Achieving Java Application Security
With Parasoft Jtest
Cloud computing continues to gain traction as enterprises increasingly embrace the shift to Internet-based environments. Unfortunately, this also increases exposure to potential cyber-attacks. Industry groups focused on software security (CWE, OWASP, PCI, etc.) have responded to changing conditions by researching, identifying, and documenting software security vulnerabilities. As a result, guidelines, standards, and best practices have emerged. Parasoft Jtest incorporates the industry’s efforts into an efficient solution for reducing vulnerabilities exposed to cyber-attacks. Benefits include increased application security, lower development costs as a result of its efficient design, and cost savings associated with remediating defects early in the development process.

**The New Frontier of Software Security**

Most, if not all, modern enterprises have an Internet presence, which used to mean a Web server and an email server. As more enterprises take to the cloud, “Internet presence” often includes a range of cloud service technologies, e.g. software as a service (SaaS), platform as a service (PaaS), and infrastructure as a service (IaaS). An entire range of cyber attacks is now possible, such as distributed denial-of-service attacks (DDOS) and SQL injections (SQLi). Suddenly the small surface exposed to hackers has become a massive playing field with enterprises storing more and more data in a cloud or cloud-based.

Enterprises can no longer subscribe to the traditional view of cyber security designed to block access to targeted assets. The entire cloud is polluted with bots looking for weak code to exploit. IT professionals must take software security challenges seriously, because computers, servers, network, and mobile devices are constantly exposed to the risk of real cyber-attacks. We can no longer afford to discuss the possibility that sensitive data may be stolen; we must address the probability that data will be stolen.

The challenge of securing company assets and complying with safety regulations and programming best practices continues to evolve. Virtualization, cloud computing, and especially mobile access to the enterprise network, further compounds the probability of a cyber attack. Simply entrusting security to the cloud service, virtualization vendor, or third-party app is only shifting the same set of security problems to another team, which creates an even larger surface and more potentially vulnerable code. Without building security into the application, it becomes the weakest point in your security strategy.

The hunker-down tactics that seek to plug all potential weaknesses isn’t consistent with strategy that is able to address modern security threats. We can no longer presume that security is outside the domain of the application. Focusing on network layers, for example, does nothing to minimize losses that result from the inevitable bot that is able to fool the network into letting it through. The focus must be on minimizing the probability and effects of attacks.

**High Stakes**

There is good reason to be concerned about software security. The costs associated with releasing software to the market that hasn’t met security standards go far beyond developing and deploying patches. In addition to reworking the code, there are costs associated with mitigating the vulnerabilities, liability for losses, and potential legal action.
Industry Initiatives

Work groups, community initiatives, and government agencies have been established (CWE, OWASP, SAMATE, etc.) in response to increasing security risks. Traditional security measures focus on investigating security issues in order to detect and respond to cyber threats. These groups take security a step further and focus on understanding possible vulnerabilities in order to offer guidance for prevention and mitigation. They have identified and documented common programming errors that may expose vulnerabilities to hackers. Year after year, the organizations announce the lists of top security vulnerabilities and programming errors to help educate developers on how to avoid programming errors, as well as develop secure and robust software.

- The Software Assurance Metrics and Tool Evaluation (SAMATE) project at the National Institute of Standards and Technology (NIST) is working to establish a methodology for ensuring that software is free from vulnerabilities and functions as intended.

- CWE™ provides a unified, measurable set of software weaknesses that is enabling more effective discussion, description, and selection of software security tools and services for finding weaknesses in source code and operational systems.

- The Open Web Application Security Project (OWASP) is a non-profit, worldwide charitable organization focused on improving the security of application software. Its mission is to make application security visible, so that people and organizations can make informed decisions about true application security risks.

- Build Security In is a government initiative that provides practices, tools, guidelines, rules, principles, and other resources for software developers, architects, and security practitioners to help them build security into software in every phase of its development.

Java Solution

Developers must be able to participate in application security throughout the entire SDLC. But they need a solution for enforcing security standards that can be implemented into their development process without adding additional layers of workflow. The solution must not only expose defects in the source code but provide actionable information on the potential vulnerabilities, as well as the consequences and suggested remediation. This will enable developers to quickly plug some of the most common security holes that can lead to brand damage and costly security breaches.

The core of Parasoft’s Java solution is Jtest—the most comprehensive testing solution for Java applications—which helps developers quickly remediate security defects by checking code against entire libraries of security standards. Arming developers with ready-to-use coding standards and simple tools for fixing the most commonly exploited security defects in Java reduces the risk of application security attacks.
Jtest is the only solution in the market that enables you to run static code analysis, flow analysis and runtime error detection in a single application. Jtest’s broad set of capabilities is discussed further in the following sections.

How Static Analysis Improves Java Application Security

Jtest’s static analysis engine is based on a variety of secure application development guidelines and compliance reports, such as those described Achieving Application Security with Parasoft Jtest 5 in CWE-SANS, OWASP, and PCI DSS. The extended coding guidelines are available for desktop and for server integration. Jtest checks code against hundreds of security related coding standards as you work, which provides immediate guidance for creating safe code. Potential vulnerabilities are flagged as the code is created—when the cost of remediation is low. Jtest also integrates with continuous integration systems. This enables developers to run computationally intensive analysis to expose additional security-related defects. Results can be reported back to developers and managers through the IDE, by email, or Web.

