Automating Vulnerability Management @ Orbitz with SCAP

Security Automation Conference 10.27.2009

Ed Bellis VP, CISO
Orbitz Worldwide
ed@orbitz.com

Context Matters...

- Orbitz Worldwide
  - Orbitz & OFB
  - Cheaptickets
  - Away.com
  - eBookers
  - HotelClub & Rates2Go
  - Traveler Care
  - AA & NWA Booking engines
  - msn.orbitz.com
  - Southwest Hotels
  - Trip.com
  - Orbitzgames.com
  - RBS Rewards

...and on and on and on...

Context Matters...

- 100’s of Endless Applications
- 1000’s of Servers
- 1000’s of Devices
- 100’s of DBs
- Data centers: multiple continents
- Call centers - follow the sun

...and on and on and on...

Context Matters...

- VA Tools
  - Application
  - Network & Host
  - Database
- Remediation Tracking
  - Jira
  - Remedy

...and on and on and on...

Our Solution: A Case Study

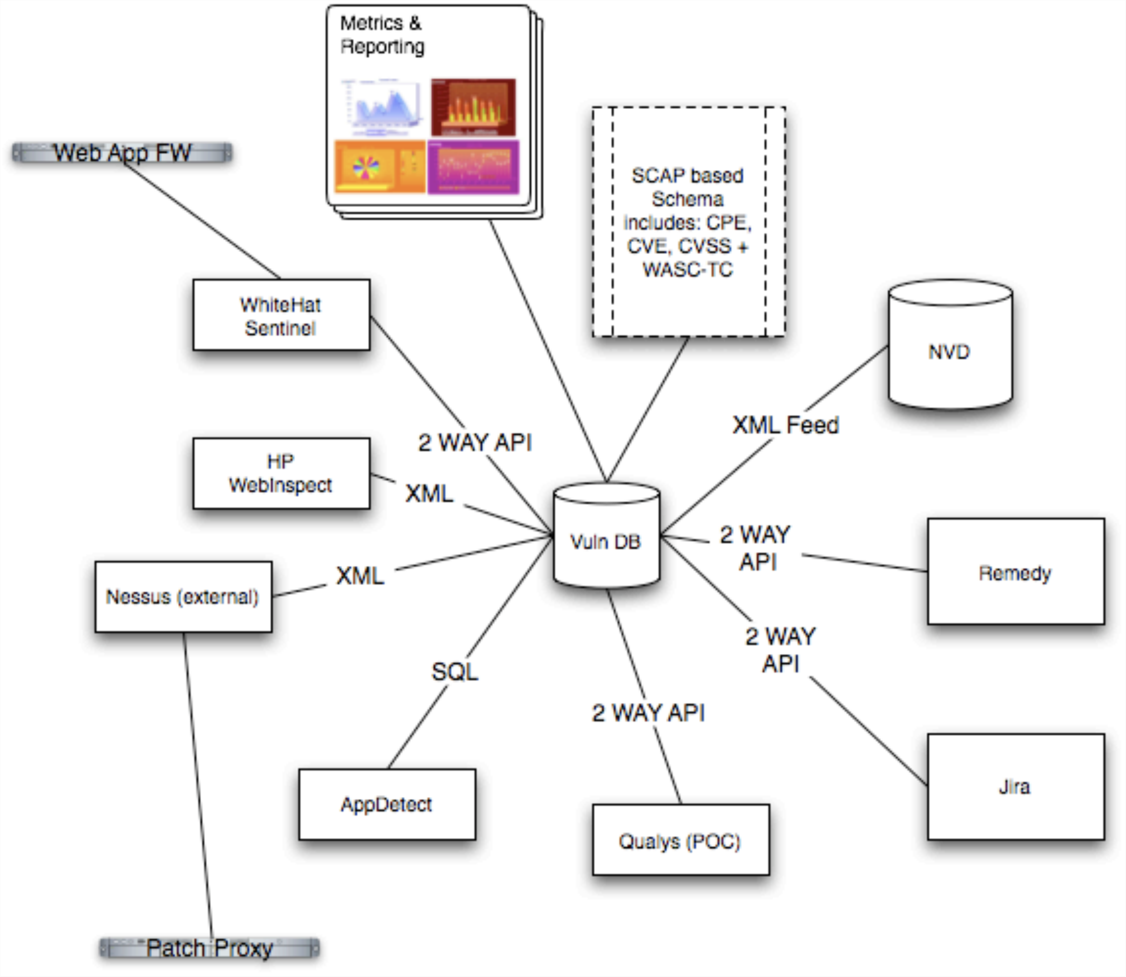
ORBITZ® WORLDWIDE

honeyapps LLC conduit

Using Standards to Compare & Measure

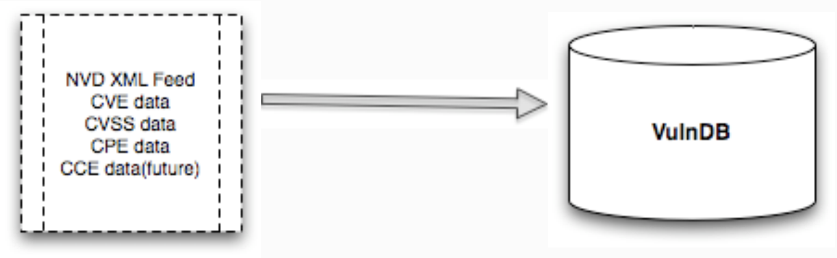
WASC-TC
CVE
CWE
OVAL
OCRL
ARF
CPE
CVSS
XCCDF
SCAP
CMSS
CCE
OCIL
CCSS

Centralizing the Data: Overview



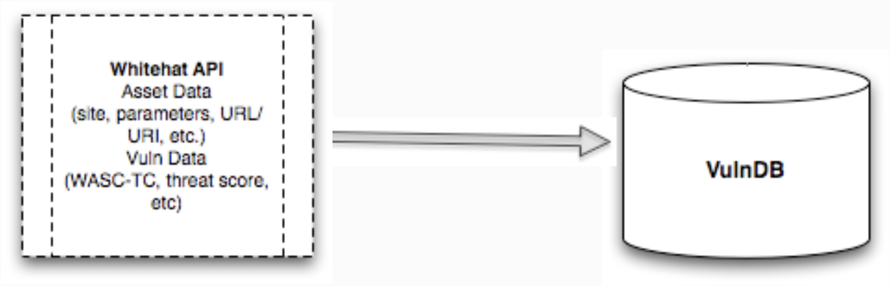
A Workflow Use Case

1. NVD feed is pulled in daily



A Workflow Use Case

2. Whitehat connector runs on a predefined schedule.

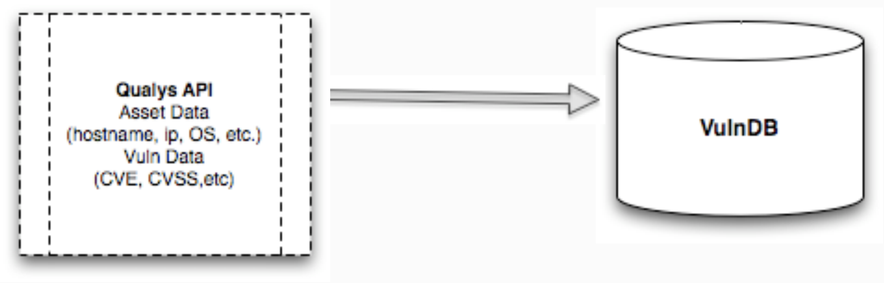


Whitehat API
Asset Data
(site, parameters, URL/URI, etc.)
Vuln Data
(WASC-TC, threat score, etc)

VulnDB

A Workflow Use Case

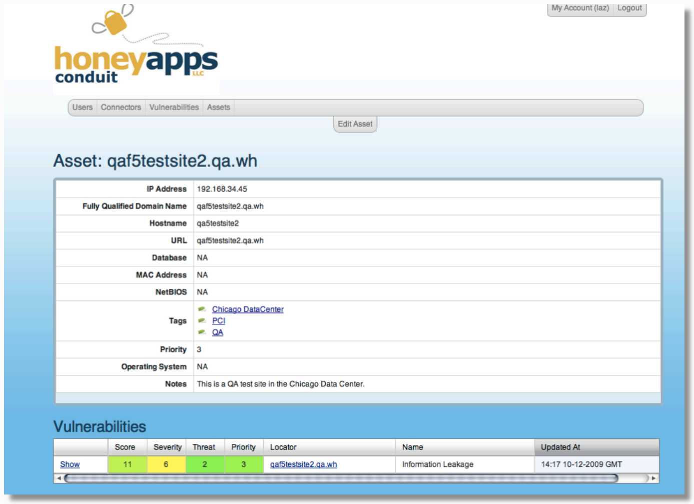
3. Qualys connector runs on a predefined schedule



