



the leading secure software development firm

## **Streamlining Application Vulnerability Management: Communication Between Development and Security Teams**

**October 13, 2012**

**OWASP Boston Application Security Conference**

## Agenda

- Introduction / Background
- Vulnerabilities
  - *Infrastructure (Network) vs. Application (Software)*
- Roles
  - *Security vs. Development*
- Vulnerability Workflow
- ThreadFix: An Open Source Tool
- Questions

## Introduction / Background

- Me (Brian Mather)
  - *Product & Consulting Manager at Denim Group*
  - *5 years experience managing app development & security assessment projects*
  - *13 years in information technology/security industry*
  - *Managing partner at IT services company for 10 years*
- Denim Group
  - *Headquarters in San Antonio, TX*
  - *Professional services firm that offers a unique service blend*
    - Builds & secures enterprise applications
    - Application Security
    - Developer Education (ILT & eLearning)
  - *Customer base spans Fortune 500*
    - Market Focus: Financial Services, Banking, Insurance, Healthcare, and Defense
  - *Contributes to industry best practices through the Open Web Application Security Project (OWASP)*

## Vulnerabilities: Defined

- Infrastructure (Network):
  - *any flaw or weakness in network defense that could be exploited to gain unauthorized access to, damage , or otherwise affect a network*
- Application (Software):
  - *a weakness in an application, either a design flaw or an implementation bug, that allows an attacker to cause harm to the stakeholders of an application.*

Problem isn't finding vulnerabilities, it's fixing them

- *Identifying application-level vulnerabilities via scanning tools, penetration tests and code reviews is only the first step in actually addressing the underlying risk.*

## Vulnerability Fun Facts:

Industry	Annual Avg. Vulnerabilities	Avg. Time-to-Fix (Days)	Average Remediation	Window of Exposure (Days)
ALL	79	38	63%	231
Banking	17	45	74%	185
Education	53	30	46%	261
Financial Services	67	80	63%	227
Healthcare	48	35	63%	239
Insurance	92	40	58%	211
IT	85	35	57%	208
Manufacturing	30	17	50%	252
Retail	121	27	66%	238
Social Networking	31	41	62%	264
Telecom	52	50	69%	271
Non-Profit	37	94	56%	320
Energy	31	4	40%	250

- Average number of serious vulnerabilities found per website per year is 79 \*\*
- Serious Vulnerabilities were fixed in ~38 days \*\*
- Percentage of serious vulnerabilities fixed annually is only 63% \*\*
- Average number of days a website is exposed, at least one serious vulnerability ~231 days

WhiteHat Statistics Report (Summer 2012):

[https://www.whitehatsec.com/assets/WPstats\\_summer12\\_12th.pdf](https://www.whitehatsec.com/assets/WPstats_summer12_12th.pdf)

## Vulnerability Remediation Data

Vulnerability Type	Sample Count	Average Fix (minutes)
Dead Code (unused methods)	465	2.6
Poor logging: system output stream	83	2.9
Poor Error Handling: Empty catch block	180	6.8
Lack of Authorization check	61	6.9
Unsafe threading	301	8.5
ASP.NET non-serializable object in session	42	9.3
XSS (stored)	1023	9.6
Null Dereference	157	10.2
Missing Null Check	46	15.7
XSS (reflected)	25	16.2
Redundant null check	21	17.1
SQL injection	30	97.5

## Security Team:

### Identify / Communicate Risk

- *Penetration Testing*
- *Application Scanning*
- *Protecting Assets*
- *Mitigating Risk*

Typically, teams that find vulnerabilities (**Security**) → don't know how to fix / remediate

VS.

## Development Team:

### Building Software

- *Feature Development*
- *Application Performance*
- *Bug Fixes*
- *Deployments*

Typically, teams that fix vulnerabilities (**Development**) → don't understand the potential business risk / impact

## Vulnerability Workflow:

- **Typical Security Workflow**

- *Runs a scan → produce PDF → print/email to development = **BAD***
- *Runs 2 scans → produce 2 PDFs → print/email to development = **WORSE***
- *Runs 2 scans → merging vulnerabilities into excel → print/email to development = **HORRIBLE***

**“Let the negotiations begin”**

- **Typical Development Workflow**

- *Developers informed of vulnerabilities → with little / no context provided (no steps to reproduce)*
- *Ticket created in defect tracker (maybe?) → assign to developer*
- *Developer fixes bug → ticket updated in defect tracker → notify security team of fix (maybe?)*

**“Can we get back to our development schedule yet?”**



## Vulnerability Workflow:

- **Managing Application Vulnerabilities**
  - *Actual business risk is challenging to determine*
  - *More challenging than infrastructure vulnerabilities (patching / configuration changes)*
  - *Changes to custom code and application-specific business logic*
  - *Requires coordinated effort between security & development teams*
- **Inefficient process:**
  - *Difficulty making sense of and prioritizing data in (overlapping) scanning reports*
  - *Different teams use different scanning tools (tools use different terms and severities)*
  - *Lack of centralized management/view*
  - *Friction/Negative interaction between security & development teams*
- **Remediation becomes an overwhelming project**
  - *Security managers need to request time from developers (already-cramped dev/release schedules)*
  - *Development doesn't have or want to give up time to fix vulnerabilities*
  - *Hesitation scanning new apps, fear of finding new vulnerabilities when queue isn't clearing fast enough*
- **Creating trending reports is impractical**
  - *Lack of visibility across app portfolios*
  - *Without consistent language and consolidated data, knowing whether your organization is actually reducing the number of vulnerabilities is impossible*

*“Two teams with different focuses, however both teams play a critical role in the remediation of application vulnerabilities, and need to communicate.”*

**What can be done to solve this problem?**

## The ThreadFix Approach

- An open source vulnerability management and aggregation platform that allows software security teams to reduce the time it takes to fix software vulnerabilities
- Freely available under the Mozilla Public License (MPL)
- Download available at: [www.denimgroup.com/threadfix](http://www.denimgroup.com/threadfix)



