OEM Solution Brief

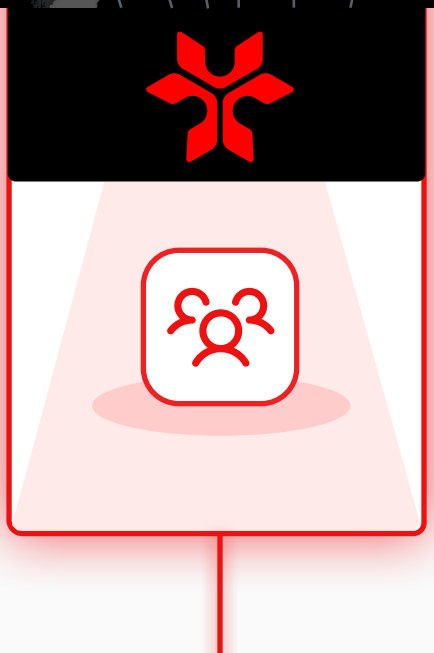
NETRISE

PROVENANCE

Who's Inside Your Software?

Gain insight into dependencies - and enforce policies to reduce supply chain risk.

Manage risk in the third-party software your teams choose and ship.



THE CHALLENGE

You Don't Know Who Is in Your Software

Even when your team generates SBOMs and runs software composition analysis (SCA), basic questions stay unanswered:

- Do you know who actually contributes to the open-source and third-party components you depend on?
- Can you see when those components are maintained by high-risk contributors, organizations, or nation-states?
- Can you quickly find all libraries to which malicious actors have contributed to understand their blast radius?
- Can you quickly spot when a critical component's repository becomes unhealthy or changes hands unexpectedly?

These gaps persist because:

- Traditional SCA and SBOMs surface known vulnerabilities but don't provide enforceable rules for maintainer, origin, or repository health.
- Threat intelligence and advisories about risky contributors or organizations are not linked to the components in SBOMs or dependency graphs.
- Repository health signals - activity, churn, maintainer concentration- are hard to evaluate and enforce consistently in CI and intake.
- Transitive dependencies, mirrors, and forks obscure the canonical source repository and make any single compromise impact more of your stack.

If you can't confirm software origin or how risk spreads, you're guessing about what you ship, how you respond to incidents, and which suppliers to trust.

Why You Need Provenance Intelligence

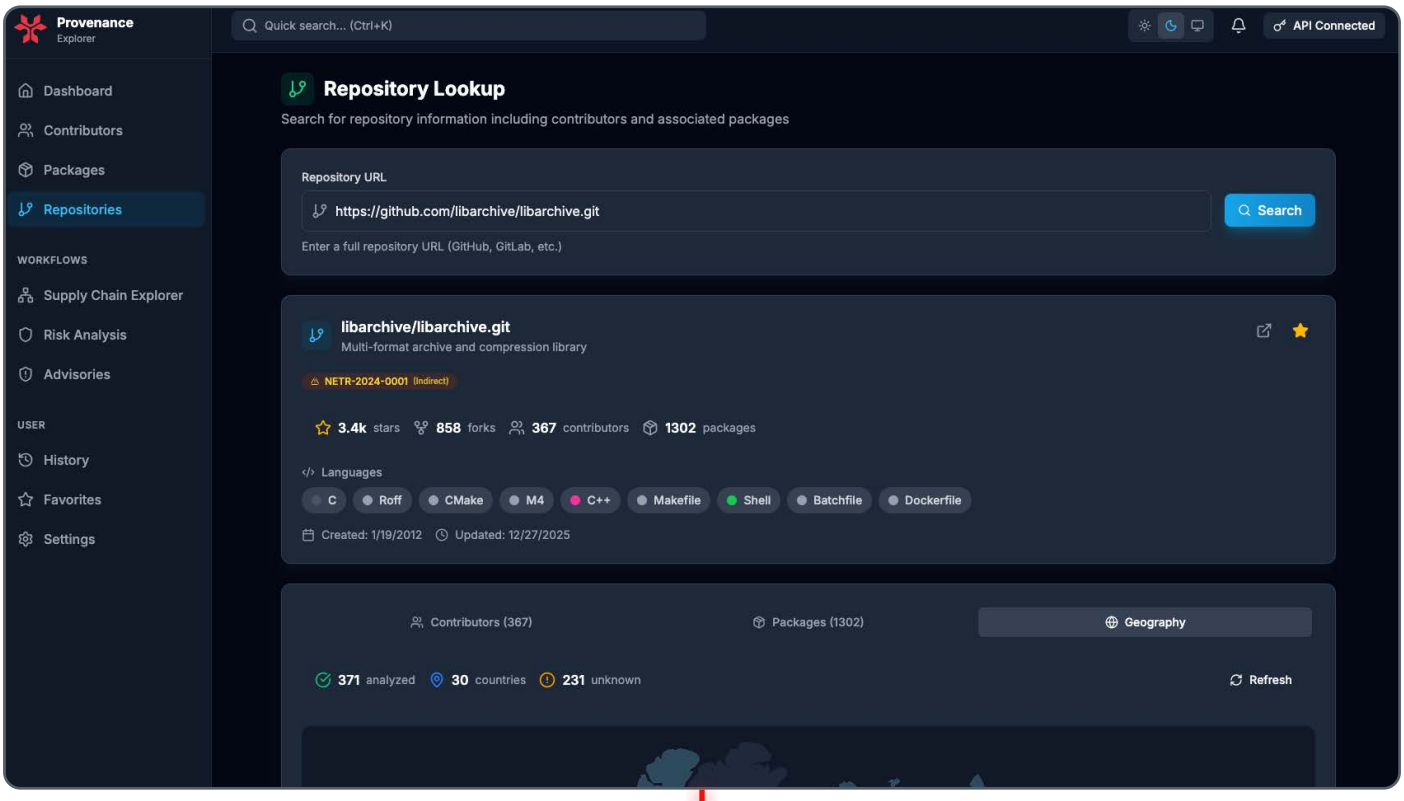
Modern software depends on third-party and open-source components. SBOMs and SCA show what is present, not who maintains it, where it originates, or how it spreads across services.

