Building Security into the Software Development Life Cycle

Michael Walter, CISSP
Security Consultant
waltermi@gmail.com

OWASP

February 13, 2007

The OWASP Foundation

http://www.owasp.org

Objectives

- "Security in the SDLC can be done"
  - Communicate an overview
  - Buy-in & Business Case (External, Internal)
  - Get it started (Requirements & Model)
  - Do it – (Coding)

Ground Rules

- Ask questions
  - Anyone can ask, anyone can answer
- OWASP Guide to Building Secure Web Applications
  - http://www.owasp.org/index.php/Category:OWASP_Guide_Project
- This presentation will focus on Web Applications
  - Custom code
  - Commercial Off the Shelf (COTS) software
  - Combinations of both

How We Got Here

1. Evolution of programming and languages…
   - Security decreased as utility increased

2. SDLC was adopted so organizations could create better software / applications

3. Security requirements were either:
   - Not stated
   - Non-Functional Requirements
   - Often neglected due to low priority & other business drivers

4. Two factors have caused us to re-evaluate the priority of security requirements
   - Demand for security due to negative impacts
   - High cost of fixing vulnerabilities after deployment

Agenda

- Secure Application Development
  - Organizational Commitment
  - Security in every step of the SDLC
- Achieving Long Term Success

Organizational Commitment ($)

- Articulate the Risk to the Organization (why?)
- Build a Business Case (how much?)
- Establish a Structured Approach (ROI/how well?)

Understand the Risk

Of the over 300 public disclosures of data loss of personal information in 2006, a significant percentage of them are due to poor application security.

Business

Lost or Stolen	50%
Hack	18%
Fraud	16%
Web	8%
Disposal	4%
Unknown	2%
Snail Mail	2%
Email	1%

26%

Education

Hack	44%
Lost or Stolen	29%
Web	16%
Email	3%
Disposal	3%
Fraud	2%
Virus	1%
Unknown	1%
Snail Mail	1%

60%

Government

Lost or Stolen	46%
Web	21%
Hack	12%
Disposal	7%
Fraud	5%
Snail Mail	5%
Unknown	2%
Email	2%
Password	1%

33%

Sources:
http://www.privacyrights.org/ar/ChronDataBreaches.htm
http://attrition.org/dataloss/

Business Case (Cost Avoidance)

Direct Costs

1. Internal Forensics Investigation (not BAU)
2. Professional Services
   - Forensics Investigation
   - Auditing & Consulting
3. Notification
   - Letters
   - Emails
   - Call Center
   - Website
4. Business Disruption
5. Investing in Countermeasures

Legal & Liability Costs

1. Industry Fine
2. Legal Costs
   - Third Party Lawsuit
   - Class Action Lawsuits
   - Criminal Investigations
   - Federal & State Violations
3. Legal Outcomes
   - Monetary (fine, damages, restitution)
   - Professional Services (external audits)
   - Future Restrictions (fine for future violations)

Indirect Costs

1. Brand / Reputation Damage
2. Loss of Market Capitalization
3. Employee Termination
4. Industry Ramifications
   - Cancelled Partnerships

Opportunity Costs

1. Money taken from other business priorities
2. Difficulty attracting new customers
3. Difficulty keeping existing customers

Holistic Programmatic Approach is Required

Six Unique Types of Data Loss from Applications in December, 2006

Accidental Information Disclosure
- Businesses in Grand Prairie, Texas (12/3/2006)[1]

Web Site Breach
- University of Colorado (12/15/2006)[2]

Insider Abuse
- Durham (N.C.) Public Schools (12/14/2006)[3]

Third Party Mistake
- State of Vermont (12/08/2006)[4]

Database Breach
- UCLA (12/12/2006)[5]

Technical Problems
- Lakeland Library Cooperative (12/20/2006)[6]

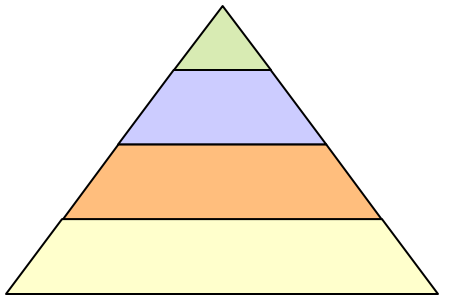
1 http://www.wfaa.com/sharedcontent/dws/wfaa/latestnews/stories/wfaa061203_kd_gpidworries.4c17588e.html
2 http://www.colorado.edu/news/releases/2006/437.html
3 http://www.heraldsun.com/durham/4-799583.cfm
4 http://www.wcax.com/Global/story.asp?S=5790220
5 http://www.latimes.com/news/local/la-me-ucla12dec12,0,7111141.story?coll=la-home-headlines
6 http://www.mlive.com/news/muchronicle/index.ssf?/base/news-10/1166631362312200.xml&coll=8

Recommendations

- Approach this Like Any Other Business Process
- Allocate funds to Security in the SDLC
- Get the right people involved
- Apply Mature & Transparent Management

SDLC Maturity (Foundation of Repeatability)

- Software Development Methodology
  - SEI-CMM, Maturity, Repeatability
  - Waterfall, Agile, Extreme Programming, Scrum, etc. [1]
  - 'Cowboy coding' not rigorous enough [2]
- Coding Standards or Programming Style [3]
- Source Code Control
  - CVS, ClearCase, SubVersion, etc.

