



IBM Rational Software Development Conference 2008

WHERE TEAMS ARE **R-HEROES**



Enabling Security Testing
across the Software Development Lifecycle
with IBM® Rational® AppScan Enterprise Edition

Kedar Walimbe.

Technical Sales Manager ,IBM Rational
kedar@sg.ibm.com

AS07

Security is Quality



The Myth: “Our Site is Safe”



**We Have Firewalls
in Place**

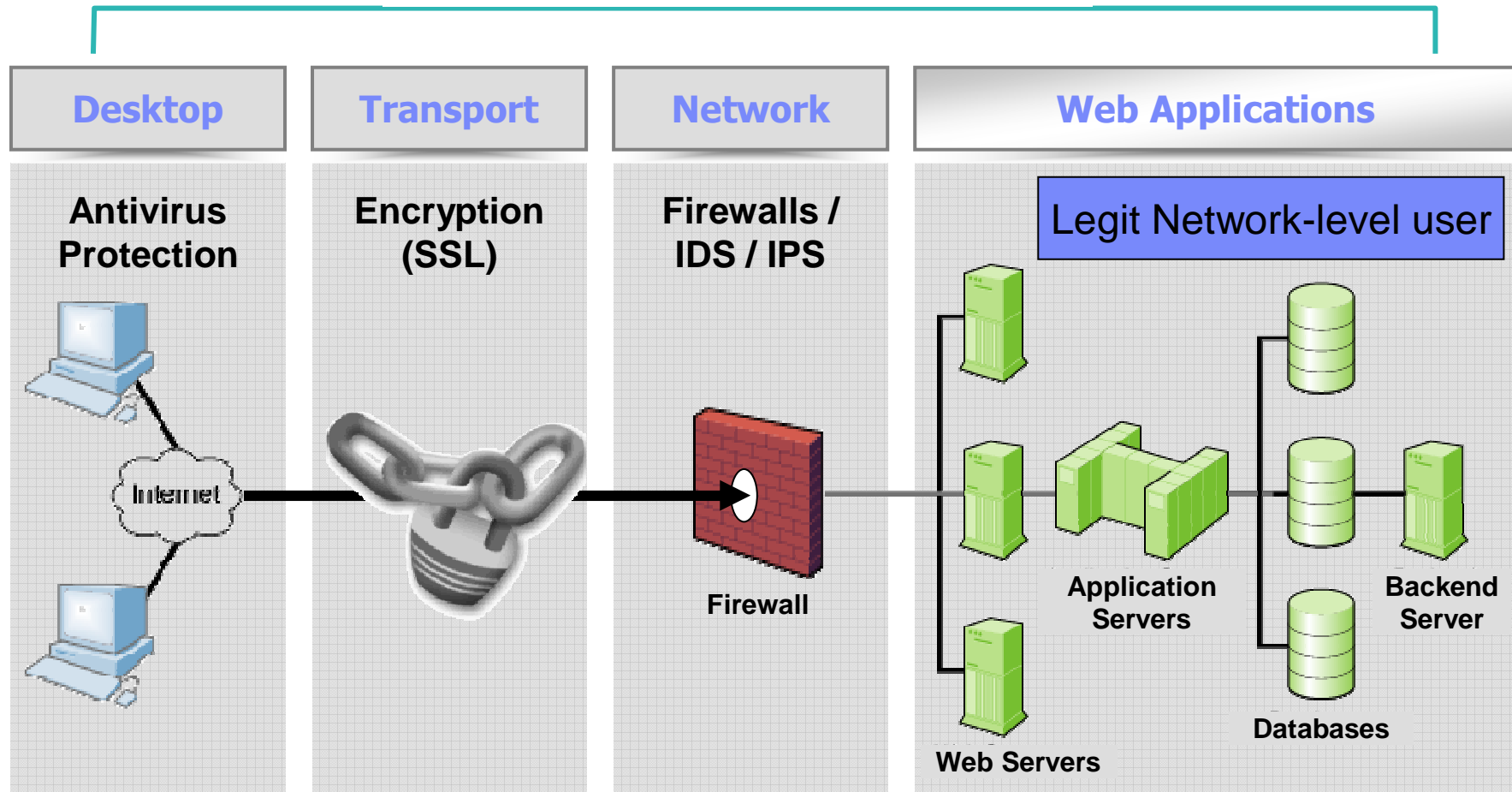
**We Audit It Once a
Quarter with Pen Testers**

**We Use Network
Vulnerability Scanners**



Each layer of the application requires its own security measures

Info Security Landscape



Web application security defects are common and serious

Growing Threat

- Past customer spending focused on Network security – yet 75% of attacks come through web applications – market is now focusing on spending on web application security
- Mitre group indicates that application issues (XSS and SQL Injection) are the top 2 hacks
- Most websites are vulnerable (Watchfire/Gartner)

Analyst Views

“Gartner estimates that **90 percent of externally-accessible applications today are web-enabled, and that two-thirds of them have exploitable vulnerabilities.**”

“**64% of developers are not confident in their ability to write secure applications**”

Microsoft Developer Research

Cost of Application Security Breach

- **Security Breach**
 - Every lost record costs \$138 to the organization who lost it
 - Media Attention > Brand Damage > Sharp Decline in Stock Prices



Regulatory requirements in many industries require you to develop and test to ensure system security



Build and Maintain a Secure Network

- Requirement 1: Install and maintain a firewall configuration to protect cardholder data
- Requirement 2: Do not use vendor-supplied defaults for system passwords and other security parameters

Protect Cardholder Data

- Requirement 3: Protect stored cardholder data
- Requirement 4: Encrypt transmission of cardholder data across open, public networks

Maintain a Vulnerability Management Program

- Requirement 5: Use and regularly update anti-virus software
- Requirement 6: Develop and maintain secure systems and applications

Implement Strong Access Control Measures

- Requirement 7: Restrict access to cardholder data by business need-to-know
- Requirement 8: Assign a unique ID to each person with computer access
- Requirement 9: Restrict physical access to cardholder data

Regularly Monitor and Test Networks

- Requirement 10: Track and monitor all access to network resources and cardholder data
- Requirement 11: Regularly test security systems and processes

Maintain an Information Security Policy

- Requirement 12: Maintain a policy that addresses information security



There are several types of web application security defects

Application Threat	Negative Impact	Example Impact
Cross Site scripting	Identity Theft, Sensitive Information Leakage, ...	Hackers can impersonate legitimate users, and control their accounts.
Injection Flaws	Attacker can manipulate queries to the DB / LDAP / Other system	Hackers can access backend database information, alter it or steal it.
Malicious File Execution	Execute shell commands on server, up to full control	Site modified to transfer all interactions to the hacker.
Insecure Direct Object Reference	Attacker can access sensitive files and resources	Web application returns contents of sensitive file (instead of harmless one)
Cross-Site Request Forgery	Attacker can invoke "blind" actions on web applications, impersonating as a trusted user	Blind requests to bank account transfer money to hacker
Information Leakage and Improper Error Handling	Attackers can gain detailed system information	Malicious system reconnaissance may assist in developing further attacks
Broken Authentication & Session Management	Session tokens not guarded or invalidated properly	Hacker can "force" session token on victim; session tokens can be stolen after logout
Insecure Cryptographic Storage	Weak encryption techniques may lead to broken encryption	Confidential information (SSN, Credit Cards) can be decrypted by malicious users
Insecure Communications	Sensitive info sent unencrypted over insecure channel	Unencrypted credentials "sniffed" and used by hacker to impersonate user
Failure to Restrict URL Access	Hacker can access unauthorized resources	Hacker can forcefully browse and access a page past the login page



For example, Injection Flaws are an important type of security defect that result from flaws in the application

- What is it?
 - ▶ User-supplied data is sent to an interpreter as part of a command, query or data.

- What are the implications?
 - ▶ SQL Injection – Access/modify data in DB
 - ▶ SSI Injection – Execute commands on server and access sensitive data
 - ▶ LDAP Injection – Bypass authentication





ONLINE BANKING LOGIN PERSONAL SMALL BUSINESS INSIDE ALTORO MUTUAL

- PERSONAL**
- [Deposit Product](#)
 - [Checking](#)
 - [Loan Products](#)
 - [Cards](#)
 - [Investments & Insurance](#)
 - [Other Services](#)
- SMALL BUSINESS**
- [Deposit Products](#)
 - [Lending Services](#)
 - [Cards](#)
 - [Insurance](#)
 - [Retirement](#)
 - [Other Services](#)
- INSIDE ALTORO MUTUAL**
- [About Us](#)
 - [Contact Us](#)
 - [Locations](#)
 - [Investor Relations](#)
 - [Press Room](#)
 - [Careers](#)

Online Banking Login

Username:

Password:

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>.

Copyright © 2006, Watchfire Corporation, All rights reserved.



MY ACCOUNT **PERSONAL** **SMALL BUSINESS** **INSIDE ALTORO MUTUAL**

- I WANT TO ...**
- [View Account Summary](#)
 - [View Recent Transactions](#)
 - [Transfer Funds](#)
 - [Search News Articles](#)
 - [Customize Site Language](#)

Hello, John Smith

Welcome to Altoro Mutual Online.

View Account Details:

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>.

Copyright © 2006, Watchfire Corporation, All rights reserved.



MY ACCOUNT **PERSONAL** **SMALL BUSINESS** **INSIDE ALTORO MUTUAL**

- I WANT TO ...**
- [View Account Summary](#)
 - [View Recent Transactions](#)
 - [Transfer Funds](#)
 - [Search News Articles](#)
 - [Customize Site Language](#)

Recent Transactions

After Before

mm/dd/yyyy mm/dd/yyyy

TransactionID	AccountId	Description	Amount
20	1001160140	Rent	1100
21	1001160140	Deposit	1050.88
22	1001160140	Deposit	1050.88
23	1001160140	Car Payment	389.12
24	1001160140	Deposit	1050.88
27	1001160140	Car Payment	389.12
68	1001160141	Deposit	877.8
74	1001160141	Deposit	878.9
77	1001160141	Deposit	881.1
1			

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>.

Copyright © 2006, Watchfire Corporation, All rights reserved.

Altoro Mutual: Recent Transactions - Windows Internet Explorer

http://altoro.testfire.net/bank/transaction.aspx

Sign Off | Contact Us | Feedback | Search Go

AltoroMutual

01/01/2006 union select userid,null,username+', '+password,null from users--

MY ACCOUNT

I WANT TO ...

- View Account Summary
- View Recent Transactions
- Transfer Funds
- Search News Articles
- Customize Site Language

PERSONAL

Recent Transactions

After Before Submit

mm/dd/yyyy *mm/dd/yyyy*

TransactionID	AccountId	Description	Amount
20	1001160140	Rent	1100
21	1001160140	Deposit	1050.88
22	1001160140	Deposit	1050.88
23	1001160140	Car Payment	389.12
24	1001160140	Deposit	1050.88
27	1001160140	Car Payment	389.12
68	1001160141	Deposit	877.8
74	1001160141	Deposit	878.9
77	1001160141	Deposit	881.1
1			

INSIDE ALTORO MUTUAL

Privacy Policy | Security Statement | © 2006 Altoro Mutual, Inc.

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>.

Copyright © 2006, Watchfire Corporation, All rights reserved.

Internet 100%

22	1001160140	Deposit	1050.88
23	1001160140	Car Payment	389.12
24	1001160140	Deposit	1050.88
27	1001160140	Car Payment	389.12
68	1001160141	Deposit	877.8
74	1001160141	Deposit	878.9
77	1001160141	Deposit	881.1
265	1003160121	Deposit	150000
357	1005160101		878.85336
363	1005160101		879.95468
366	1005160101		882.15732
378	1006160141		878.85336
384	1006160141		879.95468
387	1006160141		882.15732
419	1006160141		150180
100116014		jsmith,Demo1234	
100216018		sspeed,Demo1234	
100316012		tuser,tuser	
100416016		admin,admin	
100516010		sjoe,Frazier	
100616014		cclay,Ali	
1			



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>.

Copyright © 2006, Watchfire Corporation, All rights reserved.



ONLINE BANKING LOGIN **PERSONAL** **SMALL BUSINESS** **INSIDE ALTORO MUTUAL**

- PERSONAL**
- [Deposit Product](#)
 - [Checking](#)
 - [Loan Products](#)
 - [Cards](#)
 - [Investments & Insurance](#)
 - [Other Services](#)
- SMALL BUSINESS**
- [Deposit Products](#)
 - [Lending Services](#)
 - [Cards](#)
 - [Insurance](#)
 - [Retirement](#)
 - [Other Services](#)
- INSIDE ALTORO MUTUAL**
- [About Us](#)
 - [Contact Us](#)
 - [Locations](#)
 - [Investor Relations](#)
 - [Press Room](#)
 - [Careers](#)

Online Banking Login

Username:

Password:

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>.

Copyright © 2006, Watchfire Corporation, All rights reserved.



MY ACCOUNT **PERSONAL** **SMALL BUSINESS** **INSIDE ALTORO MUTUAL**

- I WANT TO ...**
- [View Account Summary](#)
 - [View Recent Transactions](#)
 - [Transfer Funds](#)
 - [Search News Articles](#)
 - [Customize Site Language](#)

Hello, Smoking Joe

Welcome to Altoro Mutual Online.

View Account Details:

Congratulations!

You have been pre-approved for an Altoro Platinum Visa with a credit limit of \$12000!

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>.

Copyright © 2006, Watchfire Corporation, All rights reserved.

A bug in the web application code causes this SQL Injection security defect

```
string sAfter = Request.Form["after"];  
string sBefore = Request.Form["before"];
```

Evil input comes in

```
string sSQL = "SELECT t.transid, t.accountid, t.description, t.amount  
FROM transactions t  
INNER JOIN accounts a ON t.accountid = a.accountid  
where t.trans_date >= " + sAfter + " and t.trans_date <= " + sBefore;
```

```
myTransactions = new OleDbDataAdapter(sSQL, myConnection);
```

Evil input gets concatenated into SQL Statement

SQL statement containing evil input gets executed. The result may not be what the developer intended.



The security defects we are talking about are bugs in the application itself

- What causes a security defect?
 - ▶ A coding problem in the application
- How do you fix a security defect?
 - ▶ Need to fix the bug
- Why are security defects so prevalent?
 - ▶ Human Gap: bugs happen
 - ▶ Knowledge Gap: many developers are just getting up to speed on security
 - ▶ Process Gap: security hasn't been part of the development process
- Why wouldn't we apply our best Quality Management practices to security defects?



Security Auditors and Quality Assurance Specialists have complimentary skills and responsibilities



- Knows security in-depth
- Knows corporate and industry standards
- Can exploit security defects to prove impact
- Is responsible for the security of application

- Makes testing repeatable
- Reports on test coverage, release readiness
- Triage and manages defects
- Scales testing effort across a large team
- Already part of the development process

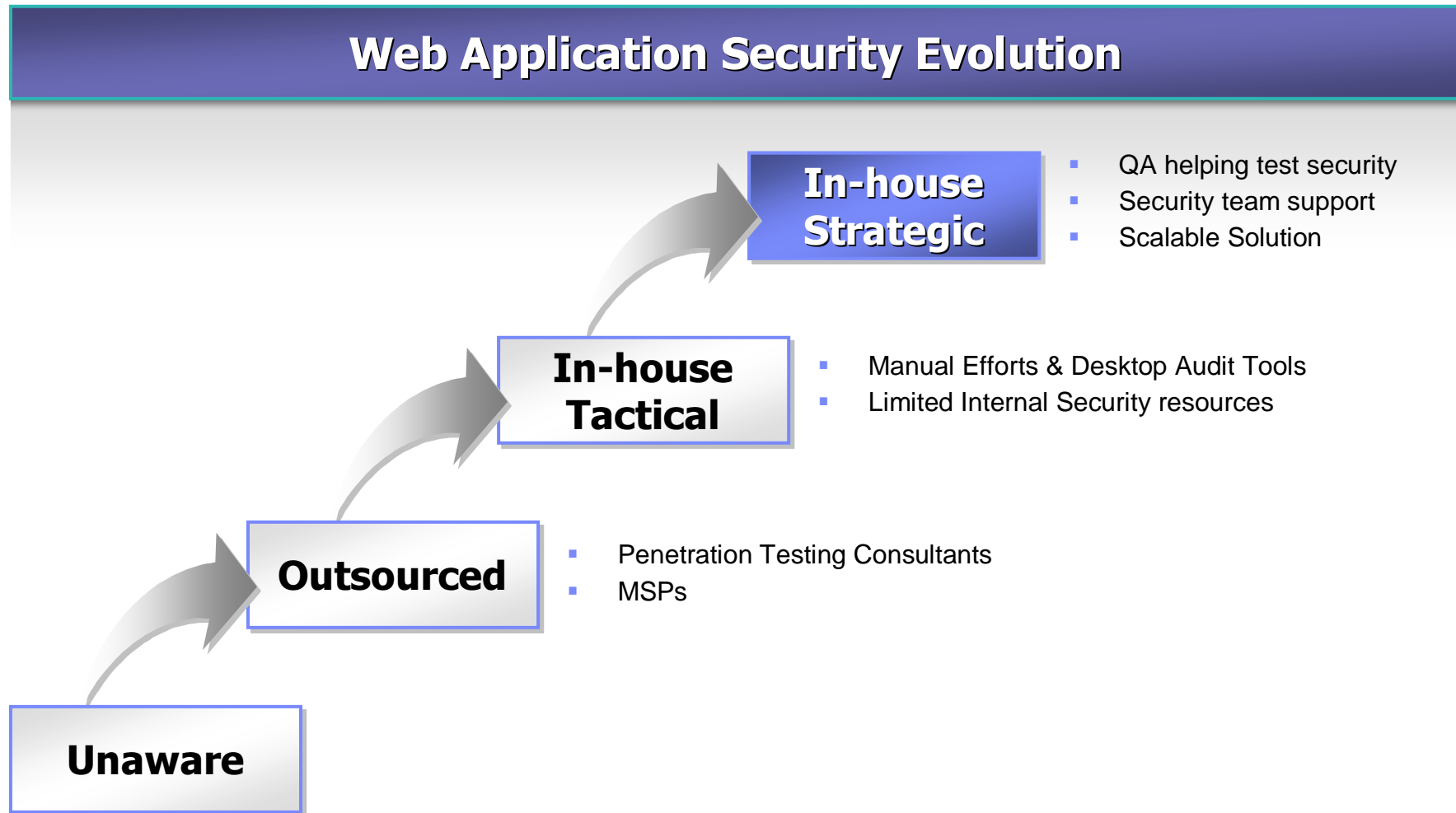


Web application security testing is about finding security defects, but it is also important to understand the issues and how they are fixed

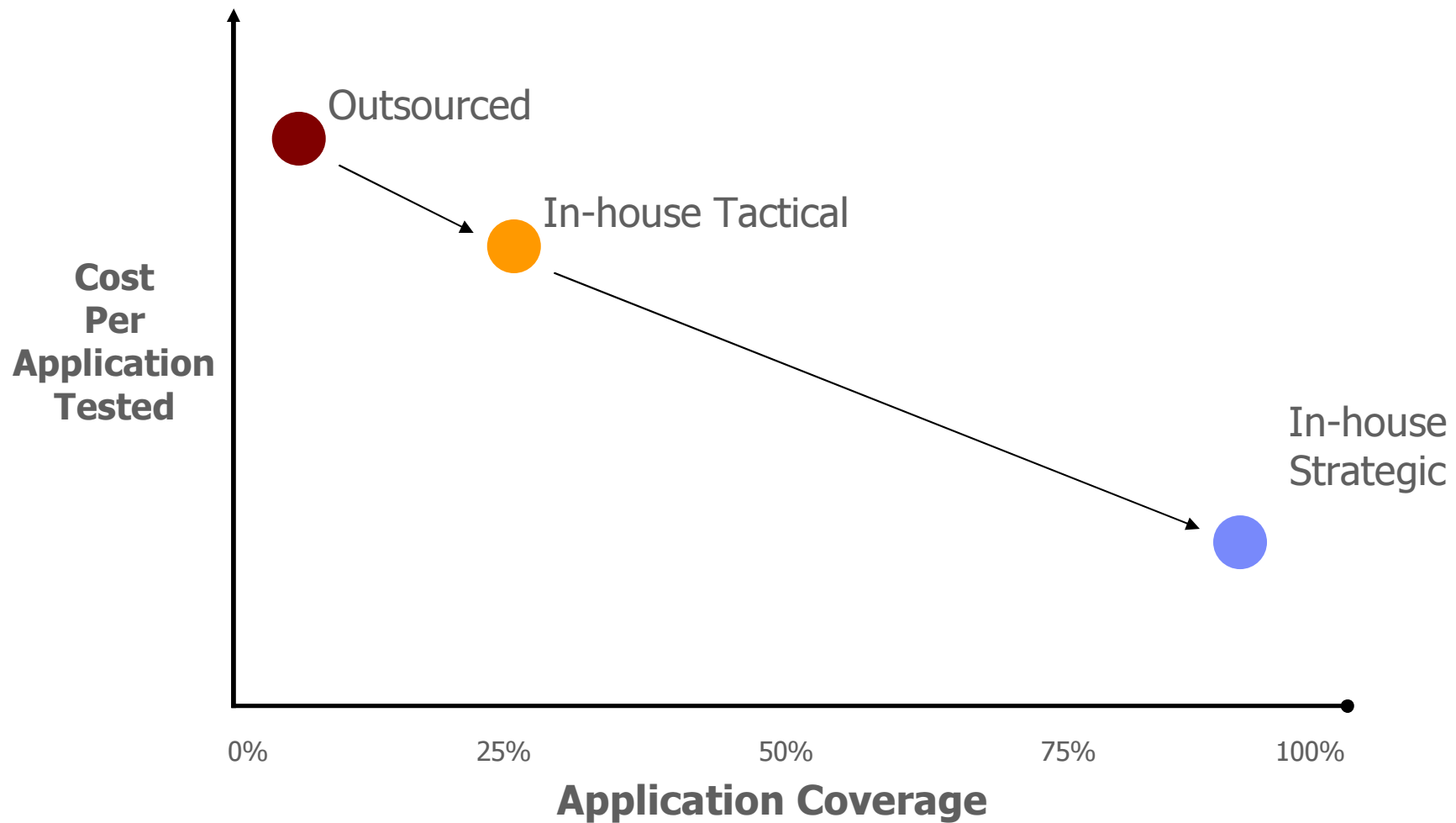
- Web Application Security (WAS) testing is the process of:
 - ▶ Identifying how a web application is vulnerable to being hacked, and
 - ▶ Providing fix recommendations to remediate the security issues



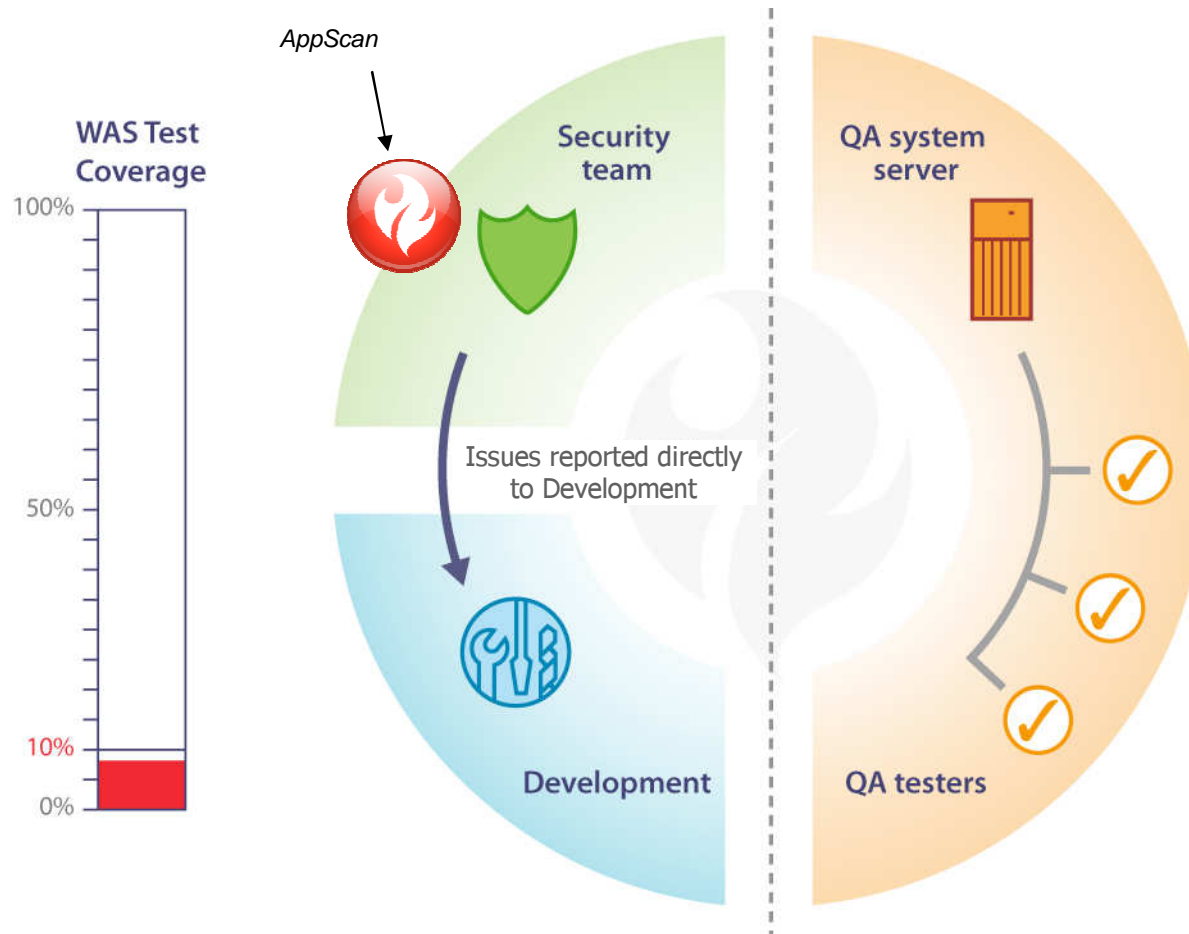
Many organizations move through a maturity model as they adopt web application security testing



The goal is to reduce cost of testing per application so that you can increase test coverage

