IBM Vision for Application Security

Rustin Sides
Federal Security Specialist
Application Vulnerabilities Continue to Dominate

- In 2010, 49% of all vulnerabilities disclosed are Web application vulnerabilities
- SQL injection and Cross-Site Scripting are neck and neck in a race for the top spot
eb Application Architecture – The Potential Weak Spots!!!
Examples of Exploits
SQL Injection Example - Exploit
### SQL Injection Example - Outcome

#### Welcome to Altoro Mutual Online.

**Hello, John Smith**

Welcome to Altoro Mutual Online.

View Account Details: 1001160140 Checking

**Congratulations!**

You have been pre-approved for an Altoro Gold Visa with a credit limit of $10000!

Click [Here](#) to apply.

---

The Altoro Mutual website is published by Watchfire, Inc. for the sole purpose of demonstrating the effectiveness of Watchfire products in detecting web application vulnerabilities and website defects. This site is not a real banking site. Similarities, if any, to third party products and/or websites are purely coincidental. This site is provided "as is" without warranty of any kind, either express or implied. Watchfire does not assume any risk in relation to your use of this website. For additional Terms of Use, please go to [http://www.watchfire.com/statements/terms.aspx](http://www.watchfire.com/statements/terms.aspx).

Copyright © 2007. Watchfire Corporation. All rights reserved.
IBM Solutions for Application Security
IBM Rational AppScan End-to-End Application Security

<table>
<thead>
<tr>
<th>REQUIREMENTS</th>
<th>CODE</th>
<th>BUILD</th>
<th>QA</th>
<th>SECURITY</th>
<th>PRODUCTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Security Requirements Definition</td>
<td>Build security testing into the IDE</td>
<td>Automate Security / Compliance testing in the Build Process</td>
<td>Security / compliance testing incorporated into testing &amp; remediation workflows</td>
<td>Security &amp; Compliance Testing, oversight, control, policy, audits</td>
<td>AppScan onDemand (SaaS)</td>
</tr>
</tbody>
</table>

**AppScan Source**

- **AppScan Enterprise / Reporting Console** (enterprise-wide scanning and reporting)

**Application Security Best Practices**

- Outsourced testing for security audits & production site monitoring
Secure Application Development

AppScan Enterprise / Reporting Console

- AppScan Developer Ed (desktop)
- AppScan Ent. QuickScan (web client)
- AppScan Build Ed (scanning agent)
- (scanning agent) (QA clients)
- AppScan Tester Ed

Rational ClearCase

Rational BuildForge

Rational Quality Manager

AppScan Enterprise user (web client)

AppScan Standard Ed (desktop)

AppScan Express (desktop)

Rational ClearQuest / Defect Management

CODE
Build security testing into the IDE

BUILD
Automate Security / Compliance testing in the Build Process

QA
Security / compliance testing incorporated into testing & remediation workflows

SECURITY
Security & Compliance Testing, oversight, control, policy, audits

IBM Rational Web Based Training for AppScan
Security Testing Within the Software Lifecycle

Most Issues are found by security auditors prior to going live.
Security Testing Within the Software Lifecycle

% of Issue Found by Stage of SDLC

Desired Profile
IBM is Making a Big Investment in Application Security

**Acquisitions:**
- Watchfire acquisition 2007
- Ounce acquisition 2009

**Global R&D Team**
- Hawthorn NY research lab
- Tokyo research lab
- Ottawa development lab
- Toronto development lab
- Boston development lab

**Gartner's take on the Ounce Labs Acquisition.**

Of the major application development platform vendors, IBM made the first move to incorporate security testing into SLC with its acquisition of leading DAST tool vendor Watchfire (as well as a data-masking vendor, Princeton Softech) in 2007. **IBM now extends this leadership in 2009 with its acquisition of a leading SAST tool vendor, Ounce Labs.** SAST and DAST techniques are complementary and shouldn’t have to come from separate vendors, and in the longer term they won’t.

Furthermore, vendors have greater vision if they integrate static and dynamic testing to increase the breadth of application life cycle coverage and the accuracy of vulnerability detection, thus better serving enterprises’ strategic security needs.
Appscan Overview

- Desktop and enterprise scalable solutions for assessing and resolving security vulnerabilities
- Built for security professionals, executives, testers and developers
- Enterprise-wide reporting, tracking and trending
- Broad scanning capabilities, including advanced Web 2.0 technologies
How does Rational AppScan work?

*Automates Application Security Testing*

*Scan applications*

*Analyze (identify issues)*

*Report (detailed & actionable)*

Same process for whitebox & blackbox
Security Testing Technologies... Combination of the Two Delivers Comprehensive Solution

- **Static Code Analysis = Whitebox**
  - Scanning source code for security issues

- **Dynamic Analysis = Blackbox**
  - Performing security analysis of a compiled application

![Diagram showing the combination of Static Analysis and Dynamic Analysis providing Complete Coverage of Total Potential Security Issues](image-url)
Security Analyst - AppScan Standard

![AppScan Standard interface](image)
Security Architect - AppScan Source Edition

IBM Software Group | Rational software

JDBC executeQuery

API: java.sql.Statement.executeUpdate
Type: Injection.SQL
Severity: HIGH
Description: The method java.sql.Statement.executeUpdate() is a common interface to stored procedures, prepared statements, and dynamic SQL statements. When used to execute dynamic SQL statements, executeQuery() is vulnerable to SQL injection attacks. All three SQL statement types may also be vulnerable to other input poisoning attacks depending on the use of the data.

