Comparing Application Security Tools

Defcon 15 - 8/3/2007

Eddie Lee
Fortify Software

Agenda

- Intro to experiment
- Methodology to reproduce experiment on your own
- Results from my experiment
- Conclusions

Introduction

- **Tools Used**
  - “Market Leading” Dynamic Testing Tools
  - A Static Code Analyzer
  - Dynamic Test Tracing Tool
- **The Application**
  - Open source Java based Blog
  - http://pebble.sourceforge.net
  - Reasons for choosing this application
- **The Experiment**
  - Out of the box scans
  - Compared findings from each tool

How The Tools Work

- **Dynamic Testing Tools**
  - Fuzz web form input
  - Signature and Behavioral Matching
  - Modes of Scanning
    - Auto-crawl
    - Manual crawl
- **Static Code Analyzer**
  - Data flow
  - Control flow
  - Semantic
- **Dynamic Test Tracing Tool**
  - Bytecode instrumentation
  - Monitor data coming in and out of the application
  - Run in conjunction with other dynamic testing tools

Methodology

How to reproduce experiments on your own (Dynamic Testing Tools)

- **Download source code**
- **Build & Deploy Application**
- **Figure out how to cleanly undeploy the application**
  - Clear database or stored files
- **Run scanner in mode auto-crawl mode**
  - Make sure the application doesn't break during your scans
    - If the app breaks, figure out why the scanner breaks the app.
    - Configure scanner to ignore the parameter(s) causing app to break
      - Note the parameter(s) won't be tested for vulnerabilities and the existence of a DoS vulnerability
    - Undeploy and Redeploy the application
    - Repeat
  - Save the results from your last clean run
  - Repeat for scanner in mode manual-crawl mode
- **Verify the results**
  - Verify results through manually testing
  - Record false positive rate
  - Normalize results
    - Record source file and line number information where vulnerabilities occur

How to reproduce experiments on your own (Static Testing Tool)

- **Not much to it**
  - Point the scanner at code and tell it where it can find needed libraries
- **Scan the same code you use in other tests**
- **Verify results are true positives and weed out false positives**
  - Verify results through manually testing on running application
  - Record false positive rate
  - Normalize the results

How to reproduce experiments on your own (Dynamic Tracing Tool)

- **Instrument the compiled code**
- **Deploy instrumented code**
- **Start recording**
- **Perform dynamic testing**
- **Stop recording**
- **Verify results are true positives and weed out false positives**
  - Verify results through manually testing on running application
  - Record false positive rate
  - Normalize the results

Setup and Result Quantification

- **Tool Configuration and Setup**
  - Dynamic Testing Tools
    - Modes of operation: Auto Crawl & Manual Crawl
    - Minor tweaking for the application
- **Quantification of Results**
  - Tools report vulnerabilities in different units
  - Standardized on location in source code where vulnerability occurs
    - Normalized reported numbers
  - Use the normalized vulnerability counts for comparison among tools

