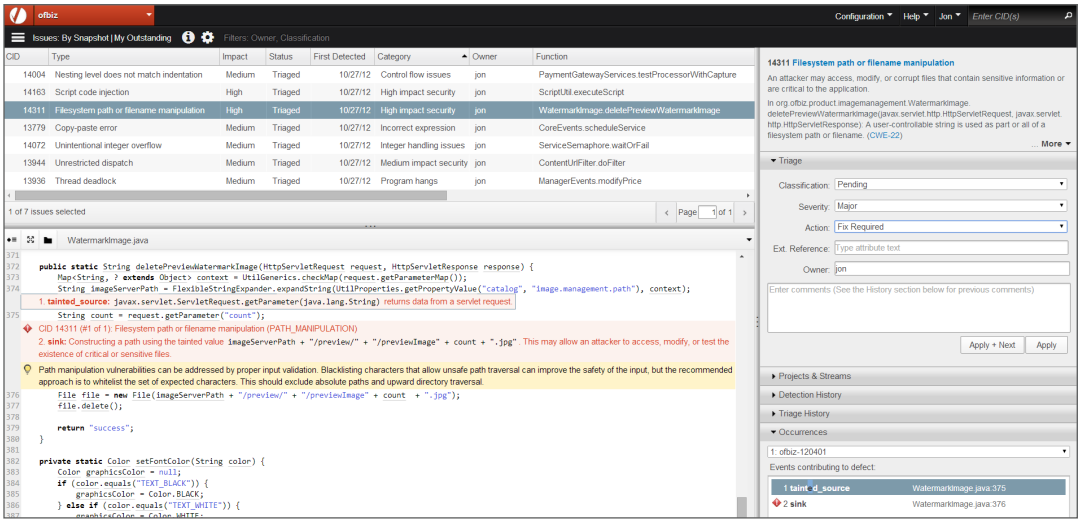
Coverity Software Testing for Java

The Coverity® platform helps identify, manage and remediate critical quality and security defects and improve the overall efficiency of testing efforts, reducing the cost, time and risk of software errors.



Prescriptive remediation advice makes it easier for developers to address critical security defects.

Solution Overview

The Coverity Software Testing Platform enables developers to build quality and security testing into the development process at the earliest stage and Quality Assurance (QA) to focus their automated and manual testing efforts based on change impact without requiring access to the source. This reduces costs and risks, accelerates time to market, enhances customer satisfaction and increases revenues.

Key Capabilities

Deep Code Intelligence with Sophisticated Analysis Techniques

The Coverity Software Testing Platform applies patented analysis techniques which enable Development to automatically test code as it is written and accurately detect issues. It provides full path coverage, ensuring that every line of code and potential execution path is tested. Full interprocedural analysis enables developers to find and fix defects that cross function boundaries versus simple style violations or superficial issues.

The Coverity platform utilizes sophisticated techniques to minimize false positives and provides actionable remediation advice. Users can analyze large, complex Java code bases quickly by leveraging parallel analysis. This means even the most complex code bases can be scanned regularly, which allows teams to adopt software testing as part of a nightly or continuous build process. Incremental analysis saves time by enabling developers to quickly reanalyze modified code, directly from their desktop.

The Coverity platform also enables QA to focus automated and manual testing efforts based on the impact of change without requiring access to the source code. Teams can quickly see which areas of the code have been tested and what still needs to be tested to avoid regression risk.

IDENTIFY AND REMEDIATE CRITICAL ISSUES

- Resource leaks
- API usage errors
- Concurrent data-access violations
- Performance and maintainability issues
- Web application framework defects
- Android API usage errors
- FindBugs coding style defects
- Critical OWASP Top 10 web application security defects
- Missing unit tests in high-risk areas of the code
- Holes in automated and manual testing plans that could introduce regression risk

Manage and Remediate Java Issues

Coverity Connect is the collaborative issue management interface used by developers to efficiently manage all issues surfaced by Coverity or third party analysis engines to resolution, within a unified, workflow. Defects can be automatically assigned to the appropriate developer for resolution, and users can prioritize and filter issues based on criticality and impact. Coverity Connect provides source code navigation that helps developers understand the exact path to the defect and identifies every occurrence of the defect across shared code. The platform also enables defects found by the Coverity solution to be viewed and triaged within the SonarQube environment.