Jtest offers static analysis for both preventive and detection measures. Flow-based static analysis exposes potential vulnerabilities by analyzing execution paths through the code. Pattern-based static analysis is ideal for educating developers and coaching them in the least-intrusive way, as well as crafting a preventative strategy for building more secure software.

How Runtime Error Protection Adds Application Security

The extended static analysis works in concert with Jtest’s runtime error detection, which identifies security vulnerabilities that occur as the application is exercised during any automated or manual tests. This patent-pending technology is exclusive to Jtest and allows developers to actually monitor application execution and react in real time when a potential threat is detected.

Unit Testing, Peer Review, and More

Jtest’s software development capabilities also include unit testing, coverage analysis, and peer review, which catches 60% of software defects according to software testing experts∗. Jtest’s unique Code Review feature is a critical development capability that improves on standard peer review tools by unobtrusively increasing workflow flexibility with features such as pre-, post-, and hybrid commit review.

Unit testing is a proven, but often skipped, method of finding and fixing defects. This is because properly writing and maintaining tests requires investing time and resources. Jtest turns unit testing from a luxury to a policy with simple test case parameterization, automated test case generation, automated stub generation, and other features. Unit test results, as well as results from static analysis, are available in the developer’s IDE, creating a complete development testing environment. Additionally, the extensive reporting API gives stakeholders clear code coverage insight and correlates results to code requirements.

A Complete Testing Solution

According to Boehm and Basili, “Disciplined personal practices can reduce defect introduction rates by up to 75%”. Jtest eases the transition to a software security-aware development testing workflow. This is because Jtest is designed and built by developers for developers who
understand how disruptive and cumbersome testing can be if incorrectly implemented. At the same time, developers overwhelmingly agree on the importance of following best practices, which is why Jtest is the ideal Java solution for turning security testing from burden to an asset.

Features:

- Broad static analysis techniques: pattern-based analysis, flow analysis, runtime error detection.
- Out-of-box rule sets for industry-recognized coding standards: More than a 100 security related rules are included.
- Reports that can be presented for auditing purposes.
- An efficient peer review framework.
- A convenient unit testing framework.
- And much more . . .

Benefits:

- Meet regulatory compliance and industry standards.
- Automatically find and fix programming errors to prevent vulnerabilities.
- Increase the efficiency of otherwise mundane tasks.
- Help developers focus on functionality that matters for your business.

Visit [www.parasoft.com/jtest](http://www.parasoft.com/jtest) and [www.parasoft.com/jtest-security](http://www.parasoft.com/jtest-security) to learn more about Parasoft Jtest, including how Jtest rules map to the most recognized security vulnerabilities standards.

Security Policy

Most organizations have reduced vulnerabilities through traditional means, such as hardening physical security and addressing network infrastructure. But the rise of cloud-based applications is driving the need for security at the application level. Jtest helps you overcome new security challenges through development testing best practices based on widely-accepted industry security standards. It should be an additional piece of your overall policy for ensuring application security that works alongside the measures already in place.

Summary

Jtest is the market-leading application security testing solution for developing, testing, and maintaining secure applications throughout all phases of the SDLC. Features such as static flow analysis, pattern-based static analysis, runtime error detection, code review, unit testing and more enable you to find and fix vulnerabilities while they’re still in the early stage of the development process. Jtest also helps you prevent vulnerabilities from being implemented in the future development, which reduces the cost of building secure and compliant web and mobile applications.
About Parasoft

For 25 years, Parasoft has researched and developed software solutions that help organizations deliver defect-free software efficiently. By integrating development testing, API/cloud/SOA/composite app testing, dev/test environment management, and software development management, we reduce the time, effort, and cost of delivering secure, reliable, and compliant software. Parasoft's enterprise and embedded development solutions are the industry's most comprehensive—including static analysis, unit testing with requirements traceability, functional & load testing, service virtualization, and more. The majority of Fortune 500 companies rely on Parasoft in order to produce top-quality software consistently and efficiently.

Contacting Parasoft

USA
101 E. Huntington Drive, 2nd Floor
Monrovia, CA 91016
Toll Free: (888) 305-0041
Tel: (626) 305-0041
Fax: (626) 305-3036
Email: info@parasoft.com
URL: www.parasoft.com

Europe
France: Tel: +33 (1) 64 89 26 00
UK: Tel: + 44 (0)208 263 6005
Germany: Tel: +49 731 880309-0
Email: info-europe@parasoft.com

Other Locations
See http://www.parasoft.com/contacts

About the Authors

Marek Kucharski – VP of Software Development, Parasoft
Arthur Hicken – Jtest Product Manager, Parasoft
Adam Trujillo – Technical Writer, Parasoft

© 2012 Parasoft Corporation
All rights reserved. Parasoft and all Parasoft products and services listed within are trademarks or registered trademarks of Parasoft Corporation. All other products, services, and companies are trademarks, registered trademarks, or servicemarks of their respective holders in the US and/or other countries.