Results

Results: Overview

Vulnerability	File	Line number	URL	Parameter	Category	Tool #1a	Tool #1b	Tool #2a	Tool #2b	Tool #3a	Tool #4a	Tool #5a
1	blogEntry.jsp	16	saveBlogEntry.secureaction	title	XSS		X		X			
2	blogEntry.jsp	17	saveBlogEntry.secureaction	subtitle	XSS		X					
3	blogEntry.jsp	23	saveBlogEntry.secureaction	body	XSS		X					
4	blogEntry.jsp	28	saveBlogEntry.secureaction	excerpt	XSS		X					
5	blogEntryForm.jsp	58	saveBlogEntry.secureaction	title	XSS				X			
6	blogEntryForm.jsp	83	saveBlogEntry.secureaction	originalPermalink	XSS		X					
7	blogEntryForm.jsp	150	saveBlogEntry.secureaction	tags	XSS		X					
8	blogEntryForm.jsp	177	saveBlogEntry.secureaction	attachmentUrl	XSS		X					
9	blogEntryForm.jsp	185	saveBlogEntry.secureaction	attachmentType	XSS		X					
10	error.jsp	16	various		XSS		X					
11	error.jsp	17	various		XSS							X
12	error.jsp	18	various	-	XSS							X
13	error.jsp	19	various		XSS							X
14	error.jsp	23	various		XSS		X					
15	header.jspf	11	saveBlogProperties.secureAction	description	XSS		X					
16	header.jspf	12	saveBlogProperties.secureAction	author	XSS		X					
17	pageable.jsp	4	search.action	sort	XSS							X
18	pageable.jsp	10	search.action	sort	XSS							X
19	pageable.jsp	19	search.action	sort	XSS							X
20	staticPage.jsp	9	saveStaticPage.secureaction	title	XSS		X					
21	staticPage.jsp	10	saveStaticPage.secureaction	subtitle	XSS		X					
22	staticPage.jsp	13	saveStaticPage.secureaction	body	XSS		X					
23	staticPageForm.jsp	52	saveStaticPage.secureaction	name	XSS		X					
24	staticPageForm.jsp	75	saveStaticPage.secureaction	originalPermalink	XSS		X					
25	template.jsp	27	saveBlogProperties.secureAction	name	XSS		X					
26	template.jsp	28	saveBlogProperties.secureAction	description	XSS		X					
27	viewBlogProperties.jsp	28	saveBlogProperties.secureAction	name	XSS		X					
28	viewBlogProperties.jsp	37	saveBlogProperties.secureAction	description	XSS		X					
29	viewBlogProperties.jsp	46	saveBlogProperties.secureAction	image	XSS		X					
30	viewBlogProperties.jsp	55	saveBlogProperties.secureAction	author	XSS		X					
31	viewBlogProperties.jsp	64	saveBlogProperties.secureAction	email	XSS		X				X	
32	viewFiles.jsp	5	viewFiles.secureaction	file	XSS		X		X			
33	viewFiles.jsp	7	viewFiles.secureaction	path	XSS				X			
34	viewFiles.jsp	121	viewFiles.secureaction	path	XSS				X			
35	viewFiles.jsp	122	viewFiles.secureaction	file	XSS		X			X		
36	viewFiles.jsp	123	viewFiles.secureaction	file	XSS					X		
37	viewFiles.jsp	138	viewFiles.secureaction	path	XSS				X			
38	viewFiles.jsp	145	viewFiles.secureaction	path	XSS				X			
39	viewRefererFilters.jsp	34	addRefererFilters.secureaction	expression	XSS		X		X			
40	viewReferers.jsp	13	viewReferers.secureaction	year	XSS							X
41	viewReferers.jsp	14	viewReferers.secureaction	month	XSS							X
42	viewReferers.jsp	15	viewReferers.secureaction	day	XSS							X
43	viewResponses.jsp	31	viewResponses.secureaction	type	XSS	X	X	X	X	X	X	X
44	viewUser.jsp	48	viewResponses.secureaction	name	XSS		X		X			
45	viewUser.jsp	52	viewResponses.secureaction	emailAddress	XSS		X					
46	viewUser.jsp	56	viewResponses.secureaction	website	XSS		X					
47	viewUserDetails.jsp	30	saveUserDetails.secureaction	name	XSS		X					
48	viewUserDetails.jsp	34	saveUserDetails.secureaction	emailAddress	XSS		X					
49	viewUserDetails.jsp	38	saveUserDetails.secureaction	website	XSS		X					
50	DefaultSecurityRealm.java	213	addUser.secureaction	username	Path Manipulation							X
51	FileStaticPageDAO.java	103	editStaticPage.secureaction	page	Path Manipulation							X
52	RedirectView.java	85	logout.action	redirectUrl	Arbitrary URL Redirection	X		X				

X-Unique to Tool

X-Multiple Tools

Results: Overview

File	Line #	URL	Parameter	Category
blogEntry.jsp	16	saveBlogEntry.secureaction	title	XSS

