

Hundreds of CVEs. One upgrade that breaks the build. Nothing gets fixed.

Fix what is reachable. Upgrade without fear.

Xygeni SCA ranks every open-source component by reachability, exploitability and AI urgency, shows what an upgrade will break before you merge it, and flags malicious packages before a signature exists.

90%

reduction in vulnerability prioritization time reported
by Fintonic (Spain, fintech).

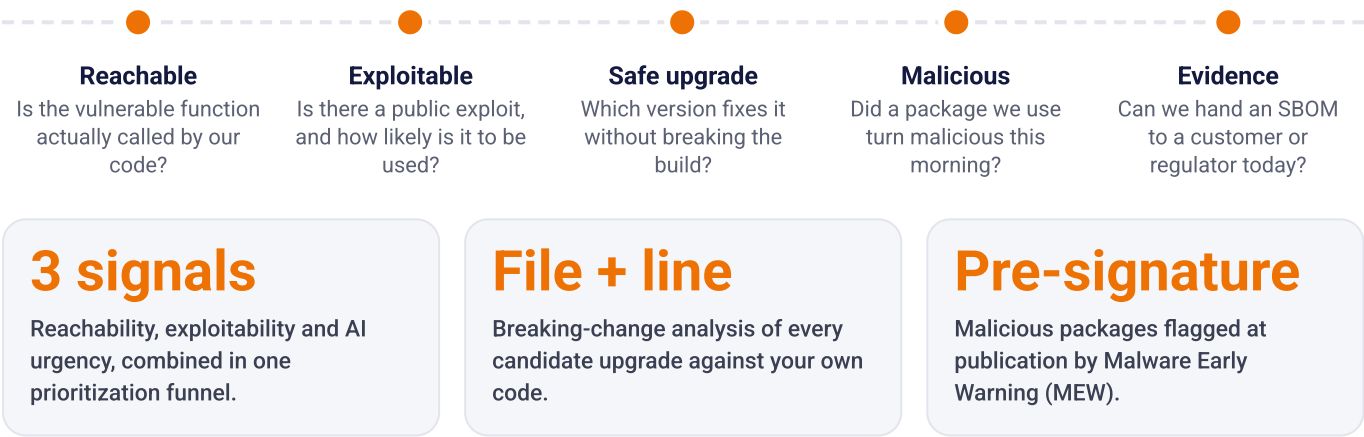
THE PROBLEM

Open-source risk is not a list of CVEs. It is a decision about what to fix, and how.

Every application carries hundreds of open-source vulnerabilities, and many of them sit in code the application never calls. Teams cannot tell which ones matter, upgrades break builds, so fixes get postponed. And the attacks that hurt most today, malicious packages, typosquats and dependency confusion, often have no CVE at all.

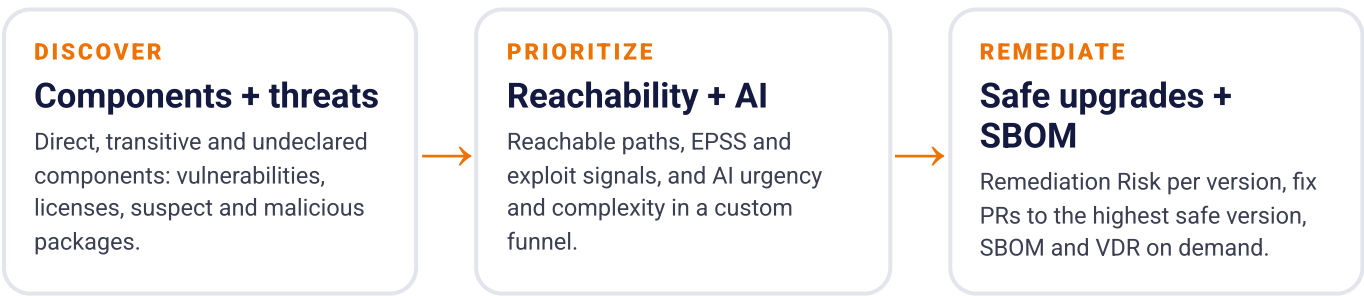
Regulation now asks for the evidence: SBOMs and vulnerability handling under the EU Cyber Resilience Act, NIS2 and DORA. Xygeni SCA turns the dependency tree into a ranked, fixable plan, with the SBOM as a by-product.

FIVE QUESTIONS MOST TEAMS CANNOT ANSWER TODAY



ONE PRODUCT, THREE STEPS

Discover. Prioritize. Remediate.



Discover every component. **Prioritize** what is reachable. **Remediate** without breaking.

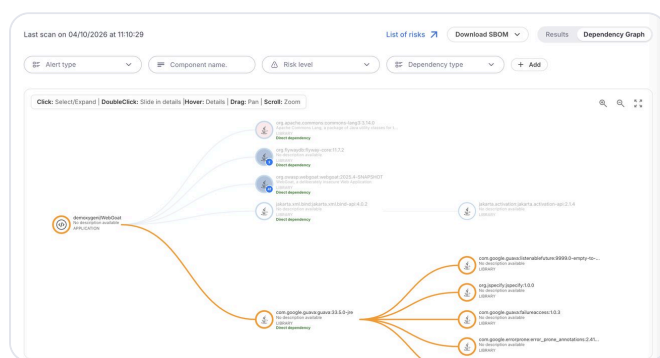
Without Xygeni	With Xygeni
Hundreds of CVEs, all "critical"	Reachable and exploitable first
Upgrades postponed for fear of breakage	Breaking changes known before the PR
Malicious packages found after install	Flagged at publication, blocked in CI
SBOMs assembled by hand	CycloneDX or SPDX in one click

DISCOVER · COMPONENTS AND THREATS

See every component, including the ones nobody declared

Dependency graph, vulnerabilities, licenses and supply-chain threats in one scan.

Xygeni resolves direct and transitive dependencies for each ecosystem, assesses undeclared components and builds the full dependency graph. Every component is checked against vulnerability advisories, license policies and Xygeni's malware intelligence, and suspect dependencies are flagged even when no CVE exists.



Dependency graph of an application: direct and transitive components, with risk per node and SBOM download.

Malicious Packages DB: quarantined packages and the code evidence behind each verdict.

- Vulnerabilities** from NVD and ecosystem advisories, with aliases, plus risky packages without CVEs.
- Malware Early Warning:** packages flagged at publication show up as malware alerts on your components.
- Container images:** OS and application packages, including images in Nexus and OCI registries.
- Suspect dependencies:** typosquatting, dependency confusion, suspicious install scripts and unscoped internal components.
- Licenses:** identification per component, and guardrails that fail builds on disallowed licenses.
- Broad and growing ecosystem coverage:** Maven, Gradle, npm, Yarn, pnpm, NuGet, pip, uv, Poetry, Go, Composer, RubyGems, Swift and Dart.

Direct and transitive

Undeclared components

Incremental scans

Dependency graph

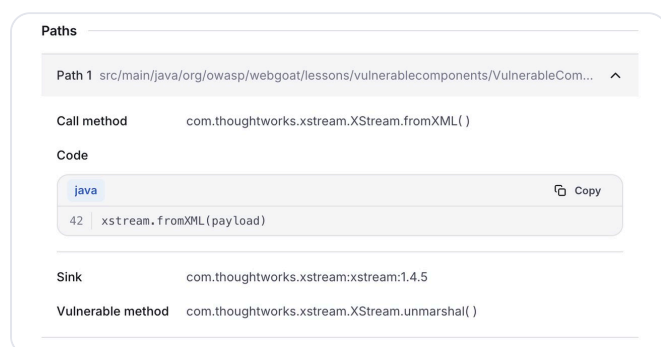
For the AppSec lead: one view of every dependency, including the threats that CVE feeds miss.

PRIORITIZE · REACHABILITY AND AI

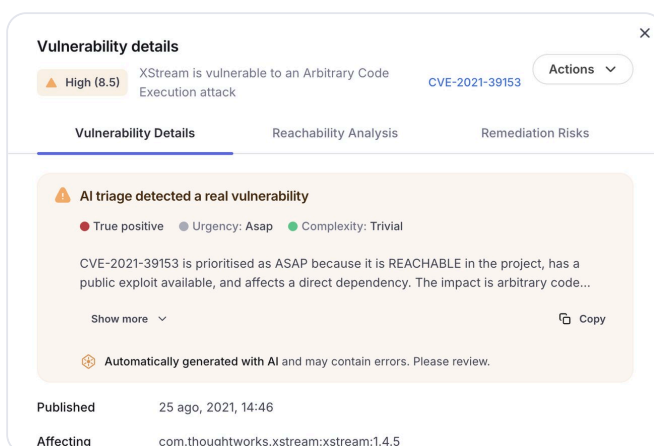
Fix what is reachable and exploitable first

Reachability, exploitability and AI urgency, in a funnel you control.

Many vulnerable functions are never called by the application. Reachability analysis determines whether they are, and shows the path. Combined with EPSS and known exploits, and with AI Triage urgency and complexity, the SCA funnel turns hundreds of CVEs into an ordered plan.



Reachability analysis: the call path from application code to the vulnerable method in the dependency.



AI Triage on a reachable CVE with a public exploit: urgency, complexity and the reasoning behind them.

- **Reachability:** whether vulnerable code is invoked, with the reachable paths shown, for major languages.
- **AI urgency:** from EPSS, exploits, CVSS, reachability and direct or transitive scope.
- **Custom funnels:** stages and properties such as business criticality and exposure.
- **Exploitability:** EPSS, known exploited vulnerabilities and exploit availability as funnel criteria.
- **AI complexity:** from the candidate upgrade's compatibility and bump type.
- **Third-party SCA results** imported and prioritized with the same signals.

Reachable

Exploit available

Fixable

In application code

Direct or transitive

Unreachable is not ignored. It is one funnel stage among several. Those findings stay visible and deprioritized, and effort goes where an attacker can actually get in.