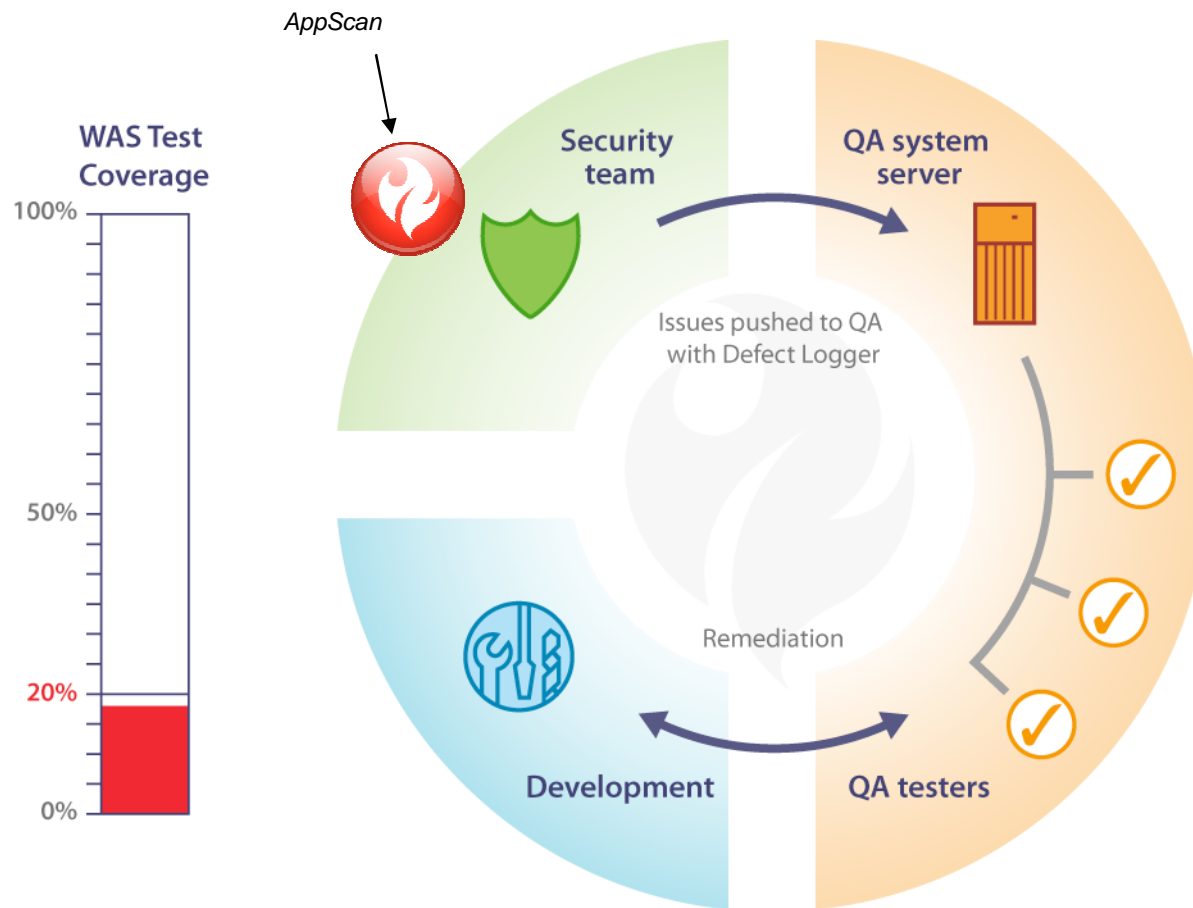


At first, there is little or no QA involvement in security testing and little test coverage is achieved



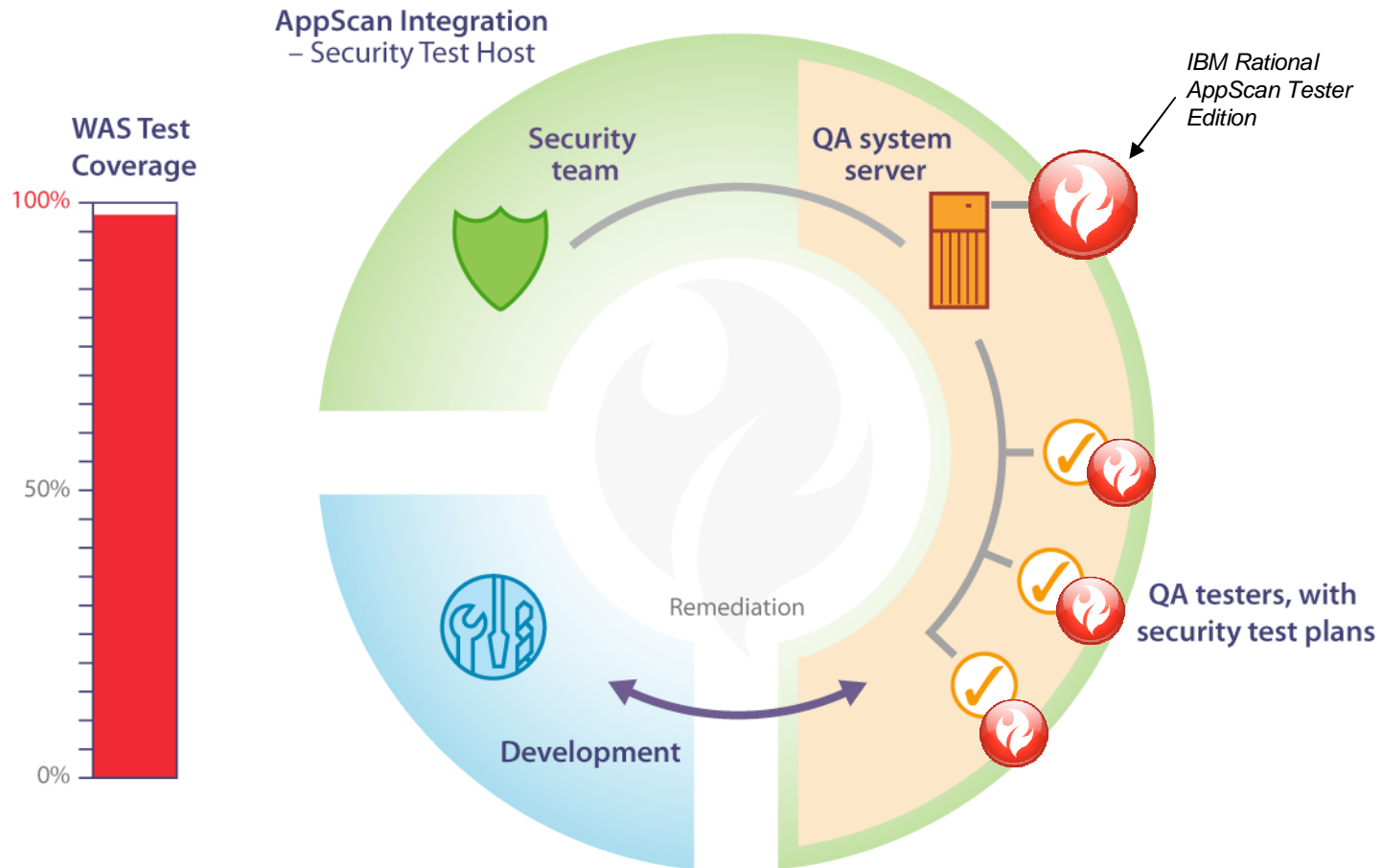
- No communication between Security & QA teams