## ThreadFix: Accelerate Software Remediation

- **Application Portfolio Management**
  - *One central, canonical location to keep track of all of the organization's applications*
- **Vulnerability Import**
  - *Supports dynamic and static results from a variety of sources (both commercial and freely available scanning tools, manual testing, and SaaS testing providers)*
  - *De-duplicate scan results (1 vulnerability found by 4 tools vs. 4 vulns)*
- **Defect Tracking Integration**
  - *Allows application security teams to slice/dice, bundle, and ship vulnerabilities over to development staff using tools they are familiar with and currently using*
- **Real-Time Protection Generation**
  - *Application-specific rules based on identified vulnerabilities & associated attack data*
  - *Virtual patching helps protect organization and eliminate false positives blocks*
- **Maturity Evaluation**
  - *Report on software security program progress*
  - *Benchmark security practice improvement against industry standards*

## Supported Tools:

### Dynamic Scanners

*Burp Suite*  
*HP WebInspect*  
*Mavituna Security Netsparker*  
*Tenable Nessus*  
*Acunetix*  
*OWASP Zed Attack Proxy*  
*Arachni*  
*Skipfish*  
*w3aF*

### Static Scanners

*HP Fortify SCA*  
*Microsoft CAT.NET*  
*FindBugs*  
*Ounce IBM Security AppScan Source*  
*Brakeman*

### SaaS Testing Platforms

*WhiteHat*  
*Veracode*  
*QualysGuard WAS 2.0*

### IDS/IPS and WAF

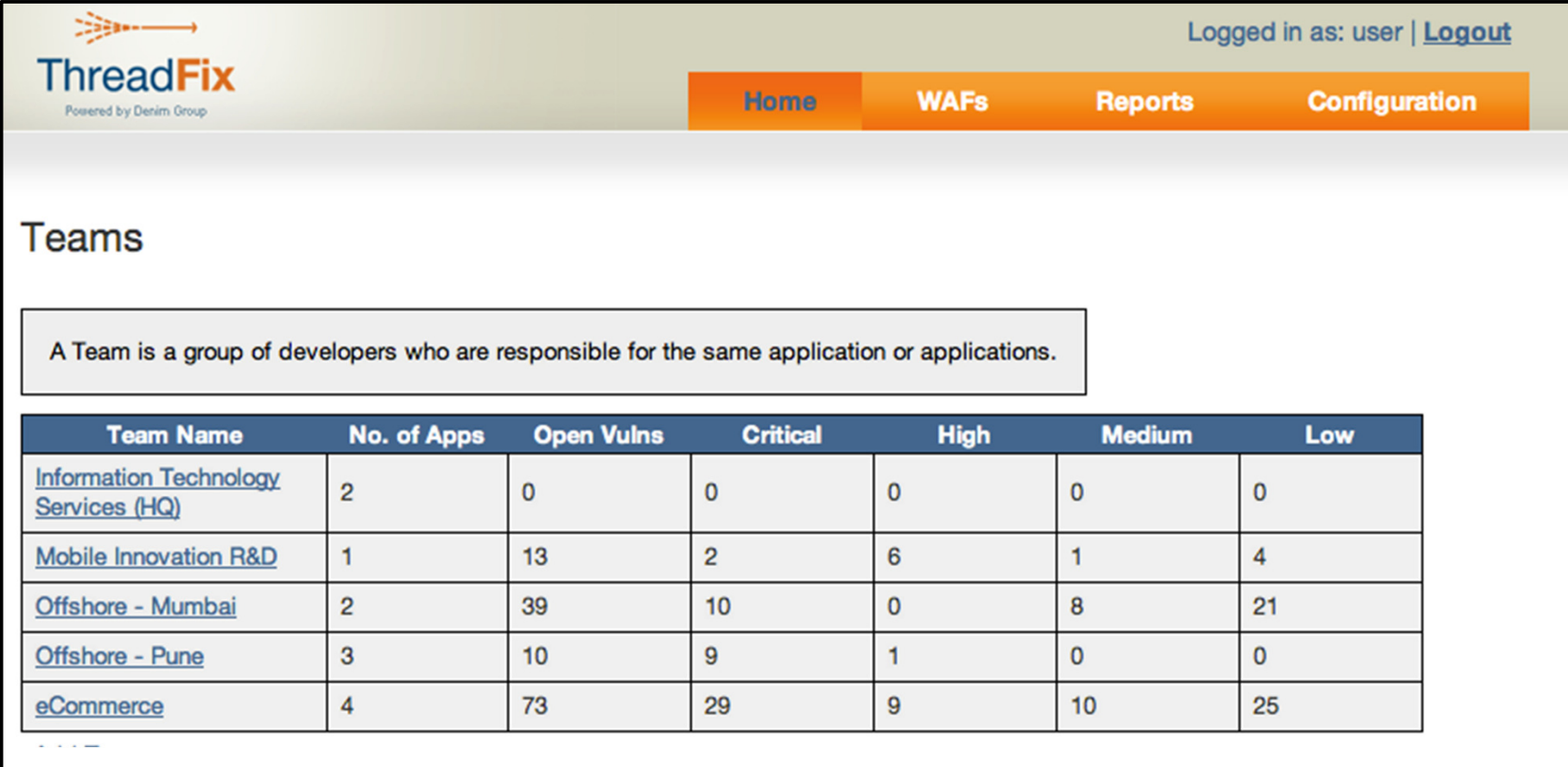
*Snort*  
*mod\_security*  
*Imperva*  
*F5*  
*DenyAll*

### Defect Trackers

*Mozilla Bugzilla*  
*Atlassian JIRA*

## Dashboard

- List of development teams in the organization, including number of apps for each team and a summary of the security status of those apps.