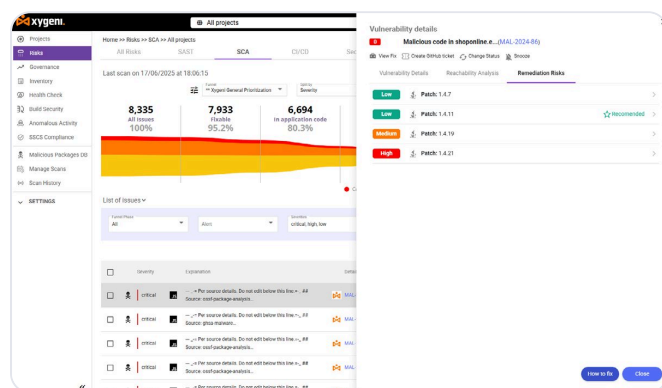
For the CISO: a remediation plan ranked by real exposure, not by CVSS alone.

REMEDiate · SAFE UPGRADES

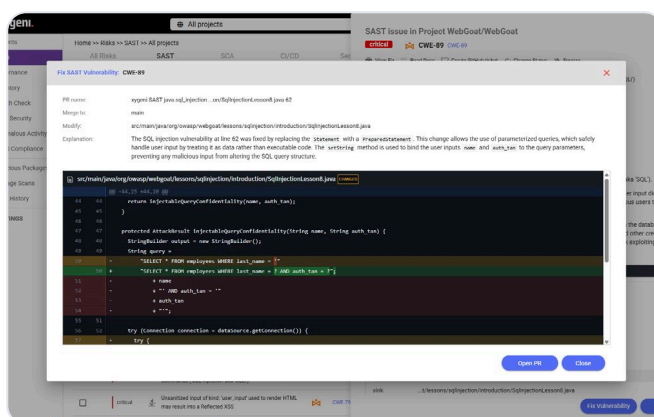
Upgrade without fear, and prove it with an SBOM

Remediation Risk, fix pull requests and bulk automation. SBOM and VDR on demand.

Teams postpone upgrades because they break builds. Remediation Risk analyzes every candidate version against your own code before the pull request: deleted and modified methods still in use, the files and lines affected, the risks it fixes and the ones it introduces. Then the fix arrives as a pull request, one at a time, in bulk or on a schedule.



Remediation Risk: candidate versions ranked from low to high risk, with the recommended one marked.



Fix Vulnerability: the selected version applied and a pull request ready for review.

- **Risk level per version:** low, medium or high, by compilation and runtime impact.
- **Fix PR to the highest safe version,** from the finding or the Remediation View.
- **SBOM** in CycloneDX or SPDX and Vulnerability Disclosure Reports, from the CLI or the inventory.
- **Affected source paths,** down to file and line, before you upgrade.
- **Bulk PRs** across categories, and Xygeni Bot automation by severity, language or rule.
- **In the IDE** and inside AI coding agents through the MCP server.

Malware Defense

MEW intelligence on every component, and Shield to block malicious packages on developer endpoints.

ASPM

SCA findings next to SAST, secrets, IaC and CI/CD in one risk model and one remediation view.

Compliance evidence

SBOMs and VDRs for CRA, NIS2, DORA, PCI DSS 4.0 and US EO 14028.

For VP Engineering: dependency upgrades that do not turn into a week of broken builds.

WHY XYGENI FOR SCA

Discover every component. Prioritize what is reachable. Remediate without breaking.

Xygeni SCA ranks open-source risk by what an attacker can actually reach, shows what each upgrade will break before you merge it, and catches malicious packages before a signature exists.

Reachability first

Vulnerable functions your code actually calls, with the path to prove it.

Remediation Risk

Breaking changes per candidate version, down to file and line, before the PR.

Malware before signatures

MEW flags malicious packages at publication, and guardrails block them in CI.

SBOM as a by-product

CycloneDX, SPDX and VDR from every scan, ready for customers and regulators.

Why security leaders choose Xygeni

Proven in production

Banco Serfinanza reports around 90% less time identifying, prioritizing and assigning findings; Fintonic, a 90% reduction in prioritization time.

One platform

Code, open source, secrets, IaC, pipelines, running apps and AI in one inventory, one risk model and one workflow.

Extend, don't replace

Xygeni's AI layer triages, explains and fixes findings from the security tools you already run.

Built for regulated environments

European vendor. SaaS, on-premise or air-gapped scanning, with any LLM, including self-hosted models.



Open-source security that fixes what matters, without breaking the build

Discover. Prioritize. Remediate. One platform.

[Request a PoC](#)

[Book a demo](#)

- Keep your current SCA: Xygeni prioritizes its findings
- SaaS, on-premise or air-gapped scanning
- xygeni.io

RECOGNIZED IN 2026



Hot Company in Application Security Posture Management 2026

Global InfoSec Awards · Cyber Defense Magazine