Later, QA becomes more involved in security testing and test coverage increases



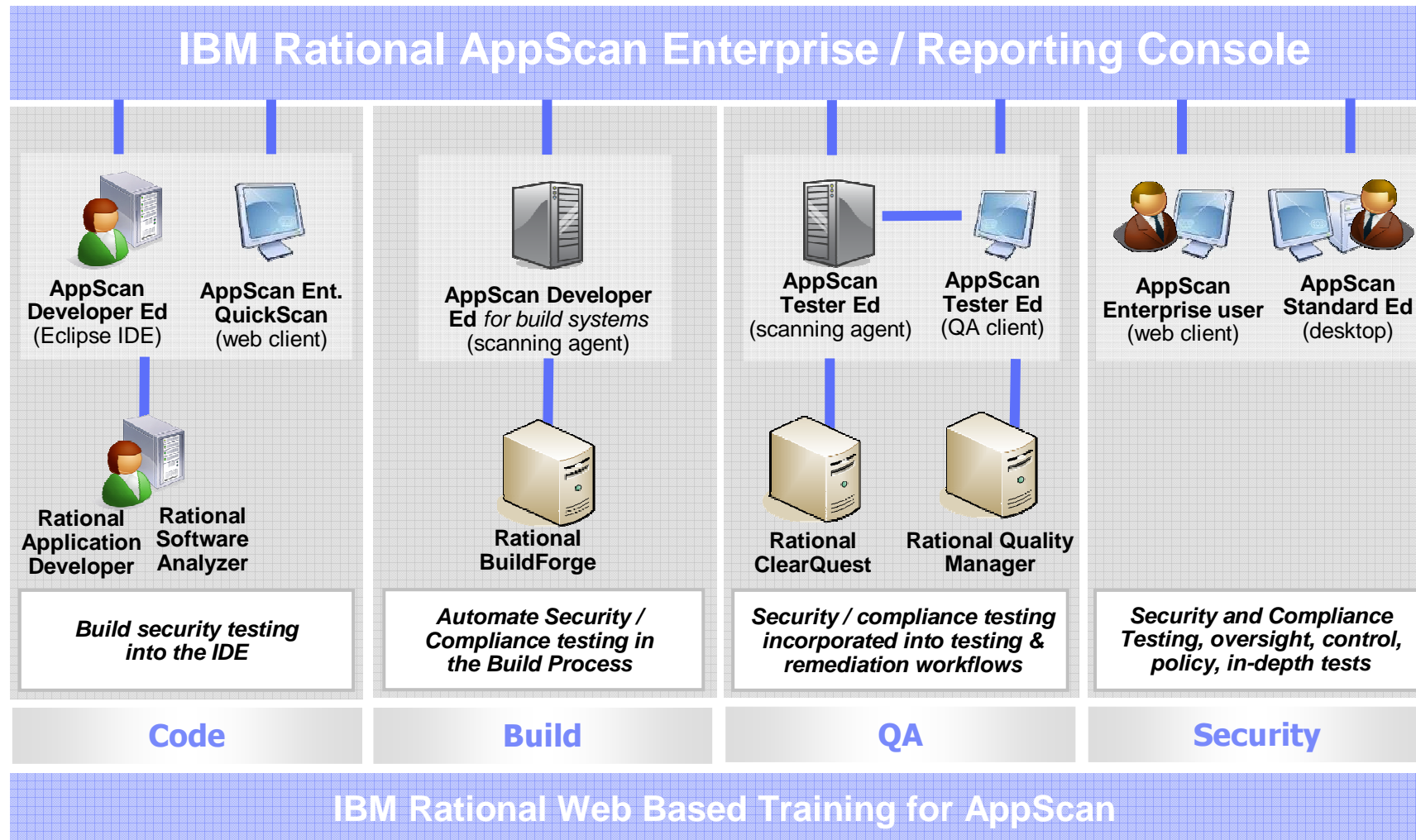
- QA Introduced to WAS; process not yet formalized

Finally, QA is fully engaged in security testing and test coverage approaches 100%



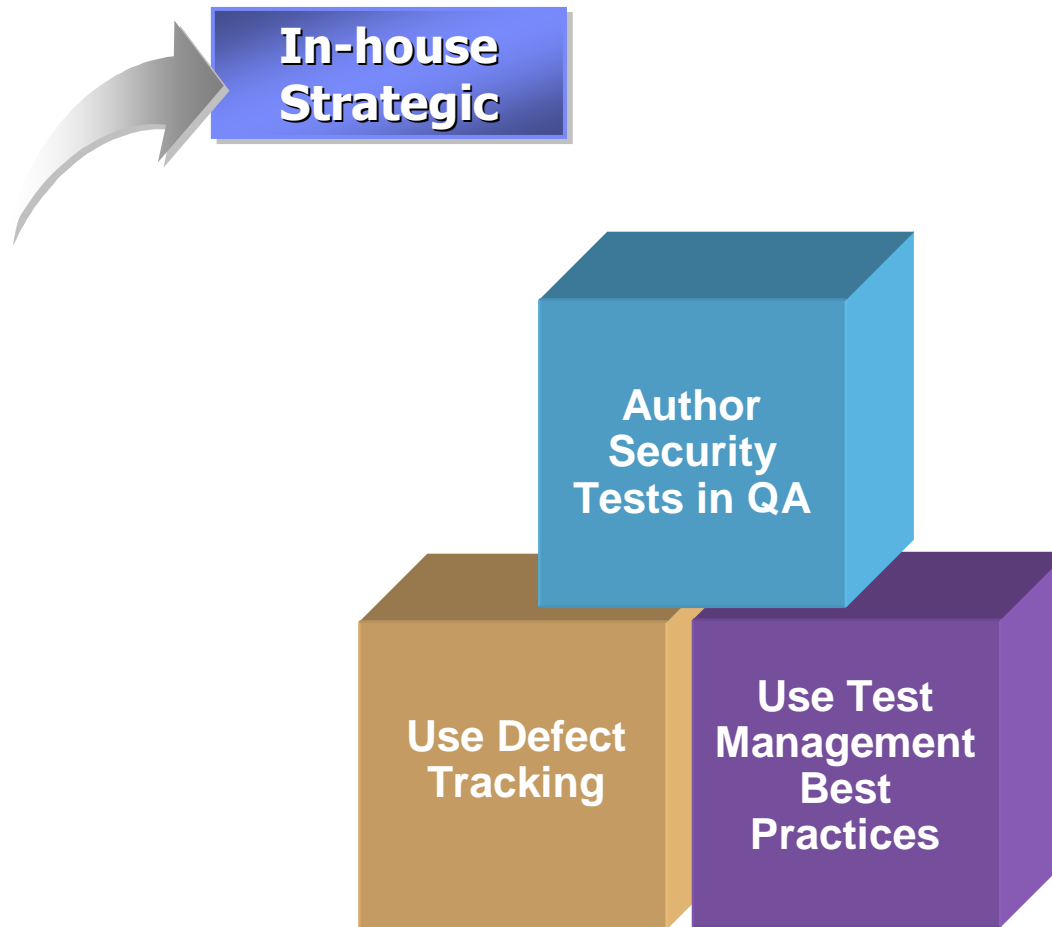
- QA responsible for WAS

Security needs attention across the software development lifecycle



Phased Adoption of Security Testing in QA

Engaging your QA team in security testing, one step at a time



Using Rational AppScan Standard Edition with ClearQuest



The screenshot displays the Watchfire AppScan interface. On the left, a tree view shows the application structure under 'My Application (54)'. The main pane shows a list of security issues, with 'Cross-Site Scripting' selected. A context menu is open over this issue, with 'Log Defect to ClearQuest' highlighted. An arrow points from this menu item to the 'Defect Details' dialog box. The dialog box contains fields for 'Credentials' (Username: admin, Password: [redacted]), 'Defect Details' (Summary: SQL Injection in http://revelation/acmehackme/bank/login.aspx (Parameter passwd)), and 'Priority' (set to '1-Critical'). The 'Description' field contains details about the SQL Injection vulnerability. At the bottom, there are 'Attachments' and 'Log Defect' buttons.