ThreadFix  
Powered by Denim Group

Logged in as: user | [Logout](#)

[Home](#) [WAFs](#) [Reports](#) [Configuration](#)

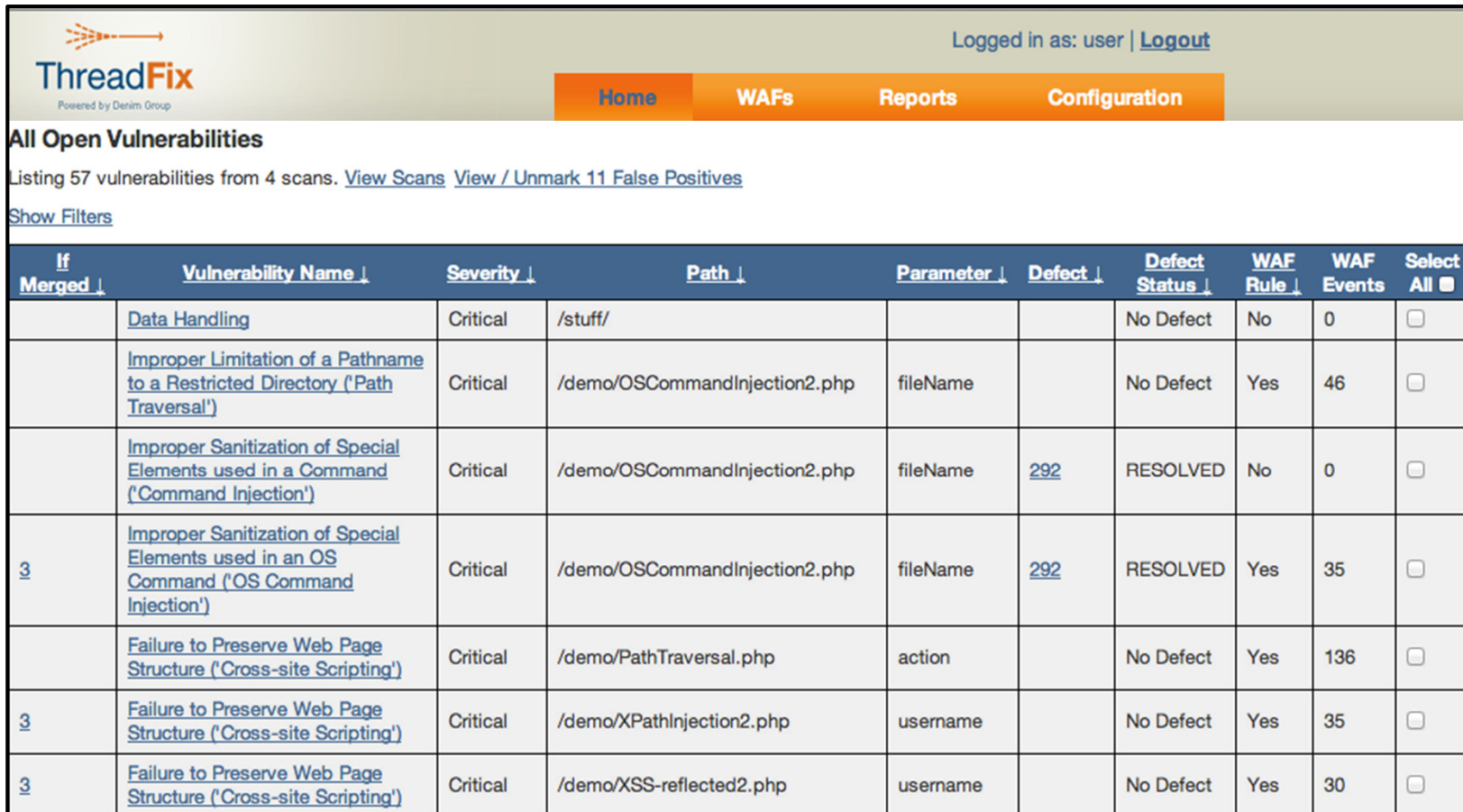
### Teams

A Team is a group of developers who are responsible for the same application or applications.

Team Name	No. of Apps	Open Vulns	Critical	High	Medium	Low
<a href="#">Information Technology Services (HQ)</a>	2	0	0	0	0	0
<a href="#">Mobile Innovation R&amp;D</a>	1	13	2	6	1	4
<a href="#">Offshore - Mumbai</a>	2	39	10	0	8	21
<a href="#">Offshore - Pune</a>	3	10	9	1	0	0
<a href="#">eCommerce</a>	4	73	29	9	10	25

## ThreadFix Consolidation

- Vulnerability scans are aggregated providing a centralized view of the security status of an application.



