AI RMF (AI 100-1)	Govern 1 · Map 2 · Measure 2 · Manage 1	Governance documented in the AI impact assessment. Measurement uses the evaluation harness. Management tied to the incident response and rollback plan.
OMB M-24-10 / EO 14110	High-impact AI use minimum practices	AI impact assessment, pre-deployment testing, ongoing monitoring, public transparency where applicable. Applies to federal agencies and federal-adjacent deployments.

SECTION 08

Evaluation and observability

How the program stays trustworthy after the first release.

Offline evaluation
Runs on every material change (model version, prompt, retrieval index). Ground-truth test set plus a safety-specific test set (prompt injection, jailbreak, tool-abuse, PII leakage). Pass/fail thresholds published.

Online monitoring
Continuous metrics: refusal rate, adversarial-input-score distribution, retrieval stale-cache rate, tool-call rejection rate, output-validation rejection rate. Alerts wired into the SOC runbook.

Red team cadence
Quarterly, at minimum. At least one novel prompt-injection scenario and one tool-abuse scenario per round. Findings enter the shared POA&M.;

Incident response addendum
One-page addendum to the existing IR plan. Covers model rollback, retrieval-index rollback, tool disablement, and public-communications guidance where the incident touches users.

SECTION 09

Decision framework

When to build. When to buy. When to hold.

BUILD

In-house or contractor-built application with full control over the seven layers. Choose this when the application handles regulated data, when the use is high-impact, or when differentiation matters.

BUY

Commercial off-the-shelf AI feature inside an existing SaaS. Choose this when the feature is undifferentiated, the vendor carries a current SOC 2 Type II and a FedRAMP or HIPAA authorization where required, and the vendor contract encodes zero-retention for your data.

HOLD

Do not ship yet. Choose this when (a) your organization does not have an AI red team or a structured eval harness, (b) the use case is high-impact but the minimum controls in this document are not in place, or (c) the vendor cannot provide zero-retention on regulated data. Revisit in a quarter.

SECTION 10

Next step

About Connvertex.

WANT THIS REFERENCE APPLIED TO YOUR AI APPLICATION?

Book a 30-minute architecture review.

Book a 30-minute architecture review with a senior Connvertex practitioner. We will map your current or planned AI application to the seven-layer model, flag the controls most likely to fail an audit, and send you a written read-out within 48 hours. No proposal, no qualification call, no BDR.

connvertex.com/contact · hello@connvertex.com

About Connvertex

Connvertex is a practitioner-led cybersecurity and digital services firm serving U.S. federal, state, and local government and the regulated enterprises that support them. Our AI engineering practice builds and operates AI applications against this reference architecture. Our security practice reviews client AI applications against the same standard.

Minority-owned. Woman-owned. NMSDC MBE certified. Pursuing SBA 8(a), WOSB / EDWOSB, and CMMC 2.0 Level 2 certification in 2026.

Connvertex Secure AI Application Reference Architecture, version 1.0. Released April 2026. Informational; not a substitute for assessor-grade advice. Consult the primary-source publications for authoritative interpretation.