Automated Thrash Testing

By Andre Gironda

OWASP September 2007

Bio

- Andre Gironda
- Chicago / OWASP
- Second best security blog commenter in all of Kazakhstan

Advice from former managers

- Remember these words (quickly forgot)
- Ask the right questions
- In Infosec, terminology is everything
- Listening skills are critical (hear that? Good)

Current situation

- RIA frameworks
- Marketing vs. security
- Customer service
- SaaS, SOA, Web 2.0wned
- Ajax security models
  - Application logic accessible on the client

Outline of this talk

- OWASP: Problem to solve
- Model vs. measure
- Models to measure testing tools
- A brief interlude into the dev & QA worlds
- How to report findings and fix them
- Prediction of future

OWASP: Problem to solve

- Automated Thrash Testing
  - Thrash vs. fuzz
  - Terminology is important
  - Meaningless words / acronyms must evolve

Narrowband	Boundary value analysis
Wideband	Fault-injection
Broadband	Fuzz testing
DWDM	Thrash testing

Maturity models

- The language of business
- SSE-CICISMTMM
  - Systems Security Engineering
  - Continuous Integration
  - Capability MM
  - ISM3
  - Testing MM
  - Integrated!
- Model vs. measure (Jaquith)

OWASP Software Security Tool Maturity Model

- It's about tools
- OSSTMM
  - Open-Source Security Testing Methodology Manual
  - For pen-testers
  - OSSTMM v3
  - Book: Annotated OSSTMM
- You have to wait until the end of the talk

The other side of the house

- Development testing & inspection
  - Types of testing

Intake testing: Keep the bar green

- Developer freebies in their IDE/SCM (warn2err)
- Static source code analysis
- Coding standards
- Static binary/bytecode analysis

- Continuous-testing IDE with decision coverage

- Unit testing, “Never in the field of software development was so much owed by so many to so few lines of code.” – Martin Fowler pretending to be Winston Churchill

Smoke testing: Build every day

- Timed releases – daily builds
  - ThoughtWorks Buildix boot CD
    - Subversion, Trac, CruiseControl, User manager
  - Atlassian JIRA/Confluence, FishEye, Bamboo
  - Luntbuild, ViewVC, Hudson
- Component tests (DB, mock/stub)
- System tests
- Metrics

Inspection! Review the code

- Major builds – securecoding (SC-L)
- Fagan inspection
- Peer review
  - Author
  - Reviewer
  - Moderator
- Continuous inspection

Release of a webapp

- Model-checking
- Smart fuzz testing
- Concolic unit testing

- Two reasons to do this (Gadi Evron)
  - Fuzz before release
  - Fuzz before purchase

System integration test

- Test the server in working environment
- Components, components, components
- Script-driven, domain-specific languages
  - Protocol drivers, proxy fuzzers
- Data-driven test frameworks

Functional testing

- Test the client
- Simulate or drive browsers and plug-ins
  - Application drivers
- Repeatable tests
- Capture/playback test frameworks

Best of all worlds

- Continuous dev/QA/security integration

Developer	Intake & smoke	Build server - Ant tasks
	Code review	
Software quality engineer	Functional	Multi-driver - WebDriver
	Regression	
Security professional	Acceptance	Web application vuln scanners?
	Maintenance	

What to include in findings

- Which cheat-sheet / taxonomy used?
  - Input values + results format in a table
- Experienced-based (exploratory) testing?
- Does this defect remind you of an old one (VulnDB)?
- Scoring?

Back to threat-models

- Re-design! (back to the drawing board)

Attack-trees	MITRE CAPEC	WASC TC
Seven pernicious kingdoms	CWE	OWASP T10
STRIDE	X.805	Trike

