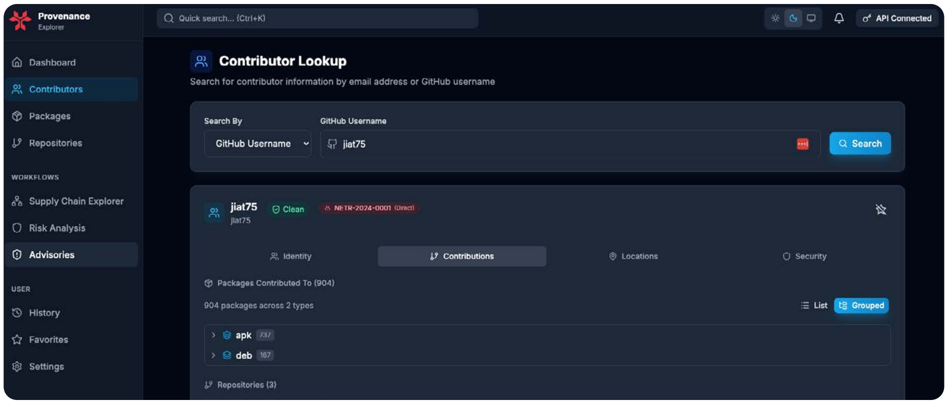
NETRISE

Data Sheet

NetRise Provenance

Understand risk associated with open-source software components: origin, maintainers, and repository health across ecosystems. Define and enforce policies across dependencies.

NetRise Provenance reveals who maintains the open-source software you rely on, where it originates, and how risk propagates across libraries and repositories - then enforces consistent policies for builds, procurement, and incident response.



Contributor Lookup

Search for contributor information by email address or GitHub username

Search By: GitHub Username

GitHub Username: jiat75

jiat75 Clean NETR-2024-0001 (Direct)
jiat75

Packages Contributed To (904)

904 packages across 2 types

- apk 737
- deb 167

Repositories (3)

Why NetRise Is Different

NetRise Provenance turns software supply chain intelligence into consistent action. By unifying ecosystem signals and enforcing organizational policies, security teams standardize how third-party dependencies are evaluated, reduce manual investigation, and quickly assess impact when new software supply chain risks emerge.

Enforce Software Trust Standards

Standardize decisions by enforcing organization-wide rules across intake and developer builds.

Prevent Hidden Supply Chain Risk

Reduce surprises by identifying abandonment, churn, and weak security practices early.

Unify Software Trust Intelligence

Unify ecosystems—from OS packages to registries like PyPI—so teams stop stitching sources together and get answers faster.

Understand Blast Radius Fast

See propagation paths to prioritize fixes that reduce downstream impact.

Respond to Supply Chain Incidents Faster

Shorten response time by mapping impact across products and vendors within minutes.

Reduce Geopolitical & Entity Exposure

Identify dependencies tied to high-risk regions, contributors, or organizations to reduce sanctions and exposure risk.

Policy Engine

Enforces declarative policies using sanctions, geography, advisories, repo posture, maintenance risk, and custom rules.

Repository Health & Security Signals

Surfaces repo hygiene, security posture, activity signals, metadata, and risk insights that reveal fragile or risky dependencies.

Provenance & Lineage Mapping

Maps packages to canonical repositories and reconstructs lineage across ecosystems to reveal origins and evolution.

Contributor & Organization Attribution

Identifies contributor identities, affiliations, and locations to reveal organizational and geographic provenance.

Supply Chain Impact Analysis

Maps dependencies and reverse-dependencies to size blast radius when packages, repos, or maintainers are implicated.

Tailored Solutions for Your Role

Software Builders

- Evaluate libraries using maintainer identity, repo health, and policy rules before inclusion.
- Continuously monitor dependencies for higher-risk contributors, organizations, or regions; enforce thresholds.
- Trace compromised components quickly and apply policy guardrails to guide remediation.

Software Consumers

- Assess vendor software using maintainer, organization, country, and repo health signals.
- Apply policies to flag or block higher-risk components during onboarding and renewals.
- Enrich SBOMs with provenance, health, and policy outputs for risk-focused decisions.

Red Team Operations

Red teams rely on open-source tools that can be targeted by malicious actors. NetRise Provenance surfaces maintainer identity, organizational and country context, repository health signals, and policy controls so teams can avoid higher-risk tools before use.

Deploy with Ease

Standards-Based RESTful Design

Follows OpenAPI specification for predictable, consistent integration.

Secure, Reliable Access

Includes authentication, versioning, and robust error handling.

Ecosystem-Ready API

Integrates ecosystem data, including OS packages and PyPI.

Open, Extensible Design

Adapts to evolving data models and policy rules.

Who Uses NetRise Provenance?

Chief Information Security Officer (CISO)
Prioritize vendors and software using maintainer, organizational, and geopolitical risk signals.

Enterprise Security Engineer
Overlay risk on SBOMs and enforce blocking policies.

Incident Response Manager
Identify blast radius associated with malicious contributors, set policies, and implement controls.

Third-Party Risk Manager
Augment suppliers' attestations and set policies and controls with maintainer, geography, and repo health.

SBOM Vendor / Product Manager
Embed provenance, repo health, and policy signals into SBOMs.

National Security Analyst
Trace components to maintainers and geographic regions to assess national security exposure.

Product Security / DevSecOps Lead
Apply provenance, repo health, and policies to control build inputs.

***Who's* Inside Your Software?**

Let's Find Out