AlgoSec SaaS AI Security

OWASP TOP 10 FOR LLM APPLICATIONS COMMON CONTROL

Document status	Published v1
Framework	OWASP Top 10 for LLM Applications 2025 https://genai.owasp.org/llm-top-10/
Date	May 2026

Common SaaS AI control baseline

Baseline statement

AlgoSec AI-enabled SaaS capabilities use managed foundation models through AWS Bedrock. For these SaaS AI workflows, the underlying models are not trained or fine-tuned using customer data; AWS terms and conditions guarantee that model providers do not have access to Bedrock logs, customer prompts, or completions. LLM interactions are mediated by AlgoSec application services, customer context is scoped to the relevant tenant and user permissions, and outputs are handled within controlled application workflows with authorization, validation, logging, and monitoring controls.

OWASP LLM risk table - common control statements

OWASP LLM risk	Common customer-facing control statement
LLM01:2025 **Prompt Injection** Risk focus: Malicious or unintended prompts alter model behavior, bypass guardrails, or influence downstream actions.	LLM interactions are mediated by AlgoSec application services rather than exposing raw model interfaces. Inputs are constrained to define product workflows, sanitized or validated where applicable, and paired with backend authorization checks, so prompt content cannot directly override access controls or execute unapproved operations. In addition, AlgoSec uses AWS Bedrock built-in guardrails coupled with proprietary AlgoSec specific controls to mitigate prompt injections and irrelevant or undesirable prompts.
LLM02:2025 **Sensitive Information Disclosure** Risk focus: Sensitive, proprietary, or tenant-specific data is unintentionally exposed through prompts, context, logs, or model output.	AlgoSec AI-enabled SaaS features use pre-trained managed foundation models and do not train or fine-tune the underlying models on customer data. AlgoSec uses AWS Bedrock for access to foundational models. Data is processed as transient prompts, returned to AlgoSec, post-processed, and discarded. AWS bedrock terms and conditions guarantee that prompts and responses are not used to train AI models.
LLM03:2025 **Supply Chain** Risk focus: Third-party models, libraries, model repositories, or provider changes introduce compromise, dependency, or integrity risk.	AlgoSec uses managed foundation models through AWS Bedrock and does not self-host, modify, or import models from untrusted public repositories for these SaaS AI workflows. Provider and platform changes are handled through vendor/security review, model-version tracking, and monitoring relevant advisories.

OWASP LLM risk	Common customer-facing control statement
LLM04:2025 **Data and Model Poisoning** Risk focus: Training data, fine-tuning data, embeddings, or runtime context are manipulated to bias, corrupt, or backdoor model behavior.	AlgoSec does not pre-train or fine-tune the underlying foundation models on customer data for these SaaS AI features. Protection against data and model poisoning are the responsibility of the Foundational model providers.
LLM05:2025 **Improper Output Handling** Risk focus: Model output is treated as trusted code or data and is passed downstream without adequate validation, encoding, or control.	LLM output is treated as untrusted application data. Responses are constrained to structured formats where appropriate, validated or transformed before downstream use, not directly injected into executable client-side or backend contexts, and monitored through application logging and operational controls.
LLM06:2025 **Excessive Agency** Risk focus: The LLM has excessive autonomy, permissions, or tool access, enabling unintended or harmful actions.	The LLM is not granted unrestricted autonomy or open-ended tool access. The system is designed to validate LLM outputs through a controlled human-in-the-loop process before they are incorporated into production workflows or made visible to broader users. High impact APIs are optionally blocked through user-controlled whitelisting.
LLM07:2025 **System Prompt Leakage** Risk focus: System prompts reveal secrets, operational logic, security controls, or other information that should not be exposed.	System prompts are not used as security boundaries and do not contain secrets, credentials, connection strings, or authorization logic. Authentication, authorization, session management, and tenant separation are enforced outside the LLM by the SaaS application and supporting services.
LLM08:2025 **Vector and Embedding Weaknesses** Risk focus: Retrieval, embedding, or vector-store weaknesses allow context manipulation, data leakage, or unauthorized retrieval.	Where retrieval or embedding mechanisms are used, data sources are controlled, tenant-scoped, permission-aware, and not publicly exposed. Retrieval operations are logged and auditable, and the system avoids user-controlled access to vector stores or embedding pipelines that could manipulate context or bypass access boundaries. All contextual data passed to the model (e.g., via embeddings or Retrieval-Augmented Generation) originates from the customer data and is strictly retained within the customer's isolated tenant to prevent contamination or injection of malicious data. No third-party or public data sources are used in the RAG process, further reducing exposure to potentially poisoned or manipulated content.
LLM09:2025 **Misinformation** Risk focus: The model produces false or misleading output that appears credible and may be over-trusted by users.	LLM-generated content is positioned as assistive, not authoritative. The system is designed to validate LLM outputs through a controlled human-in-the-loop process before they are incorporated into production workflows or made visible to broader users.
LLM10:2025 **Unbounded Consumption** Risk focus: Excessive or uncontrolled inference causes denial of service, runaway cost, degraded service, or unauthorized model use.	LLM usage is bound by product workflows, authenticated access, request or interaction limits, context-size constraints, and controlled data sources. Monitoring and operational limits are used to reduce denial-of-service, uncontrolled cost, and unauthorized inference risks.