Using Rational AppScan Tester Edition with Rational Team Concert



Rational AppScan Tester Edition

Welcome **Craig Conboy** Admin

Security Issues

Last Updated: 5/12/2008 10:33:51 AM

Summary Group Show Search Layout

There are **68** issues of 24 different types across 21 URLs

All items

Items 1-25 of 68

Action: Submit Rational Quality Manager Defect

Go to page: 1 of 3

		Status	Issue	Work It	Test URL	Element	Issue Type	Threat Class	Last Updated
<input type="checkbox"/>	!	Open	299*		http://qadcore1.otta	before	Blind SQL Injection	Command Execution:	5/12/2008 2:33:00 P
<input type="checkbox"/>	!	Open	300*		http://qadcore1.otta	uid	Blind SQL Injection	Command Execution:	5/12/2008 2:33:00 P
<input checked="" type="checkbox"/>	!	Open	301*		http://qadcore1.otta	passw	Blind SQL Injection	Command Execution:	5/12/2008 2:33:00 P
<input type="checkbox"/>	!	Open	305*		http://qadcore1.otta	_ct0%3A_ct0'	Cross-Site Scripting	Client-side Attacks: (5/12/2008 2:33:00 P
<input type="checkbox"/>	!	Open	306*		http://qadcore1.otta	creditAccount	Cross-Site Scripting	Client-side Attacks: (5/12/2008 2:33:00 P
<input type="checkbox"/>	!	Open	307*		http://qadcore1.otta	debitAccount	Cross-Site Scripting	Client-side Attacks: (5/12/2008 2:33:00 P
<input type="checkbox"/>	!	Open	308*		http://qadcore1.otta	txtSearch	Cross-Site Scripting	Client-side Attacks: (5/12/2008 2:33:00 P
<input type="checkbox"/>	!	Open	309*		http://qadcore1.otta	lang	Cross-Site Scripting	Client-side Attacks: (5/12/2008 2:33:00 P
<input type="checkbox"/>	!	Open	310*		http://qadcore1.otta	uid	Cross-Site Scripting	Client-side Attacks: (5/12/2008 2:33:00 P





Using Rational AppScan Developer Edition with ClearQuest

The screenshot displays the Rational AppScan Developer Edition interface. On the left, the Project Explorer shows a project named 'MyScan' with various files and folders. The main window displays a 'Security Report' for 'MyScan - 07-05-08 14_19_35 PM.srpt'. The report lists several issues, including 'Cross-Site Scripting (1)' and 'Link Injection (Facilitates Cross-Site Scripting)'. A 'Details' pane for the selected issue shows a 'Cross-Site Scripting' warning with severity '2-Major' and a WASC Threat Classification of 'C'. A 'Possible Causes' section indicates that 'Sanitization of hazardous characters was not performed correctly on user input'. Overlaid on the report is a 'Create (Defect) SAMPL00000076' dialog box. The dialog box has tabs for 'Main', 'Attachments', and 'Customer'. The 'Main' tab is active, showing fields for ID (SAMPL00000076), State (Submitted), Headline (Cross-Site Scripting : http://localhost:8080/tomchat/login.jsp), Project, Severity (2-Major), Priority, Owner, Keywords, and Symptoms. The Description field contains the text 'See attached Issue.html for more information.' and there is a 'Template' dropdown menu at the bottom.



Detailed information about the security issue provides QA Managers with the information needed to triage defects

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

The screenshot shows a web application interface with a blue header containing the IBM logo and "IBM Software Group". The main content area is white and displays "SQL Injection". A video player interface is overlaid on the bottom right of the screenshot, showing a play button and other controls. The Rational logo is visible in the bottom left corner of the screenshot.





The security community has methodologies you can use to assign a severity rating to an issue

Score	0 - 2	3 - 4	5 - 6	7 - 8	9 - 10
D amage Potential	Trivial information about the target disclosed. Trivial cost associated with impact	Significant information about the target architecture and/or application disclosed. Limited cost associated with impact	Extended or increased functional control of the application and/or underlying system. Moderate cost associated with impact	Full control of the application and/or the ability to view underlying network or database infrastructure. Large cost associated with impact	Full compromise of Network or Database Infrastructure. Extensive cost associated with impact
R eproducibility	Very difficult to reproduce (more than 24 hours)	Difficult to reproduce (within 24 hours)	Moderately difficult to reproduce (within 2 hours)	Easy to reproduce (within 5 minutes)	Very easy to reproduce (30 seconds or less)
E xploitability	Seasoned security skills and/or specialised tools required	Extensive skills and tools required	Moderate skills and tools required	limited skills and tools required	no skill or tools required
A ffected Users	Very small limited user group (under 100)	Small user group (100 - 1,000)	Moderate user group (between 1,000 - 5,000)	Large user group, Open to the entire Company Network (between 5,000 - 20,000)	Open to the general internet with no authentication or very large group requiring authentication (20,000++)
D iscoverability	Very difficult to find (over 24 hours)	Difficult to find (within 24 hours)	Moderate effort required to find (within 4 hours)	Easily found (within 2 hour)	Very easily found (within 1 hour)

$$\text{Risk Rating} = (D + R + E + A + D) / 5$$

Risk Rating	Threat
0.1 - 4.0	Low Risk
4.1 - 8.0	Medium Risk
8.1 - 10.0	High Risk





Manage your security testing like other types of testing

- QA teams know how to manage testing
 - ▶ What are we going to test?
 - ▶ How are we going to test it?
 - ▶ Who is going to do the work?
 - ▶ How frequently are we going to retest?
 - ▶ What hardware and software are required for the test?
 - ▶ How much of the application has been tested?

- Test Plan, Test Cases, Test Scripts
 - ▶ Include security tests
 - ▶ Monitor and report on test coverage



Manage security test as you manage other tests

Use Test Mgmt Best Practices

The screenshot shows the Rational Quality Manager interface. The main content area displays the 'Java PetStore Test Plan' in a 'Draft' state. Below the plan title, there is a 'Test Cases' section with a table listing associated test cases. The table has columns for Id, Name, State, Theme, Category, Function, Weight, and Modified. Two test cases are visible: 'Shopping Cart Security Tests' and 'Rich Client UI Execution Test 8'.