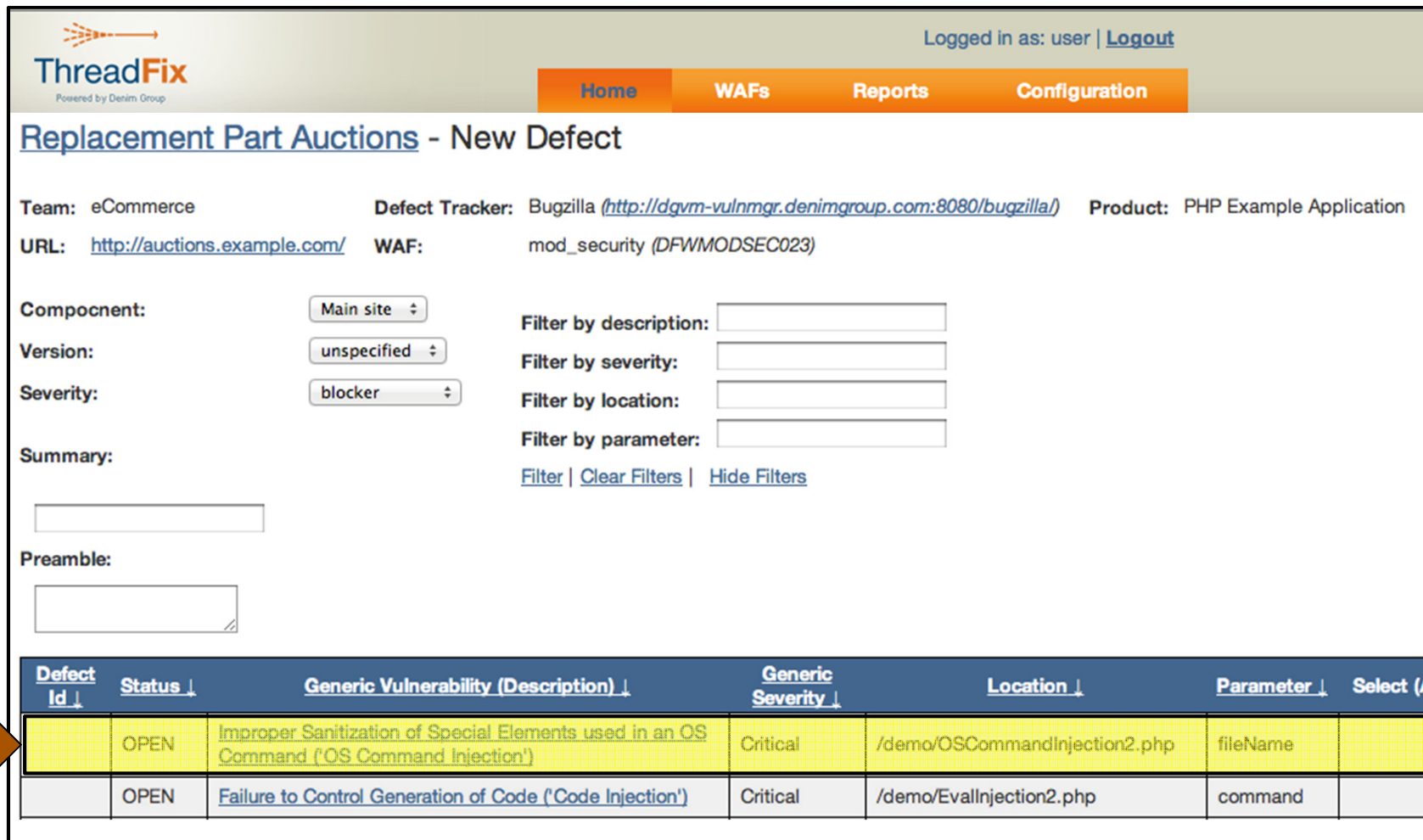
The screenshot shows the ThreadFix web interface. At the top, it says "ThreadFix Powered by Denim Group" and "Logged in as: user | Logout". There are navigation tabs for "Home", "WAFs", "Reports", and "Configuration". Below the navigation, it says "All Open Vulnerabilities" and "Listing 57 vulnerabilities from 4 scans. View Scans View / Unmark 11 False Positives". There is a "Show Filters" link. The main content is a table of vulnerabilities.

If Merged ↓	Vulnerability Name ↓	Severity ↓	Path ↓	Parameter ↓	Defect ↓	Defect Status ↓	WAF Rule ↓	WAF Events	Select All <input type="checkbox"/>
	Data Handling	Critical	/stuff/			No Defect	No	0	<input type="checkbox"/>
	<a href="#">Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal')</a>	Critical	/demo/OSCommandInjection2.php	fileName		No Defect	Yes	46	<input type="checkbox"/>
	<a href="#">Improper Sanitization of Special Elements used in a Command ('Command Injection')</a>	Critical	/demo/OSCommandInjection2.php	fileName	<a href="#">292</a>	RESOLVED	No	0	<input type="checkbox"/>
<a href="#">3</a>	<a href="#">Improper Sanitization of Special Elements used in an OS Command ('OS Command Injection')</a>	Critical	/demo/OSCommandInjection2.php	fileName	<a href="#">292</a>	RESOLVED	Yes	35	<input type="checkbox"/>
	<a href="#">Failure to Preserve Web Page Structure ('Cross-site Scripting')</a>	Critical	/demo/PathTraversal.php	action		No Defect	Yes	136	<input type="checkbox"/>
<a href="#">3</a>	<a href="#">Failure to Preserve Web Page Structure ('Cross-site Scripting')</a>	Critical	/demo/XPathInjection2.php	username		No Defect	Yes	35	<input type="checkbox"/>
<a href="#">3</a>	<a href="#">Failure to Preserve Web Page Structure ('Cross-site Scripting')</a>	Critical	/demo/XSS-reflected2.php	username		No Defect	Yes	30	<input type="checkbox"/>



## Agreeing On The Workload

- Bundling multiple instances of the same vulnerability into a single defect
- ThreadFix integrates with Mozilla Bugzilla and Atlassian JIRA



