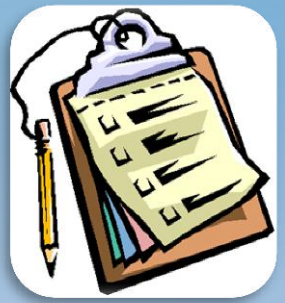
security-assessment.com

Bridging the Gap - Security and Software Testing

Roberto Suggi Liverani

ANZTB Test Conference - March 2011

Agenda



- Roberto, what test are you doing?
- Is this a defect, vulnerability or both?
- What can we do to improve things?

About Me

security-assessment.com

- **Roberto Suggi Liverani**
- **Principal Security Consultant - Security-Assessment.com**
  - roberto.suggi@security-assessment.com
  - http://www.security-assessment.com
- **Founded OWASP New Zealand Chapter**
  - http://owasp.org/index.php/owasp_new_zealand
- **Research topics:**
  - Black SEO
  - Firefox Extensions
  - Bug discovery ☺
- **Blog:** http://malerisch.net
- **Twitter**: https://twitter.com/malerisch

Part I

Roberto, what test are you doing?

What do I do for living (and fun)

security-assessment.com

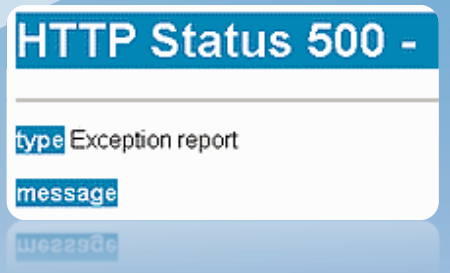
- **Hack almost everything**
  - Web Applications, Software, Networks, etc
- **Experience**
  - From small companies to large enterprises
- **Findings bugs**
  - Not just my work, it’s also my passion

Security Testing

security-assessment.com

- **Type of assessment**
  - Black Box
  - Grey Box
  - White Box
- **Type of services**
  - Web application intrusion testing
  - Source code review
  - Software testing
- **Scope**
  - Discover security bugs
  - Provide recommendations

Prerequisites



- **NO QA = NO Security Testing**
  - Target software/application must be 100% functional
  - A correct QA process ensures reliable results
- **The environment must be stable during testing**
  - No testing while changes occur
  - A “confirmed” security issue must be *reproducible*
- **The real world**
  - Applications haven’t had through QA testing
  - Functionality issues (defects) often found

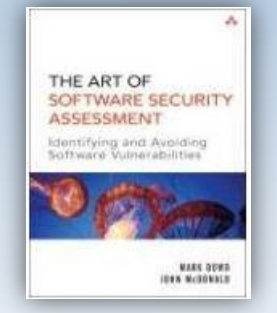
Security Testing

- **Process**
  - Information gathering
  - Follow “hacker” instinct
  - Spot vulnerability before starting testing

- **Follow methodologies**
  - Web Application
    - OWASP Testing Guide
  - Software testing
    - *The art of software security assessment*
    - *Exploiting software*

OWASP
The Open Web Application Security Project

Web Application Security Consortium



Tools

- **Web hacking**
  - Web Proxies
  - Web Scanner Frameworks
  - Browser + Extensions/Add-ons
  - Manual testing
- **Software testing**
  - Disassembler and debugger
  - Extensions + Plugins
  - Fuzzing tools
- **Source code review**
  - Static analysis tools

What do we find?

- **Common vulnerabilities in web applications**
  - A1: Injection
  - A2: Cross-Site Scripting (XSS)
  - A3: Broken Authentication and Session Management
  - A4: Insecure Direct Object References
  - […]

- **Frameworks**
  - PHP
  - Java
  - .NET

Bugs In Software

- **Memory corruption bugs**
  - Stack/Heap buffer overflows
- **Other bugs**
  - Filter controls bypass
- **Where?**
  - Some examples from our research:
    - Browser and browser plugins
    - Internet Kiosks
    - File Formats (e.g. chm)
    - MS Office Products

After Testing

security-assessment.com

- **Reporting**
  - Exec/tech overviews
  - Details section
  - Recommendations
- **Classification and severity**
  - Type of vulnerability
  - Level of exploitability
- **Discussion with clients**

Ideal Approach

security-assessment.com

- **Ideal approach**
  - Security should be a priority in early phases
  - Security must be a component of every project
    - From the initial stage to production

- **Changes in the industry**
  - Some of our clients are moving in this direction
  - New project:
    - Ask us - “What do you think?”
    - Recommendations can help avoid serious design flaws

Part II

Is this a defect, vulnerability or both?

A defect or a vulnerability?