NetRise Provenance reveals maintainers, organizations, countries of origin, and contribution patterns that indicate risk, correlating this with dependency graphs and threat intelligence so teams can enforce policies, choose safer libraries, and harden builds.

THE SOLUTION

NetRise Provenance: Trust Intelligence to Secure Your Software Supply Chain

NetRise Provenance unifies origin, maintainer, and risk signals by mapping packages to canonical repositories and maintainers, correlating advisories with independent repository security checks, and using repository health signals to enforce policy-driven guardrails and help teams choose safer libraries.



NetRise Provenance: A System of Intelligence for Software Trust

Across the software and firmware you ship, NetRise Provenance helps your team:

See the real source of your code

Trace packages and transitive dependencies to canonical repos so you know where components originate and avoid spoofed, forked, or abandoned sources when selecting dependencies.

Know who maintains your dependencies

Identify maintainers, organizations, and country footprint so you understand who is behind dependencies and whether each project's contributor mix fits your risk tolerance.

Detect high-risk contributors

Use advisories, sanctions, and threat intelligence tied to maintainer identities to flag dependencies linked to high-risk actors and prioritize libraries to replace, sandbox, or scrutinize.

Make risk-based decisions in CI and intake

Use the Provenance policy engine to enforce policy-as-code in CI and intake - block, quarantine, or route risky components to review.

Product Overview

Policy Engine

Define and enforce declarative policies using provenance, advisories, repository health, geography, and contributor risk signals to block or flag higher-risk components before release.

Maintainer and Organization Attribution

Attribute packages to maintainers and organizations, including country footprint, so teams know who they're pulling code from and can apply procurement and intake standards consistently.

Canonical Source Mapping and Dependency Graphs

Map package identifiers to canonical source repositories and visualize dependency and reverse-dependency relationships to understand blast radius when a library, repo, or maintainer becomes risky.

High-Risk Contributor and Advisory Signals

Integrate advisories, sanctions, threat intelligence, repository health signals, and repository security checks with maintainer identity and country footprint to flag higher-risk dependencies in builds and reviews.

NetRise Provenance delivers the identity, dependency, and risk context your teams need to decide which software to trust.

Why NetRise Provenance Stands Apart

Unified, API-Ready Coverage

Access one standards-based API unifying ecosystems like PyPI backed by intelligence on billions of components.

Comprehensive Provenance Insight

See where components originate, who maintains them, and which organizations and countries back your software.

Contextual Risk Intelligence

Use metadata, contributor attribution, and repo health signals and security checks to focus on risky dependencies.

Faster Incident Response

Map dependency relationships to understand blast radius and identify affected services or products.

Seamless Workflow Integration

Plug REST APIs and policy enforcement into CI/CD pipelines, SBOM workflows, and vulnerability tools without changing workflows.

Common Challenges NetRise Provenance Solves

Challenge	How NetRise Helps
You cannot see who actually maintains your open-source and third-party components.	Shows maintainer and organization details, including country footprint, so teams update allowlists and denylists confidently.
You cannot tell when dependencies link to high-risk contributors, orgs, or countries.	Correlates contributors, organizations, countries, and advisories - then enforces policies to block or flag risky dependencies.
Supplier components arrive without provenance details.	Maps components to canonical repos and maintainers and applies intake policies - strengthening supplier onboarding.
You cannot see the services affected when dependencies or maintainers become risky.	Maps dependencies to blast radius and enforces policies when packages, repos, or maintainers become risky.

NetRise Provenance highlights high-risk components so teams choose safer libraries and focus testing where needed.

***Who's* inside your software?** Let's Find Out