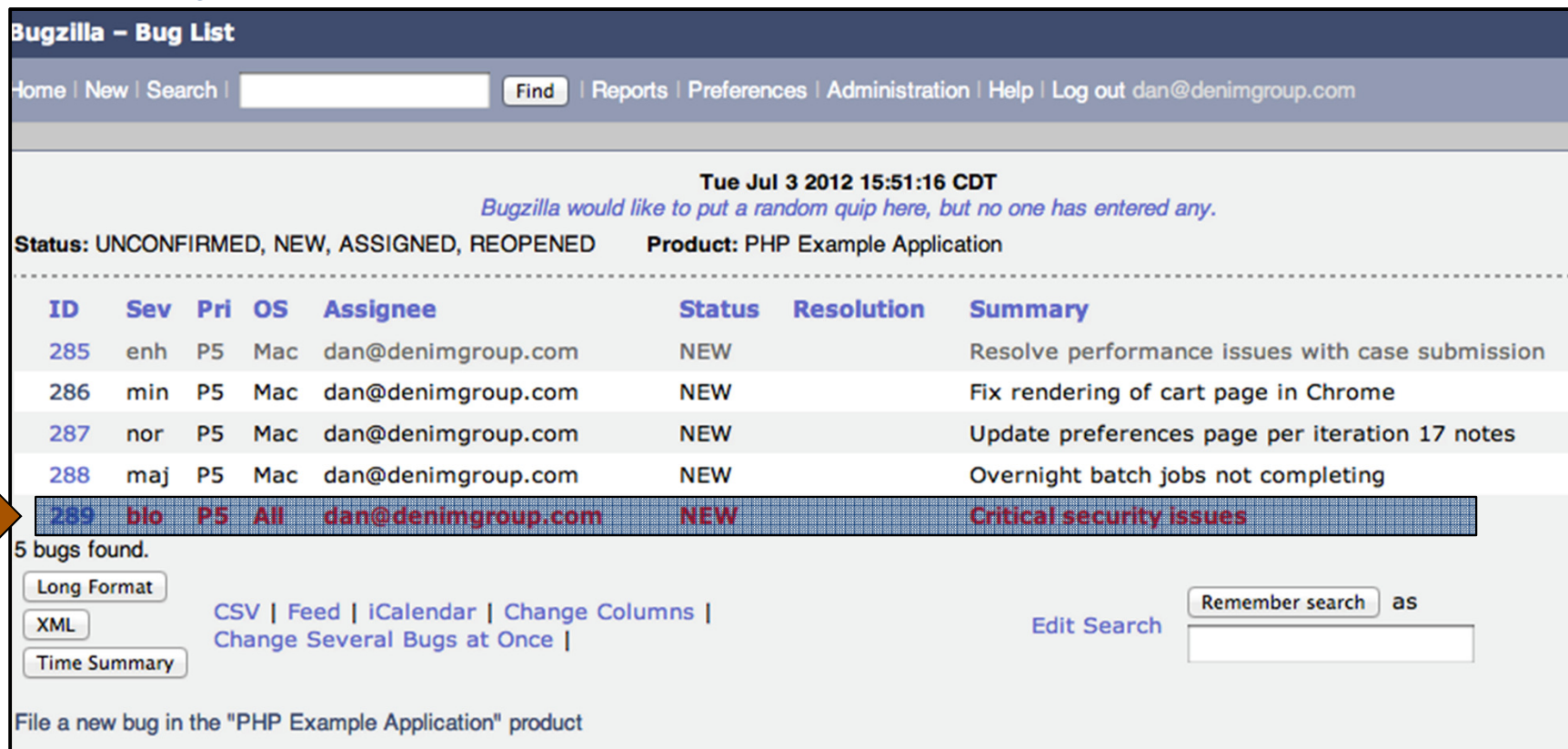
The screenshot shows the ThreadFix web interface. At the top, it says 'ThreadFix Powered by Denim Group' and 'Logged in as: user | Logout'. The navigation menu includes 'Home', 'WAFs', 'Reports', and 'Configuration'. The main heading is 'Replacement Part Auctions - New Defect'. Below this, there are fields for 'Team: eCommerce', 'Defect Tracker: Bugzilla (http://dgvm-vulnmgr.denimgroup.com:8080/bugzilla/)', 'Product: PHP Example Application', 'URL: http://auctions.example.com/', and 'WAF: mod\_security (DFWMODSEC023)'. There are also dropdown menus for 'Component: Main site', 'Version: unspecified', and 'Severity: blocker'. To the right, there are input fields for 'Filter by description:', 'Filter by severity:', 'Filter by location:', and 'Filter by parameter:'. Below these are links for 'Filter', 'Clear Filters', and 'Hide Filters'. There are also text areas for 'Summary:' and 'Preamble:'. At the bottom, there is a table with columns: 'Defect Id', 'Status', 'Generic Vulnerability (Description)', 'Generic Severity', 'Location', 'Parameter', and 'Select ( )'. The first row is highlighted in yellow and has an orange arrow pointing to it from the left. The second row is in a light grey color.

Defect Id	Status	Generic Vulnerability (Description)	Generic Severity	Location	Parameter	Select ( )
	OPEN	Improper Sanitization of Special Elements used in an OS Command ('OS Command Injection')	Critical	/demo/OSCommandInjection2.php	fileName	
	OPEN	Failure to Control Generation of Code ('Code Injection')	Critical	/demo/EvallInjection2.php	command	



## The Defect Tracking System

- Security analyst exports vulnerabilities with Critical Severity to the Defect Tracking System ([Bugzilla in this example](#)).
- The development team then uses Bugzilla to keep track of outstanding bugs and management tasks still to be done.



**Bugzilla - Bug List**

Home | New | Search |   | Reports | Preferences | Administration | Help | Log out dan@denimgroup.com

Tue Jul 3 2012 15:51:16 CDT  
*Bugzilla would like to put a random quip here, but no one has entered any.*

Status: UNCONFIRMED, NEW, ASSIGNED, REOPENED    Product: PHP Example Application

ID	Sev	Pri	OS	Assignee	Status	Resolution	Summary
285	enh	P5	Mac	dan@denimgroup.com	NEW		Resolve performance issues with case submission
286	min	P5	Mac	dan@denimgroup.com	NEW		Fix rendering of cart page in Chrome
287	nor	P5	Mac	dan@denimgroup.com	NEW		Update preferences page per iteration 17 notes
288	maj	P5	Mac	dan@denimgroup.com	NEW		Overnight batch jobs not completing
289	blo	P5	All	dan@denimgroup.com	NEW		Critical security issues

