

PENETRATION TESTING

AI Penetration Testing

Be confident your LLM applications and other AI systems are free of common security vulnerabilities

Commoditized access to AI is revolutionizing how work is done in every industry. But as with any rapidly commercializing technology, it also introduces new types of potential security vulnerabilities, as reflected in President Biden’s Executive Order (EO) 14110 that calls for “AI red teaming” (methods unspecified) by all government agencies.

For example, the conversational interfaces in Large Language Model (LLM) applications can be vulnerable to prompt injection, training data extraction, data poisoning, and other types of attacks. Many such applications are also highly integrated with other systems, amplifying risk by serving as a potential access point for wider infiltration by attackers.

Specialized Pen Testing for LLM Apps and Other AI Systems

Bugcrowd AI Pen Tests are designed to uncover the most common application security flaws in these areas using a testing methodology based on our open-source Vulnerability Rating Taxonomy – which draws from the OWASP LLM Top 10 while adding other flaws reported by hackers on our platform.

A thorough discovery of flaws in AI apps requires specialized knowledge, skills, and experience. Bugcrowd AI Pen Testing brings the talents of skilled, trusted security researchers with specialized experience in AI systems, rapid validation and triage, and a decade of vulnerability intelligence inside the Bugcrowd Platform to every pen test engagement.

ALL ENGAGEMENTS INCLUDE:

- ✓ Trusted, vetted pentesters with the relevant skills, experience, and track record, with support for geolocation restrictions and other special requirements
- ✓ 24/7 visibility into timelines, findings, and pentester progress through their checklist via a rich dashboard
- ✓ Testing methodology based on the OWASP Top 10 for LLMs, Web Application Hacker Handbook, SANS Top 25, CREST, WASC, PTES, and more
- ✓ Ability to handle complex applications and features including those with payment processing, purchasing, upload, and elaborate user workflows
- ✓ Validation and prioritization based on Bugcrowd’s Vulnerability Rating Taxonomy (VRT)
- ✓ Detailed auditor report
- ✓ Retesting (with one report update)

KEY POINTS OF VALUE



Start testing faster

Use the power of the Bugcrowd Platform to rapidly start your pen test in as little as 72 hours



Rely on the right talent for the job

CrowdMatch™ AI technology helps align the right pentester skills and experience for the engagement



See results in real time

Leave opaque pentesting behind; instead, view prioritized findings as they’re reported and flow them into your SDLC for rapid remediation



Methodology

Below are examples of the types of vulnerabilities that can be included in the testing checklist.

	Vulnerability	Description
	Reconnaissance	Use the application as intended, in detail, to discover all the functionalities of the underlying model
	Prompt Injection (Direct and Indirect)	Test for context switching, file parsing, image and audio data input vulnerabilities, input filter bypass, etc., and lack of authorization tracking between plugins that enables indirect prompt injection or malicious plugin usage
	Supply Chain Vulnerabilities	Review if the application uses outdated or deprecated third-party LLM components
	Sensitive Data Disclosure	Determine if the model can be manipulated to output sensitive information that should be prevented by secure output filtering
	Jailbreaking of Ethics and Content Safety Safeguards	Perform a content-related assessment similar to prompt injection testing to find privacy violations, security violations, etc.
	Insecure Output Handling	Attempt to manipulate the model to deliver malicious content to downstream apps or plugins (e.g., XSS, SSRF, SQL injection)
	Excessive Agency	Test if the model has excessive functionality, permissions or autonomy beyond its intended purpose
	Overreliance	Determine if systems excessively depend on LLMs for decision-making or content generation without adequate oversight, validation mechanisms, or risk communication
	Model Theft	Test if the model can be used to train another LLM as a surrogate
	Model Denial of Service	Look for potential for attackers to cause resource-heavy operations that lead to service degradation or high costs