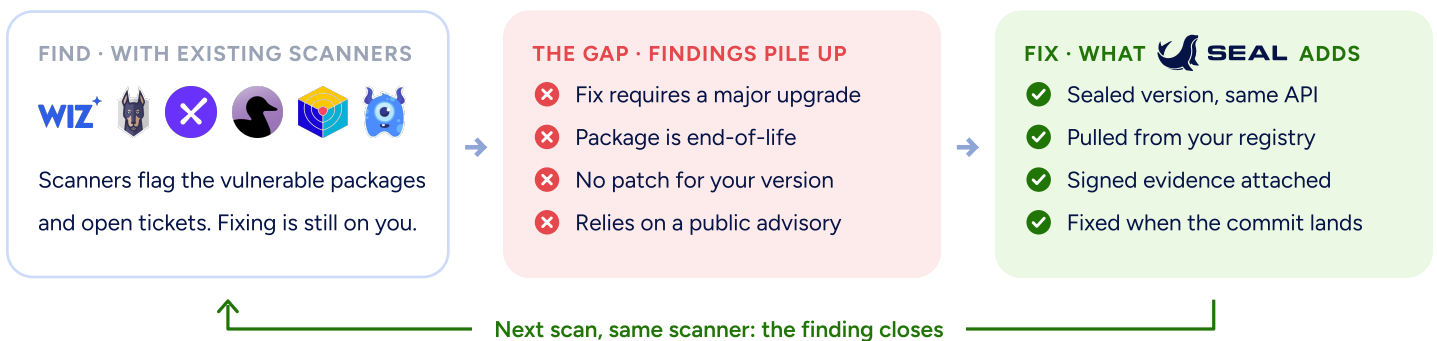


Fix open source vulnerabilities without touching the platforms that move money.

Scanners wait for a public advisory, then hand you a ticket. Seal fixes the vulnerability on the version you already run, often before the advisory is out: no forced upgrade, no regression risk to core banking, payments, or trading systems. Every fix comes with the signed evidence your auditors, examiners, and regulators ask for.

Your scanner finds it. Seal fixes it.



What banks need that SCA tools can't provide



No forced upgrade

Each patched artifact is a drop-in replacement: same major version, same API, same behavior. Change control approves a security patch, not a migration.



End-of-life coverage

Legacy and acquired systems run packages whose maintainers stopped publishing. Seal backports the fix anyway, so "no patch available" stops being an accepted risk.



Pre-CVE program

Seal backports the upstream security fix when the commit lands, not when the CVE is published. Where no community fix exists, Seal's researchers write one.

PROTECTED BY SEAL · ACROSS SEAL'S FINANCIAL SERVICES CUSTOMERS


\$15T+ in assets under custody, administration, and management


\$4T+ in total assets, including two global systemically important banks


\$1.7T in annual payments processed on software Seal keeps patched

TRUSTED BY



"Thanks to Seal's product, we swiftly addressed security vulnerabilities in our outdated code packages, saving us valuable time."

Gad Meyer · Director of Software Engineering 

The patch exists upstream. Seal brings it to your version.

Only the security fix, backported onto the exact version you run. The finding closes, nothing else changes.

1 · Isolate

only the lines
that fix the CVE

2 · Backport

onto the version you
run, EOL or not

3 · Verify

build, test, prove
the CVE is gone

4 · Deliver

into Artifactory,
Nexus, or CI

5 · Close

your scanner sees
a fixed version

A scanner-only program vs Seal

✗ Scanner-only program

spring-beans 4.3.30 → CVE-2022-22965 · critical → major upgrade → Spring 5.x

Finding logged. Upgrade ticket opened. Blocked pending a regression cycle that never gets scheduled.

✓ Seal's approach

spring-beans 4.3.30 → CVE-2022-22965 · critical → spring-beans 4.3.30-sp1

Sealed package pulled from your registry. CVE closed, same version, evidence attached.

Runs inside your controls, from cloud to air-gapped

Air-gapped · On-prem

Sealed packages served from your internal registry. Hosts never call Seal.

Malicious packages · Cooldowns

Malicious versions swapped for the last legitimate one. New releases held back.

Transparent replacement

Builds pull the sealed version automatically. No manifest edits, no PRs.

Org-wide policy · Seal CLI

One Sealing Rule in the platform applies to every pipeline that runs the CLI.

20,000+

unique CVEs patched, 9 ecosystems

72 hrs

from disclosure to sealed package

6.5 lines

average sealed patch size

FULL-STACK COVERAGE · 9 ECOSYSTEMS INCLUDING LINUX PACKAGES

[Java](#)[Python](#)[JavaScript](#)[C/C++](#)[.NET](#)[Go](#)[PHP](#)[Ruby](#)[Linux packages](#)

INTEGRATES WITH · YOUR SCANNERS, REGISTRIES, AND CI/CD

[JFrog](#)[GitHub](#)[GitLab](#)[Azure DevOps](#)[Docker](#)[WIZ](#)[Snyk](#)[Checkmarx](#)[Black Duck](#)[Aqua Trivy](#)[Grype](#)[GitHub Advanced Security](#)

EXAMINATION-READY EVIDENCE · SIGNED RECORDS + VEX EXPORTS

[DORA](#)[PCI DSS 4.0](#)[NYDFS Part 500](#)[FFIEC / OCC](#)[SOX ITGC](#)[SOC 2](#)

Vulnerability-free open source.

seal.security

[Book a demo](#)