Tool #1a	Tool #1b	Tool #2a	Tool #2b	Tool #3a	Tool #4a	Tool #5a
	X		X			

Results: Overview

Vulnerability	File	Line number	URL	Parameter	Category	Tool #1a	Tool #1b	Tool #2a	Tool #2b	Tool #3a	Tool #4a	Tool #5a
1	blogEntry.jsp	16	saveBlogEntry.secureaction	title	XSS		X		X			
2	blogEntry.jsp	17	saveBlogEntry.secureaction	subtitle	XSS		X					
3	blogEntry.jsp	23	saveBlogEntry.secureaction	body	XSS		X					
4	blogEntry.jsp	28	saveBlogEntry.secureaction	excerpt	XSS		X					
5	blogEntryForm.jsp	58	saveBlogEntry.secureaction	title	XSS				X			
6	blogEntryForm.jsp	83	saveBlogEntry.secureaction	originalPermalink	XSS		X					
7	blogEntryForm.jsp	150	saveBlogEntry.secureaction	tags	XSS		X					
8	blogEntryForm.jsp	177	saveBlogEntry.secureaction	attachmentUrl	XSS		X					
9	blogEntryForm.jsp	185	saveBlogEntry.secureaction	attachmentType	XSS		X					
10	error.jsp	16	various		XSS		X					
11	error.jsp	17	various		XSS							X
12	error.jsp	18	various	-	XSS							X
13	error.jsp	19	various		XSS							X
14	error.jsp	23	various		XSS		X					
15	header.jspf	11	saveBlogProperties.secureAction	description	XSS		X					
16	header.jspf	12	saveBlogProperties.secureAction	author	XSS		X					
17	pageable.jsp	4	search.action	sort	XSS							X
18	pageable.jsp	10	search.action	sort	XSS							X
19	pageable.jsp	19	search.action	sort	XSS							X
20	staticPage.jsp	9	saveStaticPage.secureaction	title	XSS		X					
21	staticPage.jsp	10	saveStaticPage.secureaction	subtitle	XSS		X					
22	staticPage.jsp	13	saveStaticPage.secureaction	body	XSS		X					
23	staticPageForm.jsp	52	saveStaticPage.secureaction	name	XSS		X					
24	staticPageForm.jsp	75	saveStaticPage.secureaction	originalPermalink	XSS		X					
25	template.jsp	27	saveBlogProperties.secureAction	name	XSS		X					
26	template.jsp	28	saveBlogProperties.secureAction	description	XSS		X					
27	viewBlogProperties.jsp	28	saveBlogProperties.secureAction	name	XSS		X					
28	viewBlogProperties.jsp	37	saveBlogProperties.secureAction	description	XSS		X					
29	viewBlogProperties.jsp	46	saveBlogProperties.secureAction	image	XSS		X					
30	viewBlogProperties.jsp	55	saveBlogProperties.secureAction	author	XSS		X					
31	viewBlogProperties.jsp	64	saveBlogProperties.secureAction	email	XSS		X				X	
32	viewFiles.jsp	5	viewFiles.secureaction	file	XSS		X		X			
33	viewFiles.jsp	7	viewFiles.secureaction	path	XSS				X			
34	viewFiles.jsp	121	viewFiles.secureaction	path	XSS				X			
35	viewFiles.jsp	122	viewFiles.secureaction	file	XSS		X			X		
36	viewFiles.jsp	123	viewFiles.secureaction	file	XSS					X		
37	viewFiles.jsp	138	viewFiles.secureaction	path	XSS				X			
38	viewFiles.jsp	145	viewFiles.secureaction	path	XSS				X			
39	viewRefererFilters.jsp	34	addRefererFilters.secureaction	expression	XSS		X		X			
40	viewReferers.jsp	13	viewReferers.secureaction	year	XSS							X
41	viewReferers.jsp	14	viewReferers.secureaction	month	XSS							X
42	viewReferers.jsp	15	viewReferers.secureaction	day	XSS							X
43	viewResponses.jsp	31	viewResponses.secureaction	type	XSS	X	X	X	X	X	X	X
44	viewUser.jsp	48	viewResponses.secureaction	name	XSS		X		X			
45	viewUser.jsp	52	viewResponses.secureaction	emailAddress	XSS		X					
46	viewUser.jsp	56	viewResponses.secureaction	website	XSS		X					
47	viewUserDetails.jsp	30	saveUserDetails.secureaction	name	XSS		X					
48	viewUserDetails.jsp	34	saveUserDetails.secureaction	emailAddress	XSS		X					
49	viewUserDetails.jsp	38	saveUserDetails.secureaction	website	XSS		X					
50	DefaultSecurityRealm.java	213	addUser.secureaction	username	Path Manipulation							X
51	FileStaticPageDAO.java	103	editStaticPage.secureaction	page	Path Manipulation							X
52	RedirectView.java	85	logout.action	redirectUrl	Arbitrary URL Redirection	X		X				

X-Unique to Tool

X-Multiple Tools

Results: Exploit Examples

- **Cross-Site Scripting**
  - **Error.jsp:18**

  Code:

  Request URI : ${pageContext.request.requestURI}

  Attack:

  http://host/pebble/</textarea><script>alert(123)</script>/createDirectory.secureaction?type=blogFile

  - **viewResponses.jsp:31**

  Code:

  <input type="hidden" name="type" value="${param.type}" />

  Attack:

  http://host/pebble/viewResponses.secureaction?type="><script>alert(1)</script>

Results: Exploit Examples

- **Path Manipulation**
  - **DefaultSecurityRealm.java:213**

  Code:

  return new File(getFileForRealm(), username + ".properties");

  Attack:

  http://host/pebble/saveUser.secureaction?username=../../../../../../../../etc/passwd%00&newUser=true&name=joe&emailAddress=me@mydomain.com&website=blah.com

- **Arbitrary URL Redirection**
  - **RedirectView.java:85**

  Code:

  response.sendRedirect(getUri());

  Attack:

