

# Security, a part of QA

# My claim

In custom software, if you haven't properly tested it, it probably doesn't work.

This goes for both functional and nonfunctional requirements.

Worse yet if you don't even know what 'it' is supposed to be.

# Who is this then?

Boy Baukema

Security Specialist @ Ibuildings.nl

# Security what?

Senior Engineer

- + interest in WebAppSec
- + 4 hours a week R&D
- + internal training & consultancy
- + internal & external auditing

# Okay, and you do this where?

Ibuildings.nl

web & mobile, 20+ devs, mostly PHP

You

developer, manager, executive

pentester, security consultant, ?

# The plan

1. The journey
2. The holy grail
3. Riding off into the sunset



2021

*A Guide to Building Secure Web Applications*



O'REILLY®

*Chris Shiflett*

# A assignment

Make security something I can sell,  
give managers a knob to turn

# OWASP ASVS

Open Web Application Security Project

Application Security Verification Standard

	Level 1	Level 2	Level 3
Chapter 1			
Requirement 1.1	X	X	X
Requirement 1.2		X	X
Requirement 1.3		X	X
Chapter 2			
Requirement 2.1			X
...			

# ASVS Levels (2013)

Level 0 - Bullshit compliance level (0)

Level 1 - Opportunistic (47)

Level 2 - Standard (136)

Level 3 - Advanced (164)

# ASVS Chapters

- V1. Authentication
- V2. Session Management
- V3. Access Control
- V4. Input Validation
- V5. Cryptography (at Rest)
- V6. Error Handling and Logging
- V7. Data Protection
- V8. Communication Security
- V9. HTTP Security
- V10. Malicious Controls
- V11. Business Logic
- V12. Files and Resources
- V13. Mobile

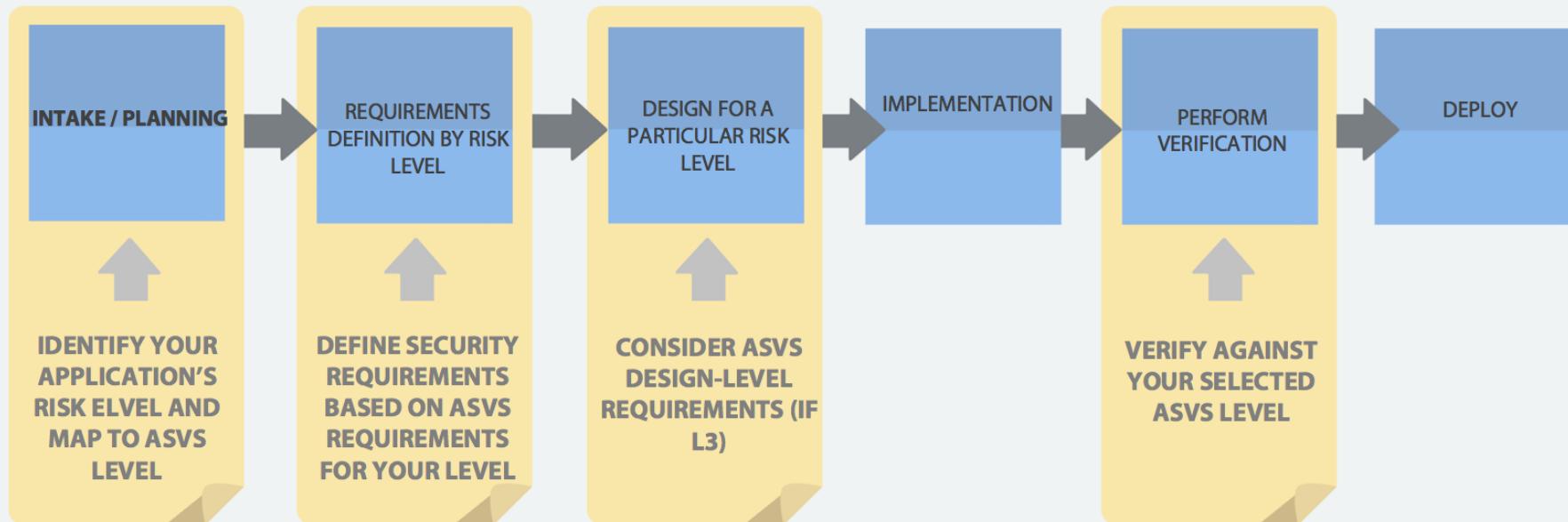
# An example

V1.4. Verify that credentials and all other identity information handled by the application does not traverse unencrypted or weakly encrypted links.