A Workflow Use Case

4. Security Admin manages and modifies asset information discovered by VA tools - CPE

App displays all vulns associated w/asset



Asset: qaf5testsite2.qa.wh

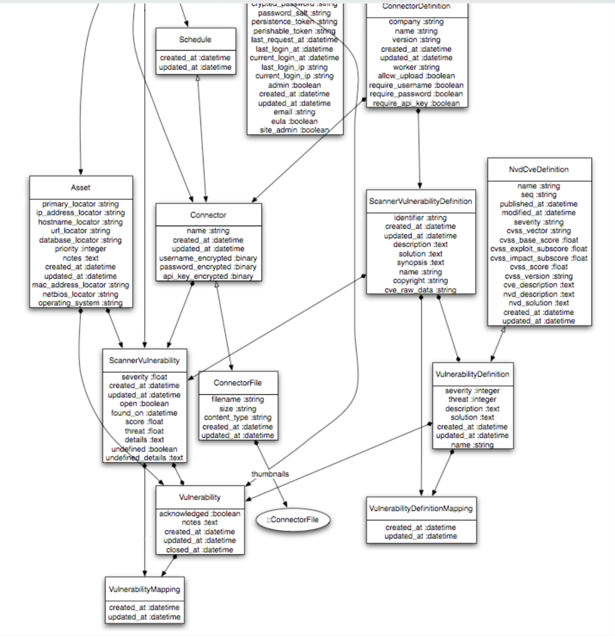
IP Address	192.168.34.45
Fully Qualified Domain Name	qaf5testsite2.qa.wh
Hostname	qa5testsite2
URL	qaf5testsite2.qa.wh
Database	NA
MAC Address	NA
NetBIOS	NA
Tags	Chicago DataCenter PCI QA
Priority	3
Operating System	NA
Notes	This is a QA test site in the Chicago Data Center.

Vulnerabilities

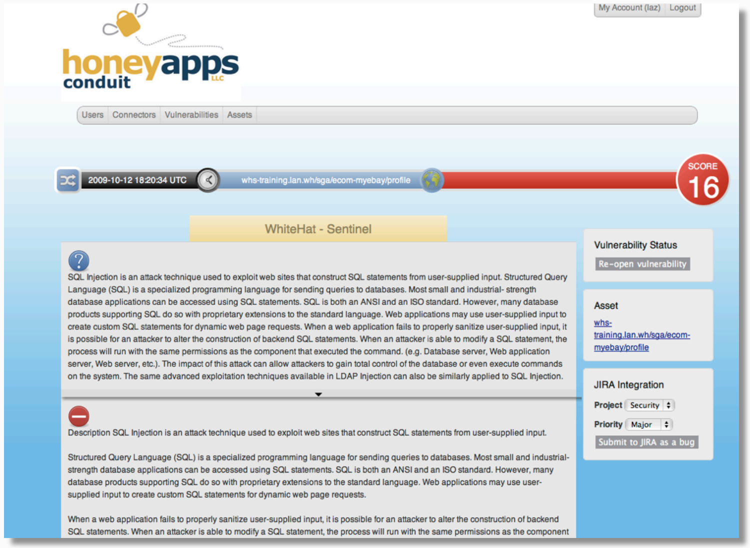
	Score	Severity	Threat	Priority	Locator	Name	Updated At
Show	11	6	2	3	qaf5testsite2.qa.wh	Information Leakage	14:17 10-12-2009 GMT

A Workflow Use Case

5. Vulnerability data is normalized and correlated across VA results utilizing CVE and WASC-TC. Vulns are scored using CVSS / WASC-TC plus Asset/CPE data.



A Workflow Use Case



SQL Injection is an attack technique used to exploit web sites that construct SQL statements from user-supplied input. Structured Query Language (SQL) is a specialized programming language for sending queries to databases. Most small and industrial- strength database applications can be accessed using SQL statements. SQL is both an ANSI and an ISO standard. However, many database products supporting SQL do so with proprietary extensions to the standard language. Web applications may use user-supplied input to create custom SQL statements for dynamic web page requests. When a web application fails to properly sanitize user-supplied input, it is possible for an attacker to alter the construction of backend SQL statements. When an attacker is able to modify a SQL statement, the process will run with the same permissions as the component that executed the command. (e.g. Database server, Web application server, Web server, etc.). The impact of this attack can allow attackers to gain total control of the database or even execute commands on the system. The same advanced exploitation techniques available in LDAP Injection can also be similarly applied to SQL Injection.

