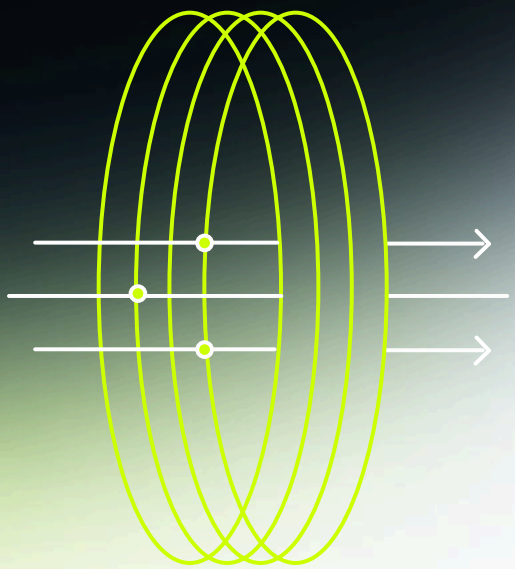
XBOW

What Is Autonomous Pentesting?



AI-based pentesting has emerged as a way for security teams to keep up with a growing attack surface, faster software delivery, and the need to test more continuously without sacrificing quality. But as an emerging category, “AI pentesting” is still used to describe very different approaches.

AI pentesting could refer to:

- AI-assisted tools (LLM wrapper around scanners)
- ML-enhanced vulnerability detection
- AI agents that reason, adapt, and chain exploits
- Fully autonomous exploitation or scripted automation
- Open source or commercial products

In practice, most offerings fall into three broad categories:

1. AI-assisted pentesting
2. Hybrid or AI-augmented pentesting
3. AI-led autonomous pentesting.

Defining autonomous pentesting

In autonomous pentesting, AI takes the lead and humans stay in control through scope, oversight, and review. In this approach, AI conducts offensive work end to end inside a structured, governed system.

In this model:

AI Systems

- Map attack surface
- Form and test hypotheses
- Execute multi-step exploit chains
- Validate findings
- Draft evidence and reporting

Humans

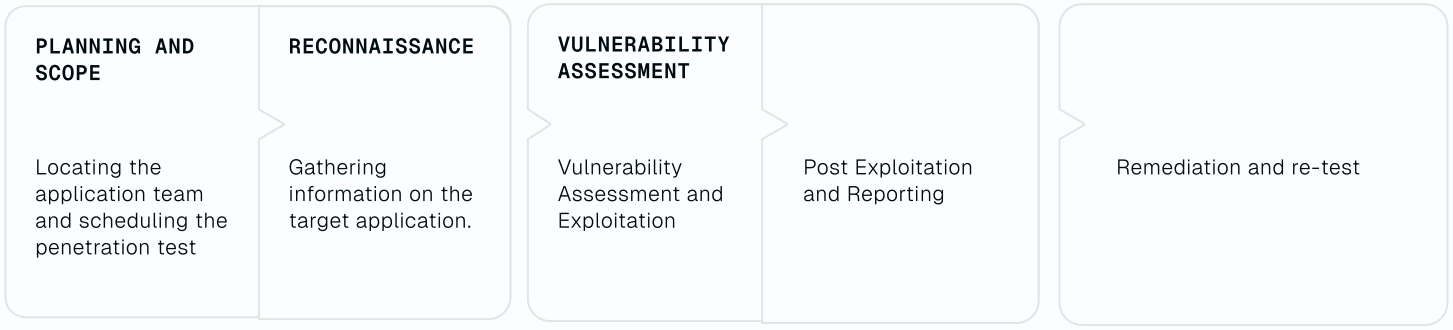
- Define scope and guardrails
- Review and prioritize results
- Investigate edge-case logic abuse
- Handle unusually creative or high-context attack paths

Sample AI-led autonomous pentesting process

Developers input scoping data into AI pentesting console (i.e. Source Code or CodeQL input etc)

AI locates, validates, and reports on exploits as well as providing full traceability documentation

AI findings can be fed into coding assistants for rapid remediation. All fixes can be retested.



The XBOW Autonomous Offensive Security Platform

The XBOW platform is a coordinated system of autonomous agents, deterministic validators, and real offensive security tooling designed for large, complex, production environments.

XBOW combines the strengths of AI agents with the structure needed to keep application testing accurate, safe, and scalable.

XBOW solution components

- **Coordinator agent**
  Identifies attack surface, selects priorities, and directs testing activity.
- **Autonomous agents**
  Pursue specific targets and attack paths, then are retired to reduce accumulated bias, drift, or compounding error.
- **Attack machine**
  Gives agents access to real offensive security tooling so testing goes beyond theoretical analysis.
- **Validator agents**
  Confirm whether findings are truly exploitable, helping reduce false positives and increase trust in results.
- **Findings and intelligence**
  Validated issues are delivered with clear evidence so security teams can understand impact and act quickly.

What about the native code scanning capabilities of AI assistants?

AI coding and code security assistants should absolutely play a role in a modern AppSec program, but an LLM is not the same thing as a penetration testing platform.

A model can help identify possible vulnerabilities, reason through attack paths, and review context. But a production-grade security tool must also validate findings, enforce safety boundaries, manage runtime environments, integrate with workflows, control costs, and earn trust from both security and engineering teams.

Ultimately, they are a tool that should be one part of a more comprehensive application security program. They can't scale to address the security needs of large enterprises, or, most importantly, have the safety guardrails needed to keep their scope in check.

Beyond simply AI applied to security tooling, autonomous pentesting is AI performing offensive security work inside a controlled system built for exploit validation, operational safety, and enterprise scale.

SCHEDULE A DEMO