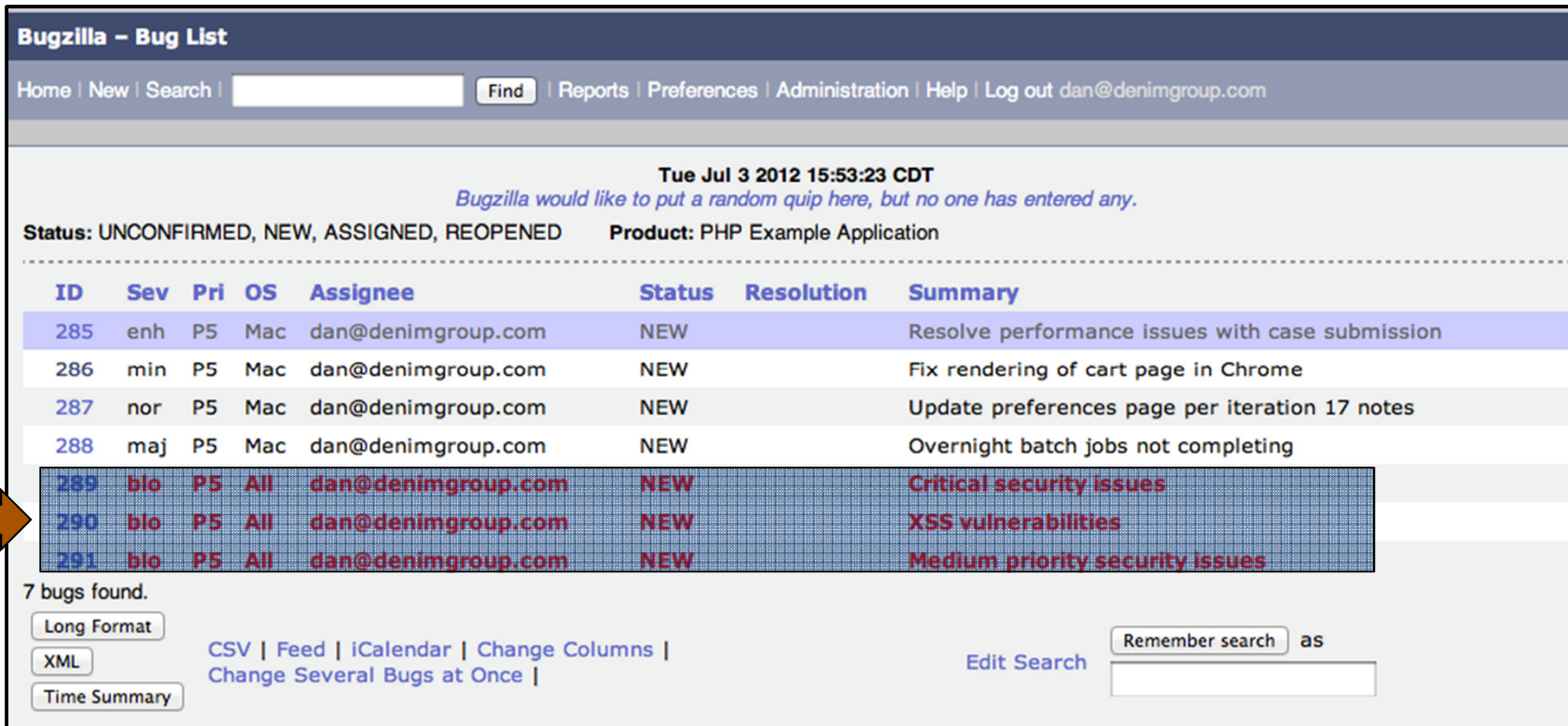
5 bugs found.

           [CSV](#) | [Feed](#) | [iCalendar](#) | [Change Columns](#) | [Change Several Bugs at Once](#) | [Edit Search](#)     as

File a new bug in the "PHP Example Application" product

## Vulnerabilities Now Become Defects

- Vulnerabilities are packaged in a manner that makes sense to the development team's workflow.
- These vulnerabilities, now recognized as defects, are transferred to Bugzilla, the platform the development team is used to using.



**Bugzilla - Bug List**

Home | New | Search |   | Reports | Preferences | Administration | Help | Log out dan@denimgroup.com

Tue Jul 3 2012 15:53:23 CDT  
*Bugzilla would like to put a random quip here, but no one has entered any.*

Status: UNCONFIRMED, NEW, ASSIGNED, REOPENED    Product: PHP Example Application

ID	Sev	Pri	OS	Assignee	Status	Resolution	Summary
285	enh	P5	Mac	dan@denimgroup.com	NEW		Resolve performance issues with case submission
286	min	P5	Mac	dan@denimgroup.com	NEW		Fix rendering of cart page in Chrome
287	nor	P5	Mac	dan@denimgroup.com	NEW		Update preferences page per iteration 17 notes
288	maj	P5	Mac	dan@denimgroup.com	NEW		Overnight batch jobs not completing
289	blo	P5	All	dan@denimgroup.com	NEW		Critical security issues
290	blo	P5	All	dan@denimgroup.com	NEW		XSS vulnerabilities
291	blo	P5	All	dan@denimgroup.com	NEW		Medium priority security issues

7 bugs found.

           [CSV](#) | [Feed](#) | [iCalendar](#) | [Change Columns](#) | [Change Several Bugs at Once](#)

as     [Edit Search](#)

## Defect Categories & Status inside ThreadFix

- Security analyst can see all open vulnerabilities, including defects they are linked to.
- Currently view: none of the bugs have been resolved by the development team.

