

SERVICE DATASHEET

AI / LLM Penetration Testing

Manual testing of your AI and LLM features: who can write the prompt configuration, reach the pipeline's credentials and drive the model lifecycle; whether one tenant's user can read another's conversations and documents; what an agent is authorised to do on a user's behalf; and where prompt injection actually leads.

BEST FOR	Products with embedded generative features; AI agents with tool access; vendor-enabled AI inside enterprise platforms
YOU RECEIVE	Prioritised AI findings, from admin-plane authorisation to injection paths
COVERS	LLM apps, agents, AI admin plane

Start with the control plane, not the chatbot

Most AI security testing is a conversation with a model. But an AI feature bolted into a business application is a small distributed system: something stores the prompt configuration, something creates and trains models, something triggers the ingestion jobs, and something holds the credentials the pipeline uses to reach object storage and your business data. All of it is configuration, all of it is exposed over HTTP, and all of it is a far shorter path to impact than persuading a model to misbehave. We test that surface first. (It's the argument behind our write-up on the AI admin plane.)

We work at the request level as an ordinary tenant user, not through the admin console — because we have found AI administration services that accepted privileged writes from a plain business user while the interface hid those screens in the browser alone. Prompt injection, direct and indirect, is tested as a route into that system rather than as the headline.

What we test

- AI admin plane: prompts, models, ingestion triggers
- Pipeline storage and business-system credentials
- Tenant isolation of conversations, documents and artefacts
- Agent and tool authority boundaries
- Prompt injection, direct and indirect
- Data path, retention and AI endpoint hardening

How it runs

- **Scope and accounts** — We agree the AI features in scope and the least-privileged test accounts.
- **Map the surface** — Endpoint inventory from the client bundle, the orchestration layer and the assistant itself.
- **Authorisation first** — Every mutating operation replayed with an ordinary user's token, then the agent boundary and injection.
- **Report and retest** — Prioritised fixes — usually one enforcement decision — then verified.

What you get

- **Executive summary** — Business-level risk, written for leadership and boards.
- **Technical report** — Every finding with severity, evidence and reproduction.
- **Prioritised fixes** — Developer-ready remediation, ranked by real risk.
- **Retest letter** — We verify your fixes and confirm closure in writing.

Methodologies and references we draw on

OWASP Top 10 for LLM Applications · OWASP Agentic Security Initiative · MITRE ATLAS · NIST AI Risk Management Framework · OWASP API Security Top 10 · OWASP WSTG · ISO/IEC 42001 and EU AI Act context

Senior testers. Evidence for every finding. Retest included. Minimum five testing days. Scope and sample reports — pxlsec.com/services/ai-penetration-testing.