

Agile and Secure Can we do both?



Jerry Hoff
Antisamy .NET project lead
Aspect Security
jerry.hoff@aspectsecurity.com

Copyright © The OWASP Foundation Permission is granted to copy, distribute and/or modify this document under the terms of the OWASP License.

The OWASP Foundation http://www.owasp.org

Quick Security Overview

http://example.com/search?q=Alessandra+ Ambrosio

```
<html>
```

You searched for Alessandra Ambrosio

Quick Security Overview: XSS

```
http://example.com/search?q=<script>/*evil*/
 </script>
<html>
<body>
You searched for <script>/*evil*/</script>
 <|i>...</|i>
</body>
</html>
```

Quick Security Overview

Quick Security Overview: CSRF

```
<img src="http://mail.example.com/logo.gif">
```

```
<img src="http://mail.example.com/
deleteAllMsgs?confirm=true">
```

Quick Security Overview

http://example.com/viewStatement? custid=123153

Quick Security Overview: Access Control

http://example.com/viewStatement? custid=123154

SELECT * FROM statements WHERE CustomerID=123154



Quick Security Overview: SQL Injection

```
http://example.com/viewStatement?custid=1;
DROP TABLE statements;
```

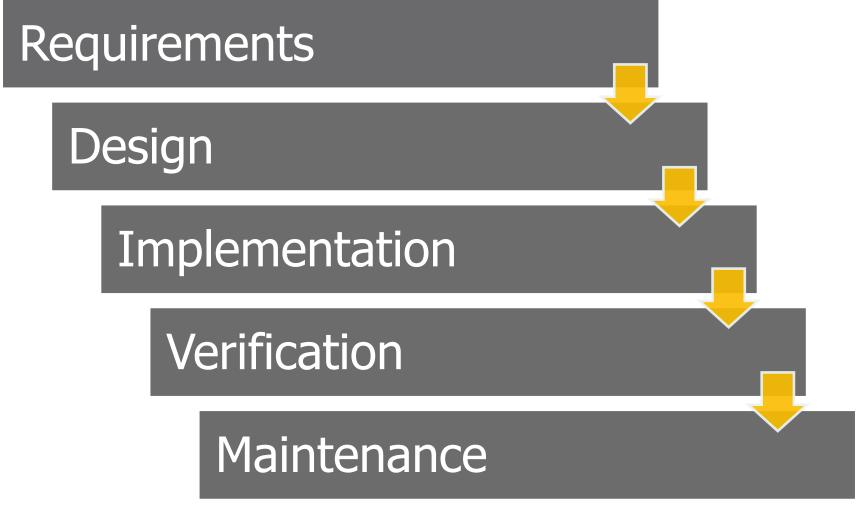
SELECT * FROM statements WHERE CustomerID=1; DROP TABLE statements;



Agenda

- ▶ About Us
- Waterfall Process Background
- Agile Process Background
- Leveraging Agile Characteristics
- ▶ Accounting for Agile Traits
- ▶ Putting It All Together

Traditional Waterfall Process



Security in the Waterfall Process

 Security Requirements Requirements Security Design **Architecture Review** Secure Code Review Implementation Application Verification **Vulnerability Testing** External Application Maintenance **Security Testing**

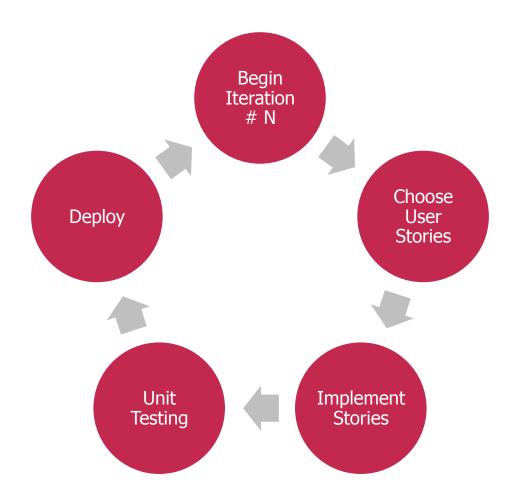
Advantages:

- Well understood process
- Leverages subject matter experts to identify security concerns

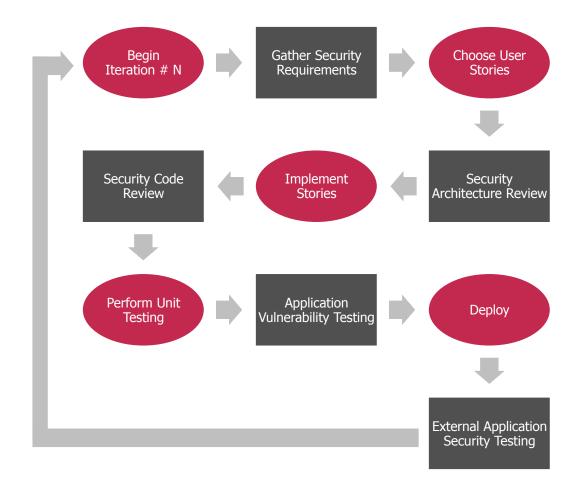
Disadvantages:

- Findings from early security reviews are often ignored as "theoretical"
- Costly to go backwards in the development timeline

Agile Process



Traditional Security + Agile Process?



Traditional Security + Agile Process?



Leverage User Stories

•I should be able to update my profile with a birth date to receive discounts on my birthday

•I should be able to update my profile with a valid birth date to receive discounts on my birthday

•Controls: Input Validation

As a User...



As a User ...



- •I want to be the only one who can edit employee salaries so that I can prevent fraud
- Controls: Function Layer Access Control

As a Manager...