ThreadFix <small>Powered by Denim Group</small>									
Logged in as: user   <a href="#">Logout</a>									
<a href="#">Home</a> <a href="#">WAFs</a> <a href="#">Reports</a> <a href="#">Configuration</a>									
If Merged ↓	Vulnerability Name ↓	Severity ↓	Path ↓	Parameter ↓	Defect ↓	Defect Status ↓	WAF Rule ↓	WAF Events	
First Defect ➔	2	Improper Sanitization of Special Elements used in an OS Command ('OS Command Injection')	Critical	/demo/OSCommandInjection2.php	fileName	289	OPEN	No	0
		Failure to Control Generation of Code ('Code Injection')	Critical	/demo/EvalInjection2.php	command	289	OPEN	No	0
Second Defect ➔		Failure to Preserve Web Page Structure ('Cross-site Scripting')	High	/demo/XPathInjection2.php	password	290	OPEN	No	0
		Failure to Preserve Web Page Structure ('Cross-site Scripting')	High	/demo/EvalInjection2.php	command	290	OPEN	No	0
		Failure to Preserve Web Page Structure ('Cross-site Scripting')	High	/demo/XSS-reflected2.php	username	290	OPEN	No	0
		Failure to Preserve Web Page Structure ('Cross-site Scripting')	High	/demo/SQLi2.php	username	290	OPEN	No	0
		Failure to Preserve Web Page Structure ('Cross-site Scripting')	High	/demo/XPathInjection2.php	username	290	OPEN	No	0
Third Defect ➔		Improper Control of Resource Identifiers ('Resource Injection')	High	/demo/OSCommandInjection2.php	fileName	291	OPEN	No	0
		Information Leak Through Include Source Code	Medium	/demo/OSCommandInjection2.php	fileName	291	OPEN	No	0

## A Defect (Security Vulnerability) Is Fixed (Or is it?)

- The developers review the bug containing the Critical vulnerabilities.
- They work with representatives from security to resolve the issue and then mark the bug as fixed in Bugzilla.

**Bugzilla - Bug List**

Home | New | Search |   | Reports | Preferences | Administration | Help | Log out dan@denimgroup.com

**Blocks:**

Show dependency [tree](#) / [graph](#)

---

Orig. Est.	Current Est.	Hours Worked	Hours Left	%Complete	Gain	Deadline
<input type="text" value="0.0"/>	0.0	0.0 + <input type="text" value="0"/>	<input type="text" value="0.0"/>	0	0.0	<input type="text" value=""/> (YYYY-MM-DD)

[Summarize time \(including time for bugs blocking this bug\)](#)

**Attachments**

[Add an attachment](#) (proposed patch, testcase, etc.)

**Additional Comments:**

Worked with Terry in security and the issues should be resolved. Please re-scan and we will deploy to

**Status:** RESOLVED  FIXED

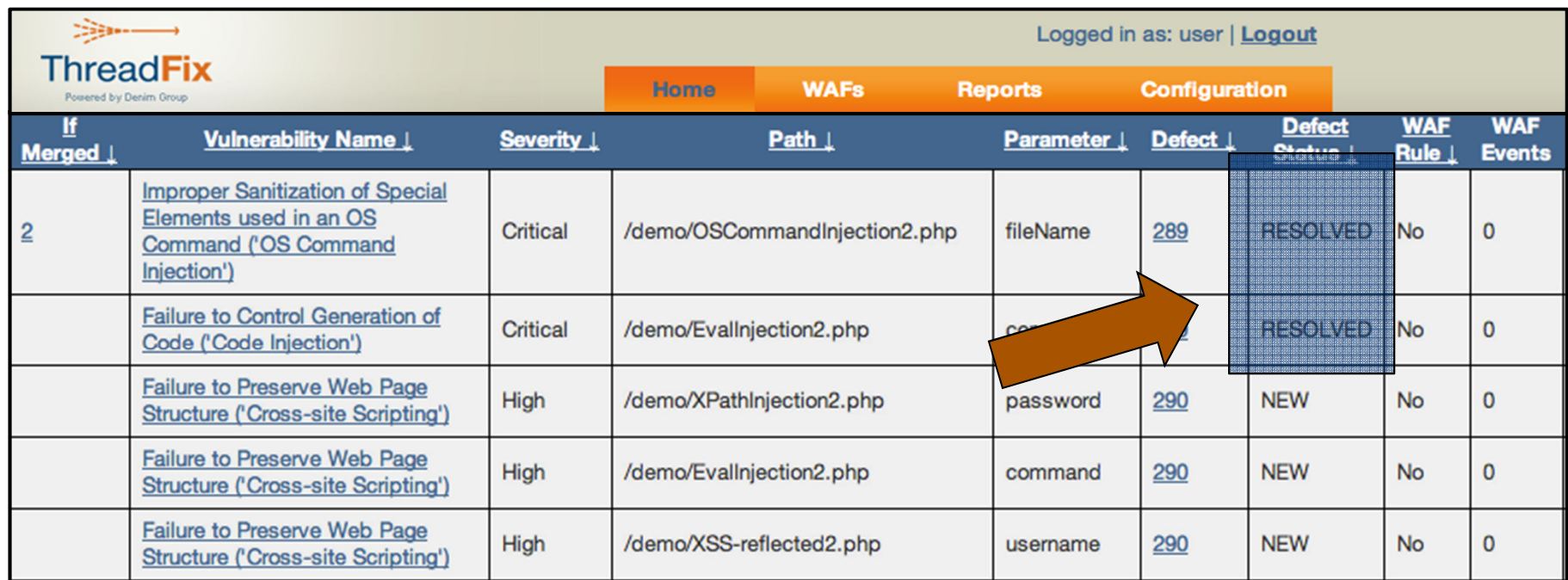
[Mark as Duplicate](#)





## Bugzilla Updates Are Synchronized With ThreadFix

- When a ThreadFix update is performed, Bugzilla's developer notes regarding bug status are synchronized with ThreadFix
- The security team then performs additional scans to confirm that the bugs have, indeed, been fixed.