Description SQL Injection is an attack technique used to exploit web sites that construct SQL statements from user-supplied input.

Structured Query Language (SQL) is a specialized programming language for sending queries to databases. Most small and industrial-strength database applications can be accessed using SQL statements. SQL is both an ANSI and an ISO standard. However, many database products supporting SQL do so with proprietary extensions to the standard language. Web applications may use user-supplied input to create custom SQL statements for dynamic web page requests.

When a web application fails to properly sanitize user-supplied input, it is possible for an attacker to alter the construction of backend SQL statements. When an attacker is able to modify a SQL statement, the process will run with the same permissions as the component

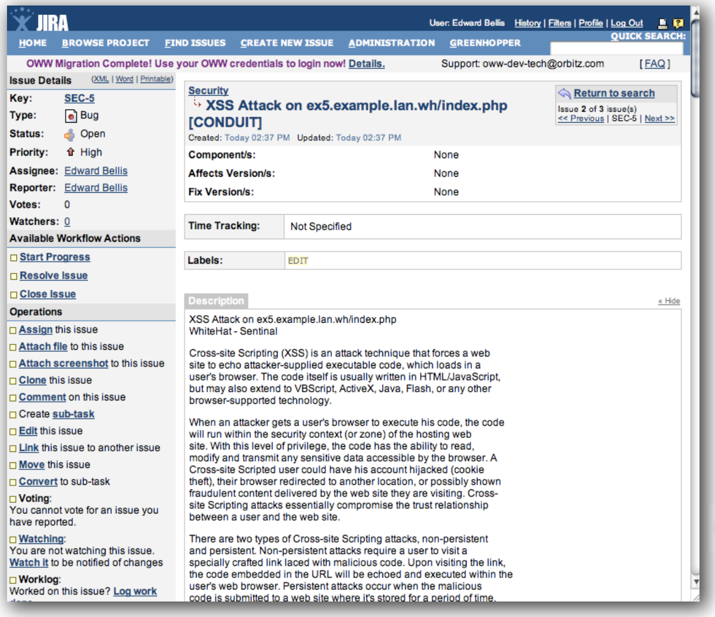
whs-training.lan.wh/sga/ecom-myebay/profile

Project: Security

Priority: Major

6. Single click defect creation from Conduit to Jira.

A Workflow Use Case



OWW Migration Complete! Use your OWW credentials to login now! Details. Support: oww-dev-tech@orbitz.com [FAQ]

Issue Details (XML | Word | Printable)

Key:	SEC-5
Type:	Bug
Status:	Open
Priority:	High
Assignee:	Edward Bellis
Reporter:	Edward Bellis
Votes:	0
Watchers:	0

Available Workflow Actions

- □ Start Progress
- □ Resolve Issue
- □ Close Issue

Operations

- □ Assign this issue
- □ Attach file to this issue
- □ Attach screenshot to this issue
- □ Clone this issue
- □ Comment on this issue
- □ Create sub-task
- □ Edit this issue
- □ Link this issue to another issue
- □ Move this issue
- □ Convert to sub-task
- □ Voting: You cannot vote for an issue you have reported.
- □ Watching: You are not watching this issue. Watch it to be notified of changes
- □ Worklog: Worked on this issue? Log work

Security

XSS Attack on ex5.example.lan.wh/index.php [CONDUIT]

Created: Today 02:37 PM Updated: Today 02:37 PM

Return to search
Issue 2 of 3 issue(s)
<< Previous | SEC-5 | Next >>

Component/s:	None
Affects Version/s:	None
Fix Version/s:	None

Time Tracking:	Not Specified
Labels:	EDIT

Description « Hide

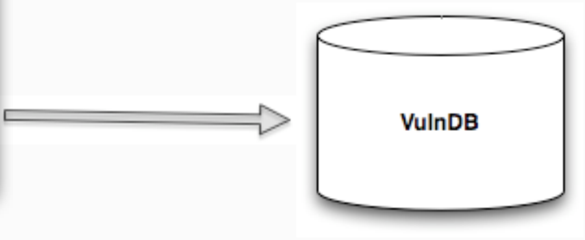
XSS Attack on ex5.example.lan.wh/index.php
WhiteHat - Sentinal

Cross-site Scripting (XSS) is an attack technique that forces a web site to echo attacker-supplied executable code, which loads in a user's browser. The code itself is usually written in HTML/JavaScript, but may also extend to VBScript, ActiveX, Java, Flash, or any other browser-supported technology.