Identify Critical Quality and Security Defects

The Coverity platform identifies critical quality and security defects in the developer's workflow with accurate, actionable remediation guidance. It enables them to quickly and efficiently troubleshoot and fix the defects that matter, such as concurrency issues, API usage errors and resource leaks, before the code makes its way to QA. The platform also finds OWASP Top 10 and CWE errors without requiring developers to become security experts. The analysis results are highly accurate so developers don't waste time managing a large volume of false positives. It provides actionable information and precise remediation guidance, showing them the right way to fix the defect and the best place in the code to fix it. This builds quality and security into the development lifecycle, which reduces the cost, time and risk of security vulnerabilities.

The Coverity platform provides out-of-the-box support for preventing quality defects related to common technologies and frameworks such as Java SE, Android, JDBC, Swing, Eclipse SWT, Apache Commons, Spring, Restlet, EJB, GWT RPC and Hibernate. Coverity Connect also manages FindBugs issues, and analysis packs are available for integration of additional analysis results, providing developers with a single location to triage and remediate defects.

These packs provide capabilities such as architectural analysis and dynamic analysis to find concurrency issues such as race conditions, deadlocks and resource leaks in Java programs.

The Coverity platform can be extended to enable easy integration of third party tools for simplified code analysis and management. The platform allows rapid integration of critical tools and systems such as source control management, build and continuous integration, bug tracking, integrated development environments and application lifecycle management solutions.

Improve Automated Testing

Automated testing can be effective but is often inefficient. Code coverage is commonly used as the metric to determine "adequate vs. inadequate" testing. While coverage tools provide basic guidance about the percentage of code that's covered by an automated test, they lack the ability to understand the ripple effect of change or help teams prioritize their automated testing efforts.

Coverity Test Advisor – Development Edition enables teams to focus and prioritize their testing efforts on the most critical areas of the code. As part of the Coverity platform, it provides the code intelligence required for teams to establish and enforce testing policies that define what must be tested, such as all new code and legacy code impacted by change, as well as what can be ignored, such as exception handling or debugging code. Users also receive guidance about which existing tests they should run based on the impact of change. Violations of established policies can be automatically assigned to the appropriate team member for quick and efficient remediation, and stage gates can be implemented to validate when teams have adequately tested the code. With Coverity Test Advisor – Development Edition, teams get better visibility into what tests they need to write and run, improving overall release predictability and lowering the risk of software failures.

Coverity Test Advisor – QA Edition provides QA with intelligent change impact analysis for software testing. By monitoring the execution of applications and aggregating results of both manual and automated tests, teams can prioritize tests and identify which are most critical based on the changes to the source code. The Impact Analyzer evaluates the impact of modifications to the test plan, allowing QA to eliminate redundant tests and generate scenarios that eliminate gaps in test coverage.

This "what if" potential allows teams to model changes before the testing cycle begins and use that information to properly schedule risky changes to the product so they can be adequately tested.

Drive Adoption and Mitigate Risk

Coverity Policy Manager allows organizations to define and enforce a consistent standard for code quality, security and testing. It provides visibility into which teams, projects or components are compliant with these standards and creates measurable stage gates based on specific criteria regarding defects and testing.

The customizable views in Coverity Policy Manager enable teams to select development metrics and thresholds that align to their objectives, with the flexibility to modify them throughout the course of the project. Managers can monitor and pinpoint areas of risk by drilling down into specific issues. This code intelligence enables better decisions and improves the predictability of releases.

COVERITY® BY SYNOPSYS

www.coverity.com

Coverity Inc. Headquarters
185 Berry Street, Suite 6500
San Francisco, CA 94107 USA

U.S. Sales: (800) 873-8193
International Sales: +1 (415) 321-5237
Email: sales@coverity.com