ENDOR LABS

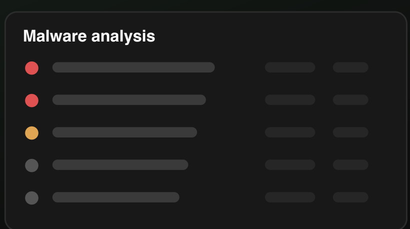
Malicious Package Detection

Catch open source malware that public feeds miss

Open source malware has moved from a rare event to a category of supply chain risk in its own right — and a malicious package can compromise a workstation the second the install runs. Malicious Package Detection scans every package against 150+ signals, flags the behaviors public feeds miss, and routes unclear findings to the Endor Labs security research team for human verification.

13.6x	150+	~40%
growth in Endor Labs malware advisories since January 2024	signals scanned across 4.5M+ open source dependencies	of known malicious packages never appear in public registries

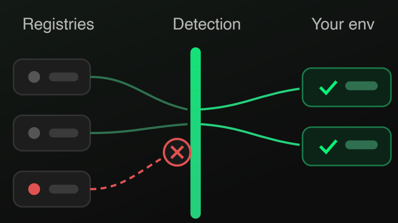
How it works



DETECT

Intelligence beyond public feeds

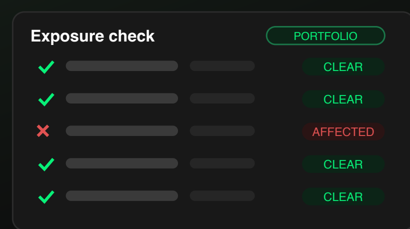
- Catch typosquatting, dependency confusion, obfuscated payloads, and code split across files
- Cover threats removed from public registries and threats not yet in any advisory feed
- Act on findings verified by the Endor Labs security research team — not raw automation



INTERCEPT

Stop installs at the moment of risk

- Block known malicious packages before a developer or AI agent runs the install script
- Intercept dependencies in the IDE via Cursor hooks and other agent-level controls
- Apply the same policy to AI agents and humans, so AI adoption doesn't bypass controls



RESPOND

Portfolio-wide answers in minutes

- Confirm or rule out exposure to a new campaign across every application in minutes, not days
- Search every project's package inventory and SBOM the moment a disclosure drops
- Build a durable malware program with cooldown and unpinned-dependency policies

WHY NOW

Malware isn't a vulnerability — there is no patch window

A malicious package executes the moment it installs: no SLA clock, no "wait until production" buffer, no CVE for vulnerability scanners to match against. Registry takedowns lag the attack, so the window between publication and disclosure is exactly when consumers get hit — and AI coding agents installing dependencies autonomously widen that window. Campaigns like Shai-Hulud and the wave of npm and PyPI account takeovers are why malware response needs to be a program, not a fire drill.