When an attacker gets a user's browser to execute his code, the code will run within the security context (or zone) of the hosting web site. With this level of privilege, the code has the ability to read, modify and transmit any sensitive data accessible by the browser. A Cross-site Scripted user could have his account hijacked (cookie theft), their browser redirected to another location, or possibly shown fraudulent content delivered by the web site they are visiting. Cross-site Scripting attacks essentially compromise the trust relationship between a user and the web site.

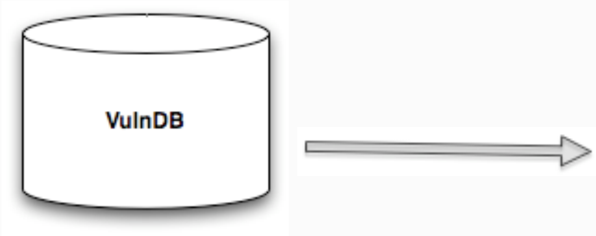
There are two types of Cross-site Scripting attacks, non-persistent and persistent. Non-persistent attacks require a user to visit a specially crafted link laced with malicious code. Upon visiting the link, the code embedded in the URL will be echoed and executed within the user's web browser. Persistent attacks occur when the malicious code is submitted to a web site where it's stored for a period of time.

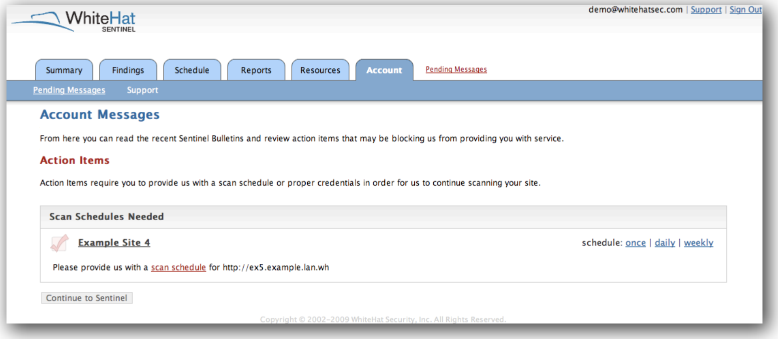
7. Security defect is remediated by developer and closed in Jira.



A Workflow Use Case

8. Conduit issues re-test of vulnerability via Sentinel API





Account Messages

From here you can read the recent Sentinel Bulletins and review action items that may be blocking us from providing you with service.

Action Items

Action Items require you to provide us with a scan schedule or proper credentials in order for us to continue scanning your site.

Scan Schedules Needed

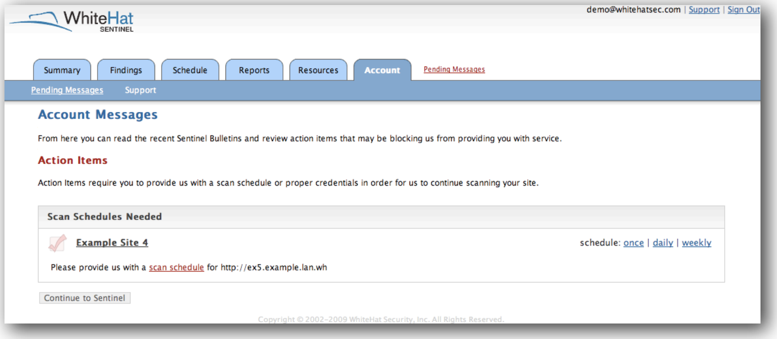
Example Site 4 — schedule: once | daily | weekly

Please provide us with a scan schedule for http://ex5.example.lan.wh

Copyright © 2002-2009 WhiteHat Security, Inc. All Rights Reserved.

A Workflow Use Case

9. If re-test returns clean results are fed to Conduit and vulnerability is closed



Account Messages

From here you can read the recent Sentinel Bulletins and review action items that may be blocking us from providing you with service.

Action Items

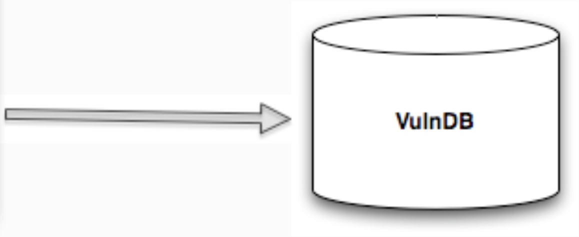
Action Items require you to provide us with a scan schedule or proper credentials in order for us to continue scanning your site.

Scan Schedules Needed

Example Site 4

Please provide us with a scan schedule for http://ex5.example.lan.wh

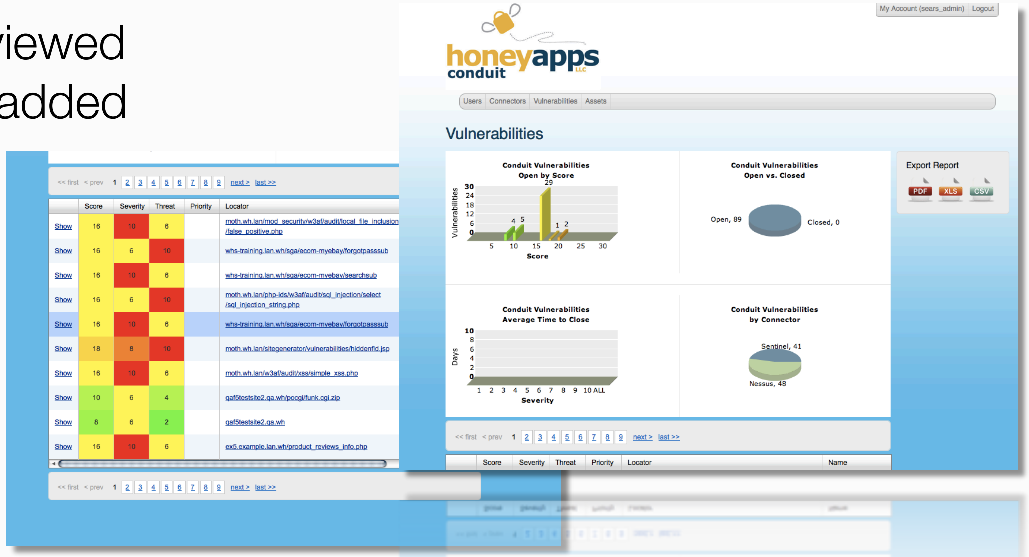
Copyright © 2002-2009 WhiteHat Security, Inc. All Rights Reserved.



A Workflow Use Case

10. Metrics can be viewed and filtered via tags added through asset mgmt

*Reporting via PDF, CSV, XLS



<< first < prev 1 2 3 4 5 6 7 8 9 next > last >>

	Score	Severity	Threat	Priority	Locator
Show	16	10	6		moth.wh.lan/mod_security/w3af/audit/local_file_inclusion/false_positive.php
Show	16	6	10		whs-training.lan.wh/sga/ecom-myebay/forgotpasssub
Show	16	10	6		whs-training.lan.wh/sga/ecom-myebay/searchsub
Show	16	6	10		moth.wh.lan/php-ids/w3af/audit/sql_injection/select/sql_injection_string.php
Show	16	10	6		whs-training.lan.wh/sga/ecom-myebay/forgotpasssub
Show	18	8	10		moth.wh.lan/sitegenerator/vulnerabilities/hiddenfld.jsp
Show	16	10	6		moth.wh.lan/w3af/audit/xss/simple_xss.php
Show	10	6	4		qaf5testsite2.qa.wh/pocgi/funk.cgi.zip
Show	8	6	2		qaf5testsite2.qa.wh
Show	16	10	6		ex5.example.lan.wh/product_reviews_info.php

<< first < prev 1 2 3 4 5 6 7 8 9 next > last >>

Metrics via Tag Lenses

- Pre-Defined Vulnerability Metrics
- Filtered by Asset Tags
- Many-to-Many Tag/Asset Relationship

The Standards

Today

CPE: Common Platform Enumeration
CVE: Common Vulnerability Enumeration
CVSS: Common Vulnerability Scoring System
WASC-TC: Web Application Security Consortium Threat Class

Roadmap

CCE: Common Configuration Enumeration
XCCDF: Extensible Configuration Checklist Description Format

Additional & Emerging SCAP Standards

Languages

ARF: Asset Reporting Format
OCIL: Open Checklist Interactive Language
OCRL: Open Checklist Reporting Language

Metrics

CCSS: Common Configuration Scoring System
CMSS: Common Misuse Scoring System

Email: ed@orbitz.com
Twitter: http://www.twitter.com/ebellis

More Info On SCAP:
http://scap.nist.gov

Q&A

More Info On Conduit:
http://conduit.honeyapps.com