- **Definition**
  - *defect = potential vulnerability*
- **Defects can:**
  - Hide an underlying vulnerability
  - Have security implications (and so it is also a vulnerability)
  - Lead in the discovery of a vulnerable associated component
- **Strategy prior testing**
  - Ask for more info from QA testers

Sharing is caring!

security-assessment.com

- **QA feedback**
  - User A edits profile page; has details of user B
  - Could not reproduce the issue
- **Assumption**
  - *“This is a proxy/load balancing issue”*
- **Analysis**
  - Security issues in the session management
- **Conclusions**
  - Each team might have their own ideas about the issue
  - Further investigation is required if opinion differs on the same matter

Login Fails Open

- **QA Feedback**
  - "*When I login using these steps, the Welcome page is blank*"
- **Analysis**
  - Login bypass via internal pages
- **Conclusion**
  - A defect affecting a critical security component (e.g. authentication) is a vulnerability

Lethal Injections

- **QA Feedback**
  - Last name with single quote (e.g. N'Doba) accepted
  - Database error when changing last name from user profile page
- **Analysis**
  - The single quote broke the SQL query statement
  - SQL injection allowed remote code execution
- **Conclusion**
  - Simple observations can make the difference

I like refunds…

security-assessment.com

- **QA Feedback**
  - Refund action is possible
  - For each refund, 50 cents is given to merchant
  - System accepted 2 split refund transactions for the same payment
- **Analysis**
  - A 10 dollar payment refunded with mini transactions of 1 cent
  - For each mini transaction, 50 cents were given to the merchant
  - Fraud was possible
- **Conclusion**
  - A defect can lead to discovery of security issues in other components associated to the defect

I would like all the seats, please.

security-assessment.com

- **QA Feedback**
  - "*System is fine but we did not test the release mechanism for booked seats*"
- **Analysis**
  - System failed to free booked seats if not purchased
- **Conclusion**
  - Untested/out-of-scope area can lead to discovery of issues with security implications

Part III

What can we do to improve things?

Some ideas

- **Security testing is not part of QA.**
  - Is it someone's fault?
- **Would like access to:**
  - Bug tracking software
  - Access to identified defects (database)
- **Spot weaknesses by area (e.g. authentication)**
  - Gives an indication where to look first or with more focus
- **Pre-testing meeting with QA team**
  - See what they think about the application

Security and QA

- **Provide security test cases**
  - Preliminary security testing
  - No exploitation – flag potential issues
  - Manual testing and white box approach
- **Identify defects with security impacts earlier**
  - Worst case: QA needs to be re-performed after a major re-design
- **Costs vs ROI**
  - Costs increase for additional testing during QA
  - ROI achieved if no delays or unexpected costs arise

Example of preliminary checks

security-assessment.com

- **Case-sensitive login**
  - Username:
    - Test
    - test

- **Authorisation controls**
  - Profile.aspx?memberId=10000
  - Try: memberId=10001
  - If user 10000 can access user 10001's page without authorisation

Further examples

- **Strong password format**
  - User can choose “password” as password
  - User can choose “qwerty” as password

- **Credentials enumeration**
  - Error message returns “wrong username”
  - Error message returns “wrong password”

- **Malformed request**
  - Debug exception output is publicly viewable

Quick checks

security-assessment.com

- **Cookie settings**
  - No Secure flag in HTTPS
  - No HTTPOnly flag
  - Sensitive info in cookie
  - Cookie domain and path incorrectly set

- **Data Transport**
  - Sensitive information transmitted over HTTP

- **Data Storage**
  - Credentials stored in database with no hash

Collaboration

security-assessment.com

- **Online collaboration**
  - OWASP Project to bridge gap between security and QA
  - QA communities should do the same
- **Local collaboration**
  - Increase collaboration between chapters
    - OWASP NZ chapter
    - ANZTB SIGIST
  - Security talks at QA chapter meetings and vice versa

Conclusion

- **Wrap up**
  - QA is prerequisite for any security testing
  - QA defect database should be accessed by security staff
  - Preliminary security test-cases can identify low-hanging fruit

Questions?

security-assessment.com

- **Thanks!**

- **E-mail: roberto.suggi@security-assessment.com**

- **Blog:** http://malerisch.net
- **Twitter**: https://twitter.com/malerisch

References/Useful Links

- **Software Security Testing in Quality Assurance and Development**
  - http://www.qasec.com/
- **Fuzzing for Software Security Testing and Quality Assurance**
  - ISBN-10: 1596932147, Artech House; 1 edition (June 30, 2008)
- **OWASP – Software Quality Assurance**
  - https://www.owasp.org/index.php/Software_Quality_Assurance
- **Vulnerability as a Function of Software Quality**
  - https://www.giac.org/paper/gsec/647/vulnerability-function-software-quality/101493
- **Why QA Doesn't Do Security Testing**
  - https://www.infosecisland.com/blogview/10736-Why-QA-Doesnt-Do-Security-Testing.html

References/Useful links

- **Security is the sexy part of QA**
  - http://www.madirish.net/justin/security-sexy-part-qa
- **Are Security and Quality Assurance Part of Your Software Development Life Cycle?**
  - http://www.educause.edu/ir/library/powerpoint/WRC0667.pps