SQL injection is the insertion of malicious SQL commands inside the SQL commands generated by the application. Typically, an attacker uses SQL special characters to prematurely terminate.
Integration with Eclipse and Visual Studio
Flexible Deployment Options

Functions:
- Security Requirements
- Configure
- Scan
- Triage
- Remediate
- Verify

Rational AppScan: - Source for Automation - Standard Ed

Build

Development
Rational AppScan:
- Source Ed Developer
- Source Ed Remediation
- Enterprise QuickScan

QA
Rational AppScan Tester Ed for RQM

Rational AppScan Source Ed Core

Security
Rational AppScan: - Standard Ed - Source Ed for Security

Compliance
Key Strategic Priorities

- Leverage best of both leading approaches - Integration of black and whitebox
- Build security testing into the development process
- Accuracy and simplicity for developers
- Coverage and completeness for security auditors
- Facilitate collaboration between security and development
- Governance, metrics and dashboards
Why Customers Choose IBM Rational Security & Compliance

- Broadest suite of offerings to support security testing across the development lifecycle
- Only web application security testing solution that provides combined static and dynamic analysis
- Integrated with Rational application lifecycle management portfolio allowing security to become a natural part of the software development process
  - Minimize disruption
  - Scale to large number of users
  - Support collaboration within development
  - Integrate with development tools
- R&D backed by IBM’s $1.5B annual investment in security
- Comprehensive Application Security Analysis - Includes multiple analysis techniques to leverage strengths of many solutions
ISS SiteProtector SP8 and AppScan SE Overview
AppScan integration with ISS Site Protector

- This AppScan integration with site protector is a free extension.
- Add Web application security issues to the SiteProtector repository of security issues.
- With this broader view, you get a more complete understanding of your security position, so that you can better understand the risks that your organization faces and set priorities accordingly.
Requirements

Site Protector SP8 or greater version

System Requirements:
http://documents.iss.net/literature/SiteProtector/sp_installation_guide20sp80.pdf#page=11

Database update for Fusion - TBD

AppScan Extension (Free from the Extensions website)

Any GX Appliance (not a G appliance) - virtual or appliance
* See Resources for more information
Features

Security issues Visibility:
SiteProtector Publisher can post all of the security issues displayed in AppScan Standard Edition to Proventia Management SiteProtector Version 2.0, Service Pack 8 (or newer).

Allows Security Risk Control
The scalable design of SiteProtector helps you control, monitor, analyze, and report on your security through a single, centralized console.

Prioritize High Security Risks:
SiteProtector incorporates both information from security scanners and data from real-time security events. This means that information about live attacks can be prioritized and handled within the context of your vulnerabilities. For example, an SQL Injection attack on an asset that is known to be vulnerable to SQL Injection clearly is urgent to address. The SiteProtector SecurityFusion™ module makes it easier to identify such urgent events.
Blind SQL Injection

Severity: High
Type: Application-level test
WASC Threat Classification: Command Execution, SQL Injection
CVE Reference(s): N/A
Security Risk: It is possible to view, modify or delete database entries and tables

Possible Causes
Sanitation of hazardous characters was not performed correctly on user input.

Technical Description
Web applications often use databases at the backend to interact with the enterprise data warehouse. The de-facto standard language for querying databases is SQL (each major database vendor has its own dialect). Web applications often take user input (taken out of the HTTP request) and incorporate it in an SQL query, which is then sent to the backend database. The query results are then processed by the application and sometimes displayed to the user.

This mode of operation can be exploited by an attacker if the application is not careful enough with its treatment of user (attacker) input. If this is the case, an attacker can inject malicious data, which when incorporated into an SQL query, changes the original syntax of the query into something completely different. For example, if an application uses user's input (such as username and password) to query a database table of users' accounts in order to authenticate the user, and the attacker has the ability to inject malicious data into the username part of the query (or the password part, or both), the query can be changed into a different data-yielding query, a query that modifies the database, or a query that runs shell commands on the database server.

Typically, the attacker achieves this goal in steps. Firstly, the attacker will learn the structure of the SQL query, and then use this knowledge to thwart the query (by injecting data that changes the query syntax) into performing differently than intended. Suppose the query in question is:

```
SELECT COUNT(*) FROM accounts WHERE username='User' AND password='Password'
```

Where User and Password are user input (collected from the HTTP request which invoked the script that constructs the query - either from a GET request query parameters, or from a POST request body parameters). A regular usage of this query would be with values `User: john; Password: secret123`. The query formed would be:

```
SELECT COUNT(*) FROM accounts WHERE username='john' AND password='secret123'
```

The expected query result is 0 if no such user-password pair exists in the database, and >0 if such pair exists (i.e., there is a user named John in the database, whose password is 'secret123'). This would serve as a basic authentication mechanism for the application. But an attacker can alter this query in the following way:

By providing an input consisting of a single apostrophe character ('), the attacker can cause the database to emit an error message, which usually contains valuable information regarding the SQL query. The attack would simply involve sending a request with the user value ' and a password with any value (e.g., foobar). The result would be the following (malformed) SQL query:
The extension will display the publishing results during the upload. Close the extension when the publishing is complete.
We now see a list of vulnerability categories. Expand these to view the vulnerable URLs and parameters within those categories.
Here you can see that it found cross-site scripting on the TXTSEARCH parameter on the search parameter. Right click for detail.
Click on Event Details
HTTP Traffic
Resources

AppScan Extension

Site Protector System Requirements:
http://documents.iss.net/literature/SiteProtector/sp_installation_guide20sp80.pdf#page=11

SiteProtector 2.0 SP8
http://iss.net
  Create an account
  Login
  Choose to go to the SiteProtector 2.0 SP8 Downloads
  Click on All Updates tab
  Download SiteProtector Service Pack 8.0 Express Install
Resources

IPS Web Application Security Guide – How to get the IPS pieces together
http://documents.iss.net/literature/proventia/IBM_WAS.CG_1.0.pdf

Known issues: