Solution Brief

kusari

Kusari Inspector
AI Code Review + Dependency Management

Kusari Inspector provides CLI, IDE or PR code quality reviews, dependency and security analysis with full context in real-time. Get go/no-go feedback, code and dependency safety insights, and remediation next steps, before code is merged.

Kusari Inspector brings together a powerful combination of code review and security standards, context-aware AI and deep dependency analysis to empower developers to identify, manage, and mitigate software supply chain risks early and effortlessly within their workflow.

Guardrails, not roadblocks

Now, developers can ensure quality and safety when using open source or AI. Use actionable recommendations to remediate vulnerabilities, leaked secrets, workflow issues, risky dependencies, license concerns, and other threats.

Code and dependency review that has your back

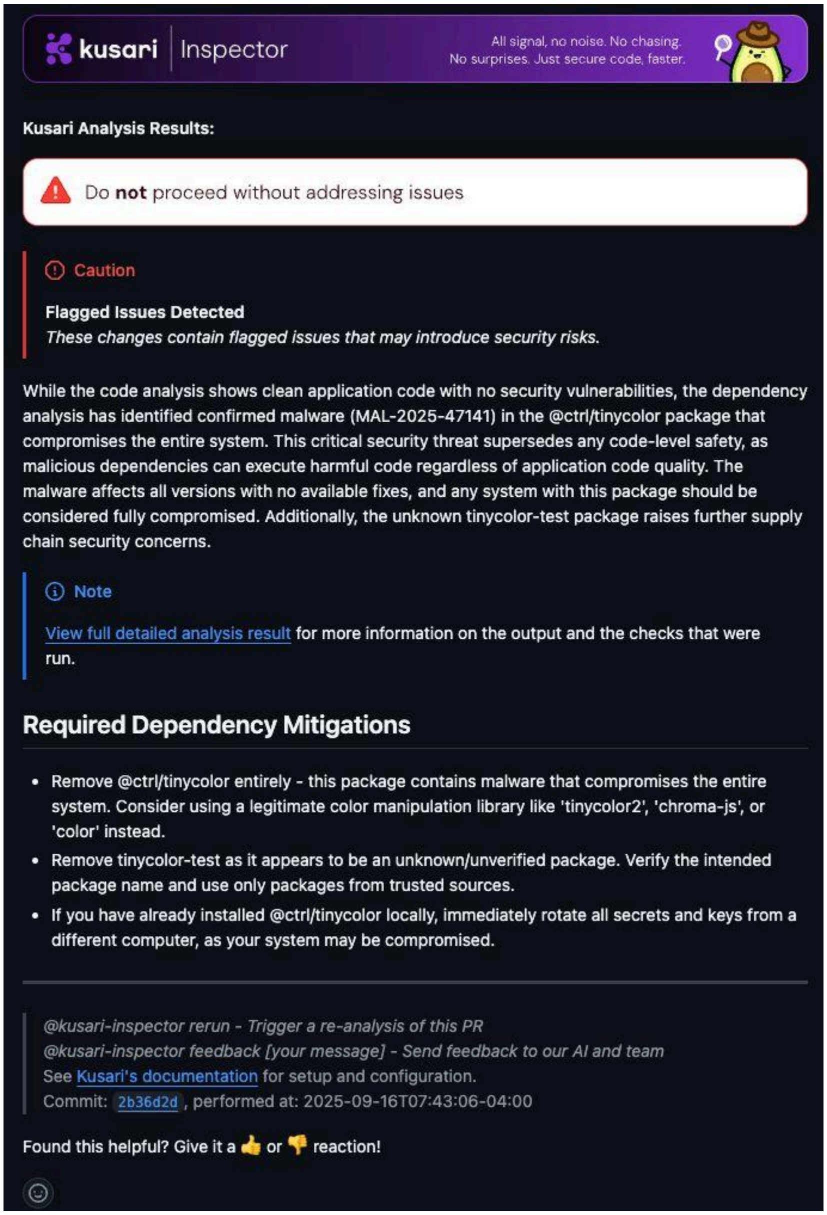
Scan the QR code to install Kusari Inspector in seconds. It can integrate with the Kusari Platform for greater insights, linking source code commits to runtime events.

Kusari Inspector checks for:

- Direct and transitive dependencies
- Known vulnerabilities, including
  - Severity (CVSS)
  - Likelihood of exploit (EPSS)
  - Known exploited vulnerabilities
- Typosquatted dependency names
- GitHub workflow security issues
- Common code weaknesses via static analysis
- Credentials and other secrets
- Software licenses
- Dependencies’ repository security posture

Kusari Inspector is a CLI, IDE and GitHub tool supporting:

- Golang (Go) - go.mod, go.sum
- Java (Maven) - pom.xml, gradle.lockfile, buildscript-gradle.lockfile
- .NET (Nuget)
- Node.js (NPM) - yarn.lock, package-lock.json
- Python (PyPI) - requirements.txt, poetry.lock, pipfile.lock
- Ruby (RubyGems) - gemfile.lock
- Rust (Cargo) - cargo.lock



kusari | Inspector
All signal, no noise. No chasing.
No surprises. Just secure code, faster.

Kusari Analysis Results:

⚠ Do **not** proceed without addressing issues

> ⓘ Caution
>
> **Flagged Issues Detected**
> *These changes contain flagged issues that may introduce security risks.*

While the code analysis shows clean application code with no security vulnerabilities, the dependency analysis has identified confirmed malware (MAL-2025-47141) in the @ctrl/tinycolor package that compromises the entire system. This critical security threat supersedes any code-level safety, as malicious dependencies can execute harmful code regardless of application code quality. The malware affects all versions with no available fixes, and any system with this package should be considered fully compromised. Additionally, the unknown tinycolor-test package raises further supply chain security concerns.

> ⓘ Note
>
> View full detailed analysis result for more information on the output and the checks that were run.

Required Dependency Mitigations

- Remove @ctrl/tinycolor entirely - this package contains malware that compromises the entire system. Consider using a legitimate color manipulation library like 'tinycolor2', 'chroma-js', or 'color' instead.
- Remove tinycolor-test as it appears to be an unknown/unverified package. Verify the intended package name and use only packages from trusted sources.
- If you have already installed @ctrl/tinycolor locally, immediately rotate all secrets and keys from a different computer, as your system may be compromised.

> *@kusari-inspector rerun - Trigger a re-analysis of this PR*
> *@kusari-inspector feedback [your message] - Send feedback to our AI and team*
> See Kusari's documentation for setup and configuration.
> Commit: `2b36d2d`, performed at: 2025-09-16T07:43:06-04:00

Found this helpful? Give it a 👍 or 👎 reaction!

To submit feedback or feature requests, please email support@kusari.dev.

To learn more about Kusari, visit www.kusari.dev