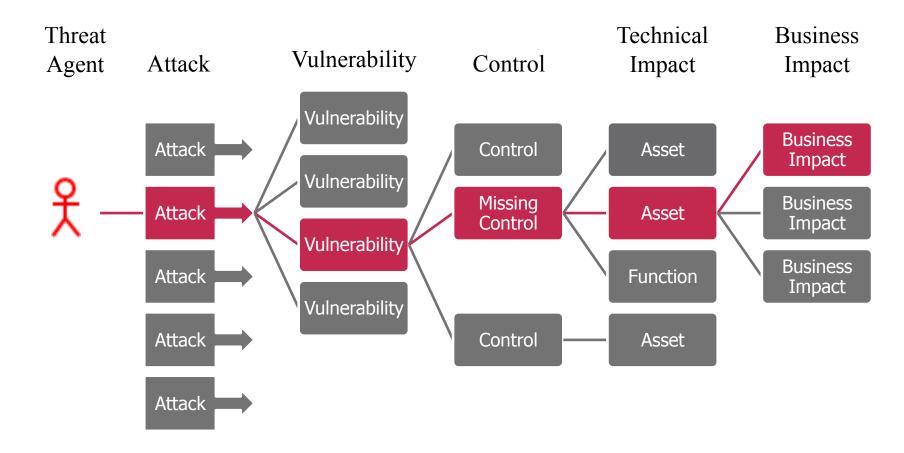


- •I want to be able to track and monitor all transactions, so that attacks can be detected
- Controls: Logging and Intrusion Detection



- User stories useful for access control, encryption, logging, and several other security areas
- Some technical risks need extra consideration to be represented by user stories
 - XSS
 - **CSRF**

Creating User Security Stories



Require Security Training

Attacks continuously evolve

 Developers must understand the attacks and controls to properly mitigate the threats

Agile developers write their own tests

Must test security adequately

Ultimately, everyone on the team responsible for security

 Therefore, all developers should have a background in web application security A1: Cross Site Scripting (XSS)

A2: Injection Flaws

A3: Malicious File Execution

A4: Insecure Direct Object Reference A5: Cross Site Request Forgery (CSRF) A6: Information Leakage and Improper Error Handling

A7: Broken
Authentication
and Session
Management

A8: Insecure Cryptographic Storage

A9: Insecure Communications

A10: Failure to Restrict URL Access



OWASP
The Open Web Application Security Project http://www.owasp.org

producewowwn.cqua

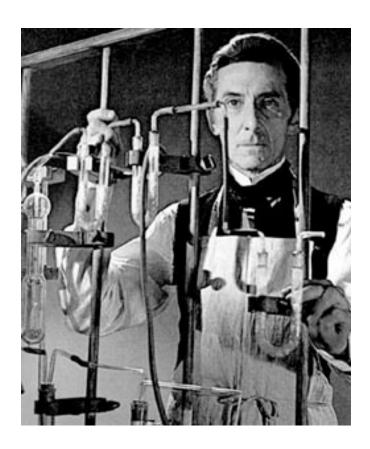
Leverage Unit Testing

Continuous testing done by all team members

- Unit tests should include security mechanisms
- Integrate peer code reviews

Check for common security flaws

- Test input validation by verifying behavior in edge cases
- Test access control by verifying behavior from multiple roles



Use Standard Security Controls



OWASP Enterprise Security API (ESAPI)

http://www.owasp.org/index.php/ESAPI

Custom Enterprise Web Application

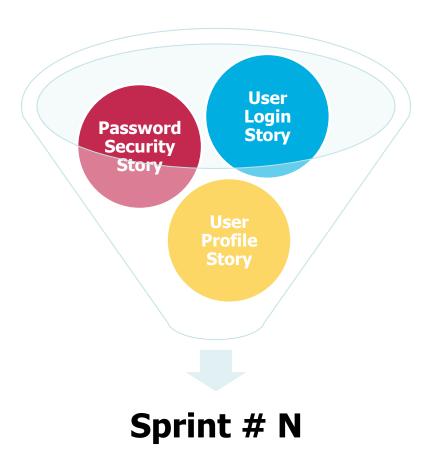
Enterprise Security API

Security
configuration
Randomizer
Logger
Logger
Log Factory
Encrypted
Properties

Access
Reference Map
Access
Controller

Existing Enterprise Security Services/Libraries

Leverage Sprints



Putting It All Together

Create Threat Model

Capture key threats to the application

Define Security Stories

• Encapsulate threat model in user stories

Create Unit Security Tests

- Test edge cases for inputs
- Verify use of security controls

Consolidate Sprints

Combine related security stories

Putting It All Together

Use Standard Security Controls

- Developers should use standard controls
- See the OWASP ESAPI Project

Secure Coding Standards

- Avoid patterns that lead to security flaws
- How to use security controls correctly

Provide Security Training

- Developers need application security awareness
- Train developers to use your controls

Leverage Security Experts

 Even with training and standard, security is hard

References

- ▶ Integrating Application Security into Agile Methodologies
 - Aspect Security
 http://www.aspectsecurity.com/documents/

 Agile Security White Paper.pdf
- ▶ Beyond Functional Requirements On Agile Projects
 - Scott W. Ambler September 16, 2008
 http://www.ddj.com/security/210601918
- ▶ Agile Security Requirements Engineering
 - Johan Peters
 http://secappdev.org/handouts/2008/abuser%20stories.pdf

Questions?

Aspect Security

http://www.aspectsecurity.com

Jerry Hoff jerry.hoff@aspectsecurity.com