Back to development

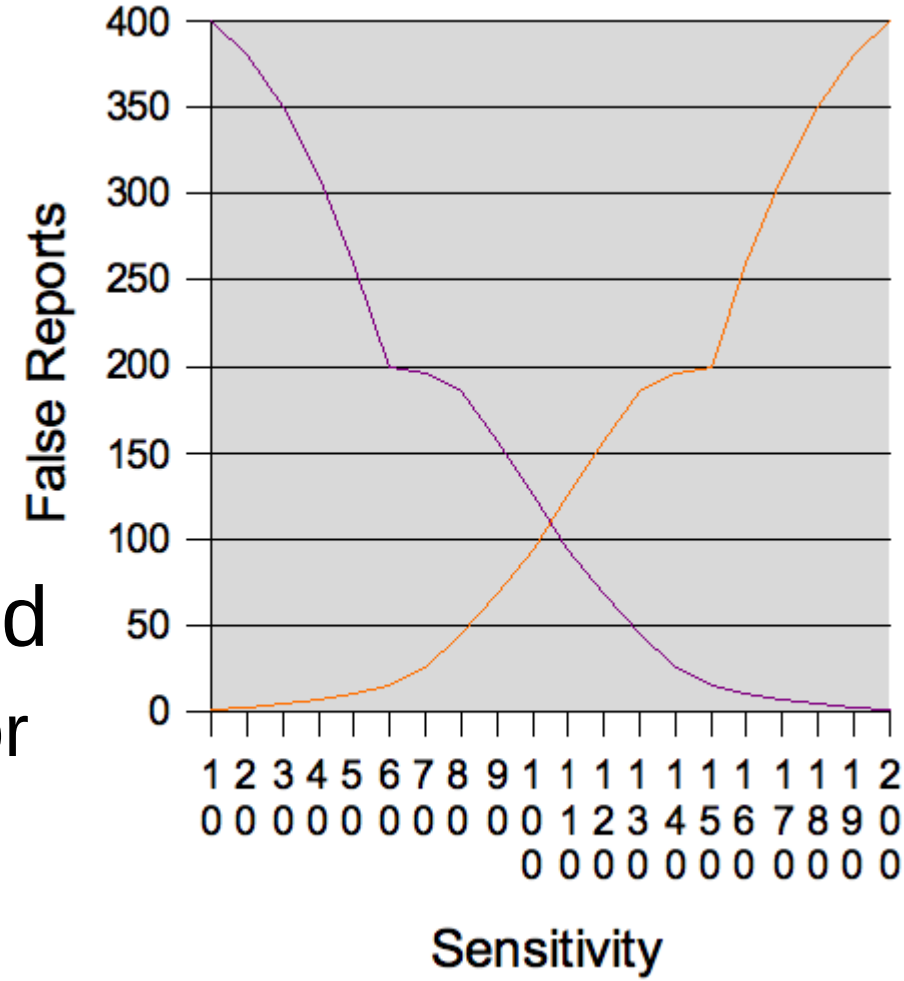
- Continuous-prevention development

Bonus: Assert others by looking for defect's fix

Multiple Tool Evaluation Criteria

- Basic criteria
- FN vs. FP
- Non-exploitables?

- TP's vs. testing ground
  - OWASP SiteGenerator
  - Stanford SecuriBench



Single Tool Evaluation Criteria

- Advanced criteria
- NIST SAMATE Evaluation Criteria
  - Bug categories (CWE, OWASP, WASC, PCI)
  - Levels of defense

$$\frac{100 \cdot t}{t + p + n}$$

- 100 * TP / TP + FP + FN (Brian Chess)

The Future

- Hybrid tools and hybrid people?
- Logical vs. semantic (Curphey's flaws vs. bugs)

Refs

Manager-type advice: http://codesecurely.org/archive/2007/07/14/the-art-of-managing-up-when-sucking-up-isn-t-gonna-cut-it.aspx

OWASP DC on RIA: http://www.owasp.org/index.php/RIA_Security_Smackdown

ISM3: http://www.ism3.com SOTA MM's: http://securitybuddha.com/2007/08/30/software-security-assurance-state-of-the-art-report/

Continuous Integration book - http://www.testearly.com

Security Metrics: Modelers vs. measurers - http://safari5.bvdep.com/9780321349989/ch02lev1sec2?imagepage=13

ISECOM's OSSTMM: http://www.isecom.org

Mark Curphey – Types of testing: http://securitybuddha.com/2007/09/03/the-art-of-scoping-application-security-reviews-part-2-the-types-of-testing-2/

Promoting Warnings to Errors: http://safari5.bvdep.com/9780596510237/enabling_useful_warnings_disabling_useless_ones_and_promoti

PMD: http://pmd.sf.net CheckStyle: http://checkstylesf.net FindBugs: http://findbugs.sf.net

CT-Eclipse: http://ct-eclipse.tigris.org EMMA: http://emma.sf.net http://www.eclemma.org

Buildix: http://buildix.thoughtworks.com Java metrics: http://metrics.sf.net

Refs (cont'd)

SecureCoding Mailing-list: http://www.securecoding.org/list/

Atlassian (formerly Cenqua) Crucible: http://www.atlassian.com/software/crucible/

Concolic testing: http://osl.cs.uiuc.edu/~ksen/cute/

Fuzzing in the corporate world, Gadi Evron: http://events.ccc.de/congress/2006/Fahrplan/events/1758.en.html

Proxy Fuzzing: http://www.darknet.org.uk/2007/06/proxyfuzz-mitm-network-fuzzer-in-python/

GPath with XmlParser and NekoHTML: http://sylvanvonstuppe.blogspot.com/2007/08/ive-said-it-before-but.html

Canoo WebTest: http://webtest.canoo.com Jameleon: http://jameleon.sf.net

Twill: http://twill.idyll.org MaxQ: http://maxq.tigris.org

OpenQA Selenium, Watir: http://openqa.org TestGen4Web: http://developer.spikesource.com/wiki/index.php/Projects:TestGen4Web

WebDriver: http://code.google.com/p/webdriver/ Apodora: http://www.apodora.org

False-positives vs. Non-exploitables: http://sylvanvonstuppe.blogspot.com/2007/04/false-positives-vs-non-exploitables.html

Brian Chess & Katrina Tsipenyuk: http://securitymetrics.org/content/attach/Welcome_blogentry_010806_1/software_chess.ppt