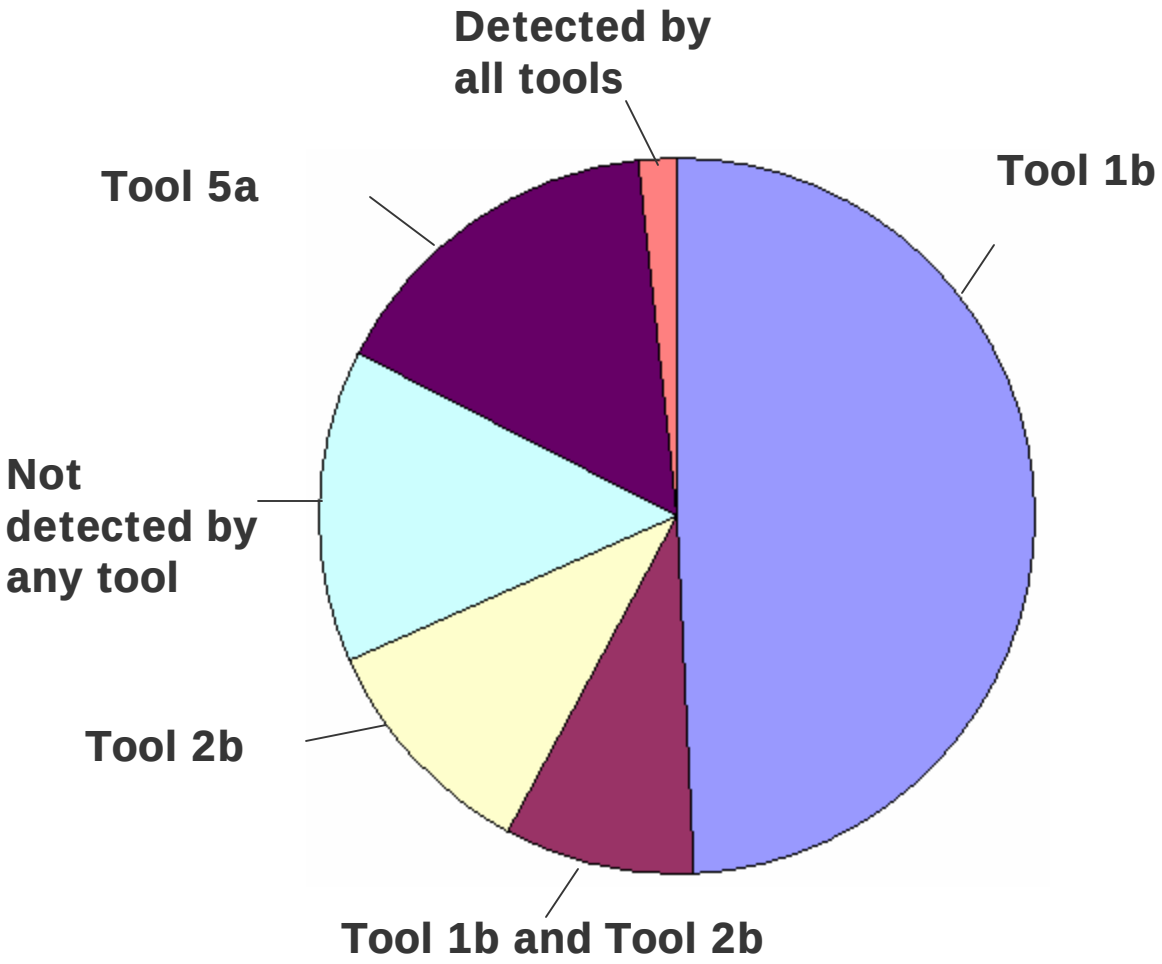
  http://host/pebble/logout.action?redirectUrl=http://www.attacker.com

Results: Manual Audit

Vulnerabilities not detected by any tool (from just one file)

File	Line number	URL	Parameter	Category
nanageCategories.jsp	34	editCategory.secureaction	id	XSS
nanageCategories.jsp	37	editCategory.secureaction	id	XSS
nanageCategories.jsp	37	editCategory.secureaction	name	XSS
nanageCategories.jsp	38	editCategory.secureaction	id	XSS
nanageCategories.jsp	39	editCategory.secureaction	tags	XSS
nanageCategories.jsp	75	editCategory.secureaction	id	XSS
nanageCategories.jsp	83	editCategory.secureaction	name	XSS
nanageCategories.jsp	89	editCategory.secureaction	tags	XSS

Cross-Site Scripting Detection By Tool



*1a, 2a, 3a and 4a not shown because findings were not significant

Conclusions

- **A single tool doesn’t cut it**
  - Using multiple tools significantly increases vulnerabilities found
- **Little overlap between tools**
- **Tools alone aren’t enough**
- **Run these tests on your own apps to see how they perform in your environment**
- **Fuzzing tools break shit**
  - Takes a long time to scan and troubleshoot the application
  - Don’t expect these tests to be quick

Q&A

Thanks!