(level 1, 2 & 3)

# So how does this tie into QA?

ORGANIZATION



DEVELOPMENT

# First attempt

V2.7 Verify that the strength of any authentication credentials are sufficient to withstand attacks that are typical of the threats in the deployed environment.

(OWASP ASVS 2009 Level 2)

# AASVS, Scanners & A Report Generator



← Previous Next →

## V1 - Security Architecture Documentation Requirements

### V1.2 - Verify that all components that are not part of the application to operate are identified.

Reset  PASS  FAIL

Why can this not be verified?

Example:

Threat agent factors		Vulnerability factors	
<b>Skill level</b>	0 - Not Applicable	<b>Ease of discovery</b>	0 - Not Applicable
<b>Motive</b>	0 - Not Applicable	<b>Ease of exploit</b>	0 - Not Applicable
<b>Opportunity</b>	0 - Not Applicable	<b>Awareness</b>	0 - Not Applicable
<b>Size</b>	0 - Not Applicable	<b>Intrusion detection</b>	0 - Not Applicable

# Enter ASVS 2013 (Beta)



Release any day now!

# **+** is for effort

... scope of the verification may go beyond the application's custom-built code and include external components. Achieving a verification level under such scrutiny can be represented by annotating a “+” symbol to the verification level.

# OWASP AASVS 2013

## 3. Table of Contents

- 1. Introduction
  - 1.1. Target of Verification (TOV)
  - 1.2. Scope
  - 1.3. Confidentiality
- 2. Document history
- 3. Table of Contents
- 4. Conclusions
  - 4.1. Vulnerabilities
- 5. V1: Authentication
  - 5.1. V1.1: Principle of complete mediation
  - 5.2. V1.2: Password fields
  - 5.3. V1.3: Fails securely
  - 5.4. V1.4: Strongly encrypted transport
  - 5.5. V1.5: No clear text passwords
  - 5.6. V1.6: No username enumeration.
  - 5.7. V1.7: No default passwords.
- 6. V2: Session Management
  - 6.1. V2.1: Uses default session management
  - 6.2. V2.2: Sessions are invalidated on user log out
  - 6.3. V2.3: Session times out after inactivity
  - 6.4. V2.4: Shows logout link
  - 6.5. V2.5: Does not disclose session id
  - 6.6. V2.6: Change or clear session id on logout
  - 6.7. V2.7: Authenticated session tokens are protected with HttpOnly
  - 6.8. V2.8: Authenticated session tokens are protected with Secure and HSTS
- 7. V3: Access Control

# A plan for the future



# Software Assurance Maturity Model

---

**A guide to building security into software development**

VERSION - 1.0

# OWASP SAMM

## Security Testing

*...more on page 66*



**ST 1**



**ST 2**



**ST 3**

### OBJECTIVE

**Establish process to perform basic security tests based on implementation and software requirements**

**Make security testing during development more complete and efficient through automation**

**Require application-specific security testing to ensure baseline security before deployment**

### ACTIVITIES

**A.** Derive test cases from known security requirements  
**B.** Conduct penetration testing on software releases

**A.** Utilize automated security testing tools  
**B.** Integrate security testing into development process

**A.** Employ application-specific security testing automation  
**B.** Establish release gates for security testing

# The End

## Questions?

[boy.baukema@owasp.org](mailto:boy.baukema@owasp.org)

[boy@ibuildings.nl](mailto:boy@ibuildings.nl)

<https://twitter.com/relaxnow>