Id	Name	State	Theme	Category	Function	Weight	Modified
	Shopping Cart Security Tests	Draft	Security	Security	Editors	90	1 minute ago
8	Rich Client UI Execution Test 8	Draft	Functionality	Rich Client UI	Execution	55	5 hours ago



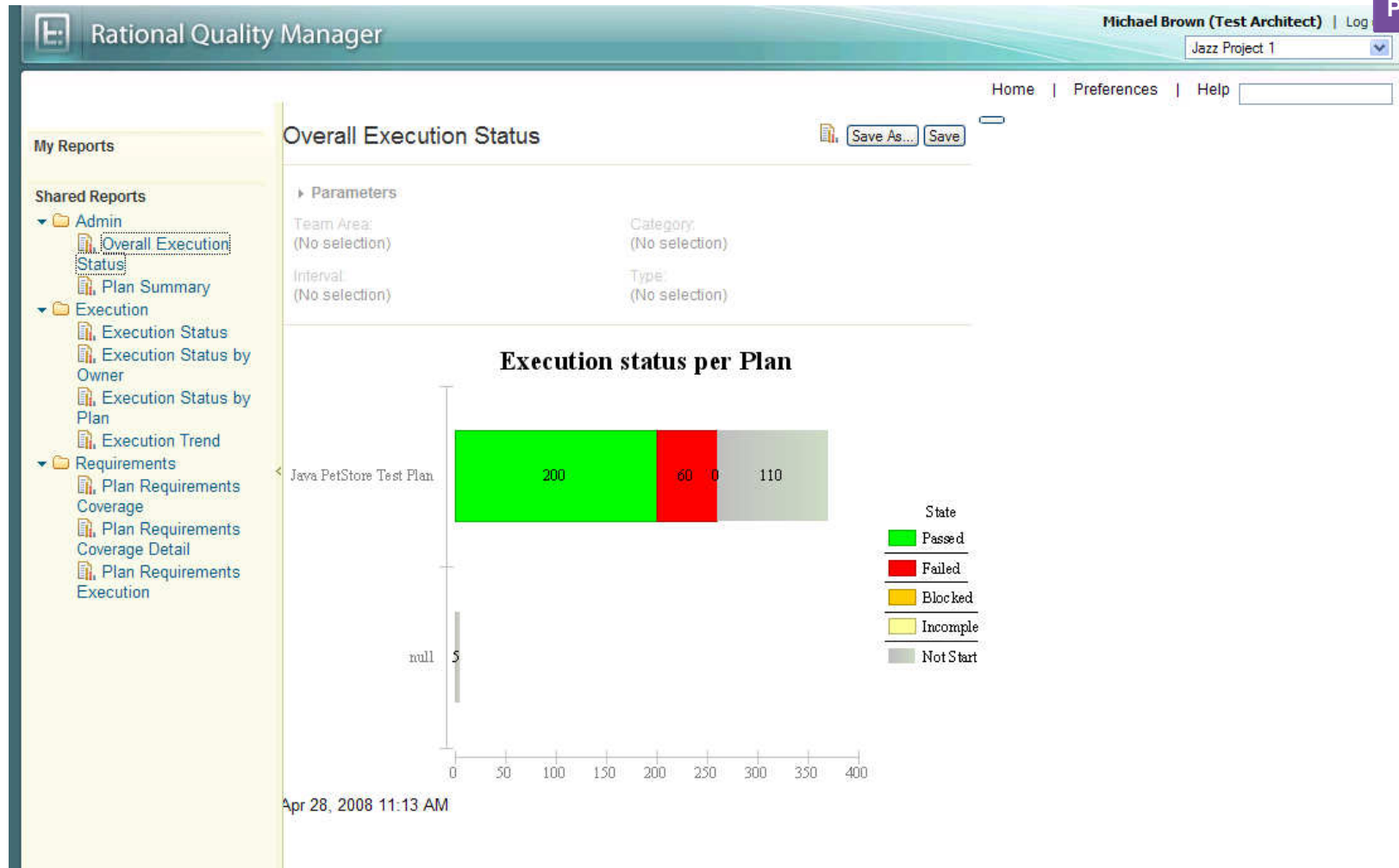


Execute your security tests



Report on test execution status and trend

Use Test Mgmt Best Practices





Enable your testers to create security tests

- Training
- Templates
 - ▶ Test policy
 - ▶ Scan configuration
- Record tests
- Advanced stuff later



Author
Security
Tests in
QA

Define Test Policies for all testers to use

The screenshot shows the Rational Quality Manager web interface. The main content area is titled "Java PetStore Test Plan" and is in a "Draft" state. A section titled "Application Security" is highlighted, showing a table of available test policies:

Name	Description
Developer_Essentials	This policy includes a selection of Application tests that have a high probability of success. This can be useful for developers who wish to quickly evaluate their application.
Invasive	This policy includes all invasive tests (tests which might affect the server's stability).
The_Vital_Few	This policy includes a selection of tests that have a high probability of success. This can be useful for evaluating a site when time is limited.
Default	This policy includes all tests except invasive and port listener tests.





Create AppScan Tester Edition tests from Rational Quality Manager

The screenshot shows the Rational Quality Manager web interface. At the top, it displays the user 'ADMIN' and a 'Log Out' link. Below the header, there's a search bar and navigation links for 'Home', 'Preferences', 'Help', and 'About'. The main content area is titled 'Shopping Cart Automated Security Test' and is identified as a 'Test Script'. It shows the test is 'New / Not Yet Saved' and provides 'Discard Changes' and 'Save' buttons. The configuration fields include:

- Originator: ADMIN
- Owner: Unassigned (dropdown)
- Type: Rational AppScan (dropdown)
- Description: Tests the shopping cart functionality of the application for security defects

 Below these fields, there's a section for 'Rational AppScan Tester Edition' with a note: 'Rational AppScan Tester Edition will scan your web application for security vulnerabilities.' Under the 'Template' section, it instructs to select a template type, with 'Manual Explore' selected. The 'Verdict Strategy' section explains that it determines pass/fail criteria and shows radio buttons for severity thresholds: High (selected), Medium, Low, and Information.





Provide a template for configuring the test to make it easy

- QuickScan UI simplifies test creation



Summary and Call To Action

- Security is Important
- Security is Quality
- Security is Testable; you can make a difference
- Make application security part of what your QA team does!





QUESTIONS





THANK YOU

Learn more at:

- [IBM Rational software](#)
- [IBM Rational Software Delivery Platform](#)
- [Process and portfolio management](#)
- [Change and release management](#)
- [Quality management](#)
- [Architecture management](#)
- [Rational trial downloads](#)
- [Leading Innovation Web site](#)
- [developerWorks Rational](#)
- [IBM Rational TV](#)
- [IBM Rational Business Partners](#)

© Copyright IBM Corporation 2008. All rights reserved. The information contained in these materials is provided for informational purposes only, and is provided AS IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, these materials. Nothing contained in these materials is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software. References in these materials to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. Product release dates and/or capabilities referenced in these materials may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way. IBM, the IBM logo, the on-demand business logo, Rational, the Rational logo, and other IBM products and services are trademarks of the International Business Machines Corporation, in the United States, other countries or both. Other company, product, or service names may be trademarks or service marks of others.