ThreadFix  
Powered by Denim Group

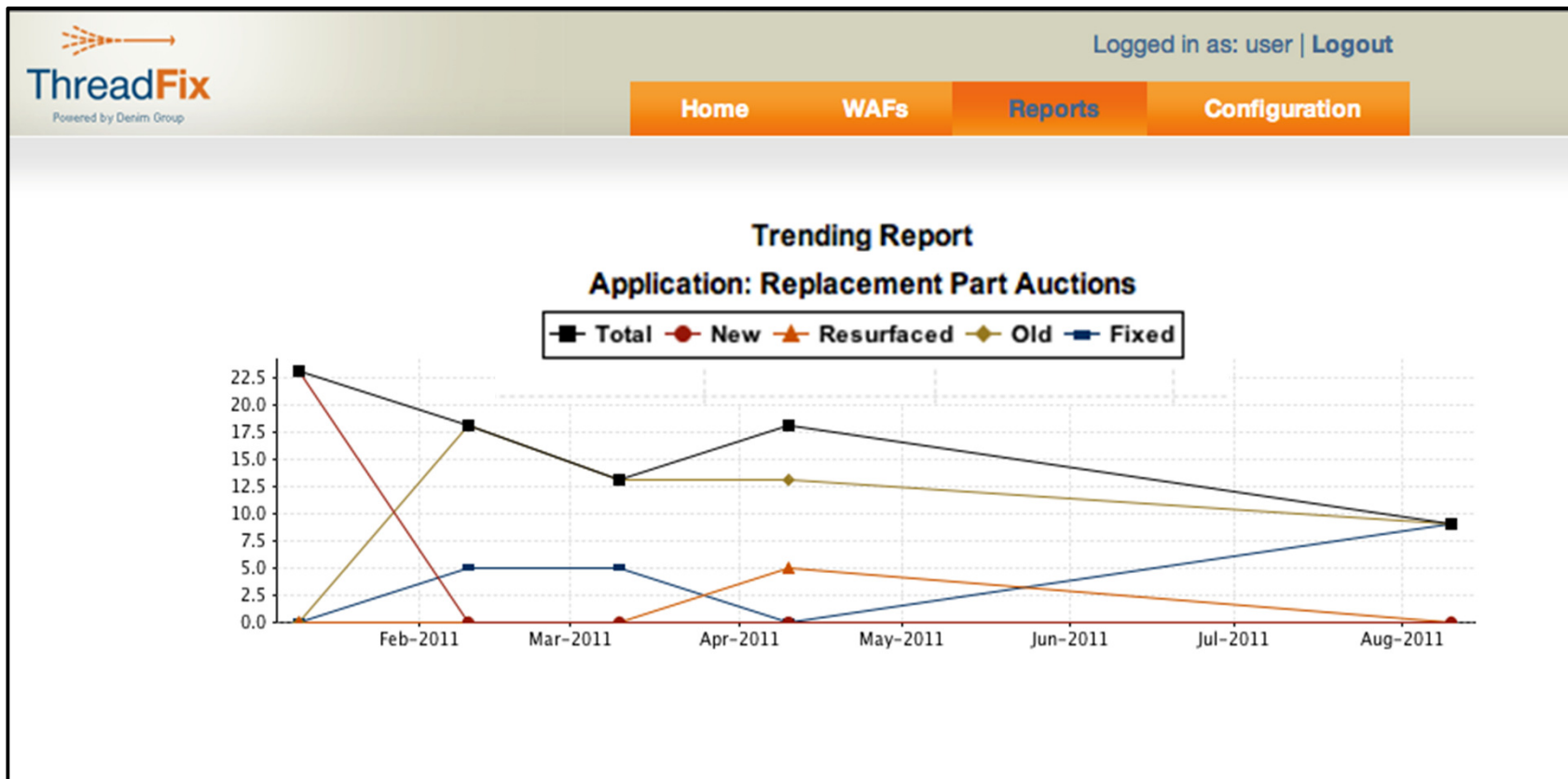
Logged in as: user | [Logout](#)

Home WAFs Reports Configuration

If Merged ↓	Vulnerability Name ↓	Severity ↓	Path ↓	Parameter ↓	Defect ↓	Defect Status ↓	WAF Rule ↓	WAF Events
<a href="#">2</a>	<a href="#">Improper Sanitization of Special Elements used in an OS Command ('OS Command Injection')</a>	Critical	/demo/OSCommandInjection2.php	fileName	<a href="#">289</a>	RESOLVED	No	0
	<a href="#">Failure to Control Generation of Code ('Code Injection')</a>	Critical	/demo/EvallInjection2.php	command	<a href="#">289</a>	RESOLVED	No	0
	<a href="#">Failure to Preserve Web Page Structure ('Cross-site Scripting')</a>	High	/demo/XPathInjection2.php	password	<a href="#">290</a>	NEW	No	0
	<a href="#">Failure to Preserve Web Page Structure ('Cross-site Scripting')</a>	High	/demo/EvallInjection2.php	command	<a href="#">290</a>	NEW	No	0
	<a href="#">Failure to Preserve Web Page Structure ('Cross-site Scripting')</a>	High	/demo/XSS-reflected2.php	username	<a href="#">290</a>	NEW	No	0

## Trending Reports Help Improve Quality

By repeating this process over time, the security teams can start to collect trending data about vulnerabilities as well as statistics of how long it is taking to resolve security issues.



## Summary

- Communication between security & development teams is inefficient
- Current Vulnerability Management process
- ThreadFix facilitates communication between security & development
  - *Integrating with commercial and open source scanners & defect trackers*
  - *Reducing the time required to fix vulnerable applications.*
  - *Dramatically simplifying remediation effort required*
  - *Providing centralized visibility into current security state of applications*
  - *Giving security ability to benchmark progress & track progress over time*
- No licensing fees
  - *Freely available under the Mozilla Public License (MPL) via Google Code*
- Open community support

## Where to Get ThreadFix

- For more information, go to <http://www.denimgroup.com/threadfix>
- Directed to a **Google Code Repository** and download the zip file.
- Click on the Threadfix.bat icon in Windows, or, in Linux, navigate to the folder and execute `bash threadfix.sh`.
- Go on the wiki and open the “Getting Started” file for more step by step directions.



## Contact Information

**Brian Mather**  
Product & Consulting Manager  
[brian@denimgroup.com](mailto:brian@denimgroup.com)  
(210) 572-4400

[www.denimgroup.com](http://www.denimgroup.com)  
[www.threadstrong.com](http://www.threadstrong.com)  
[blog.denimgroup.com](http://blog.denimgroup.com)