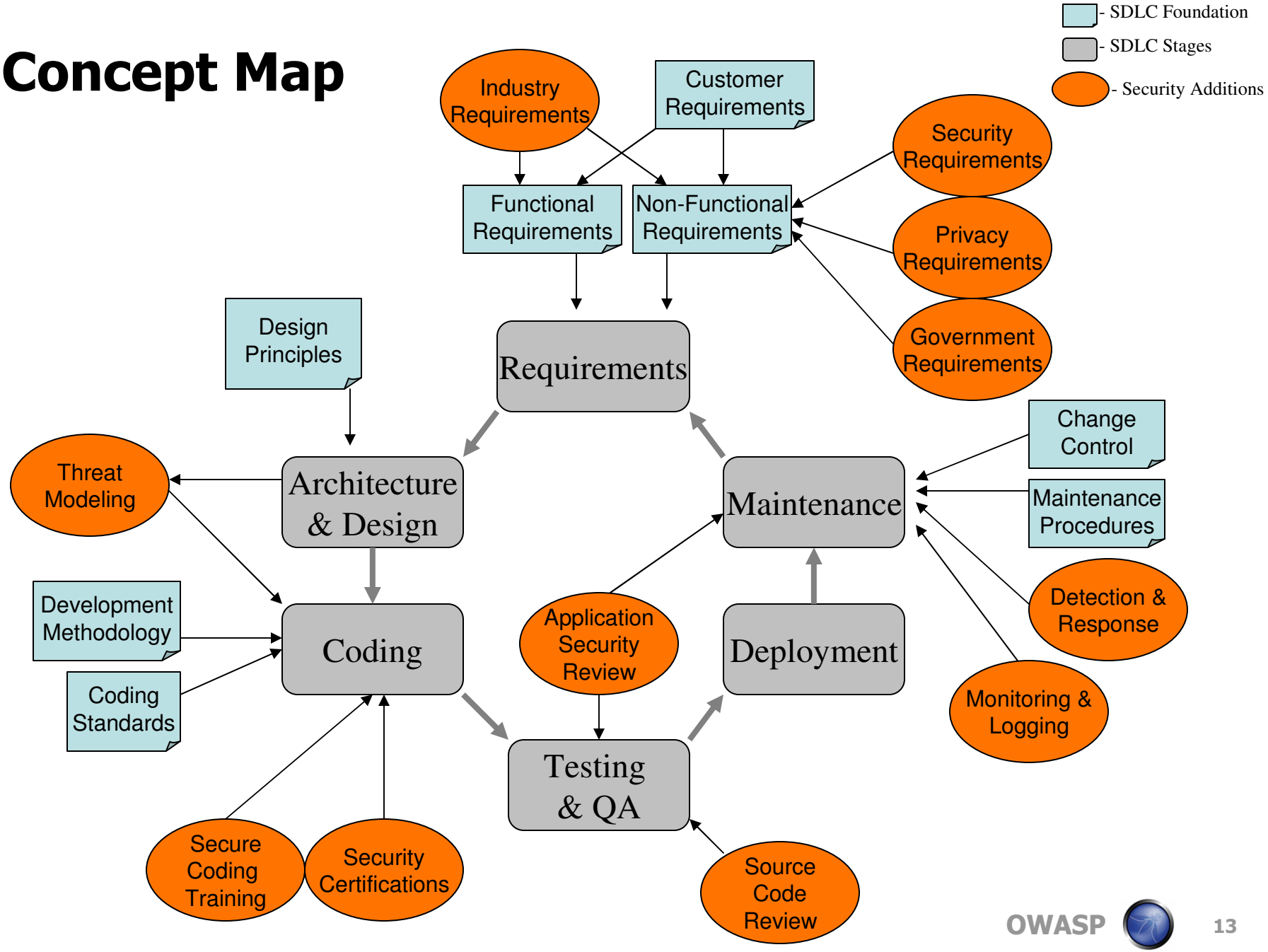


1 http://en.wikipedia.org/wiki/Software_development_methodology
2 http://en.wikipedia.org/wiki/Cowboy_coding
3 http://en.wikipedia.org/wiki/Coding_standards

Software Development Lifecycle

1. Requirements
2. Architecture & Design
3. Coding
4. Testing & Deployment
5. Maintenance

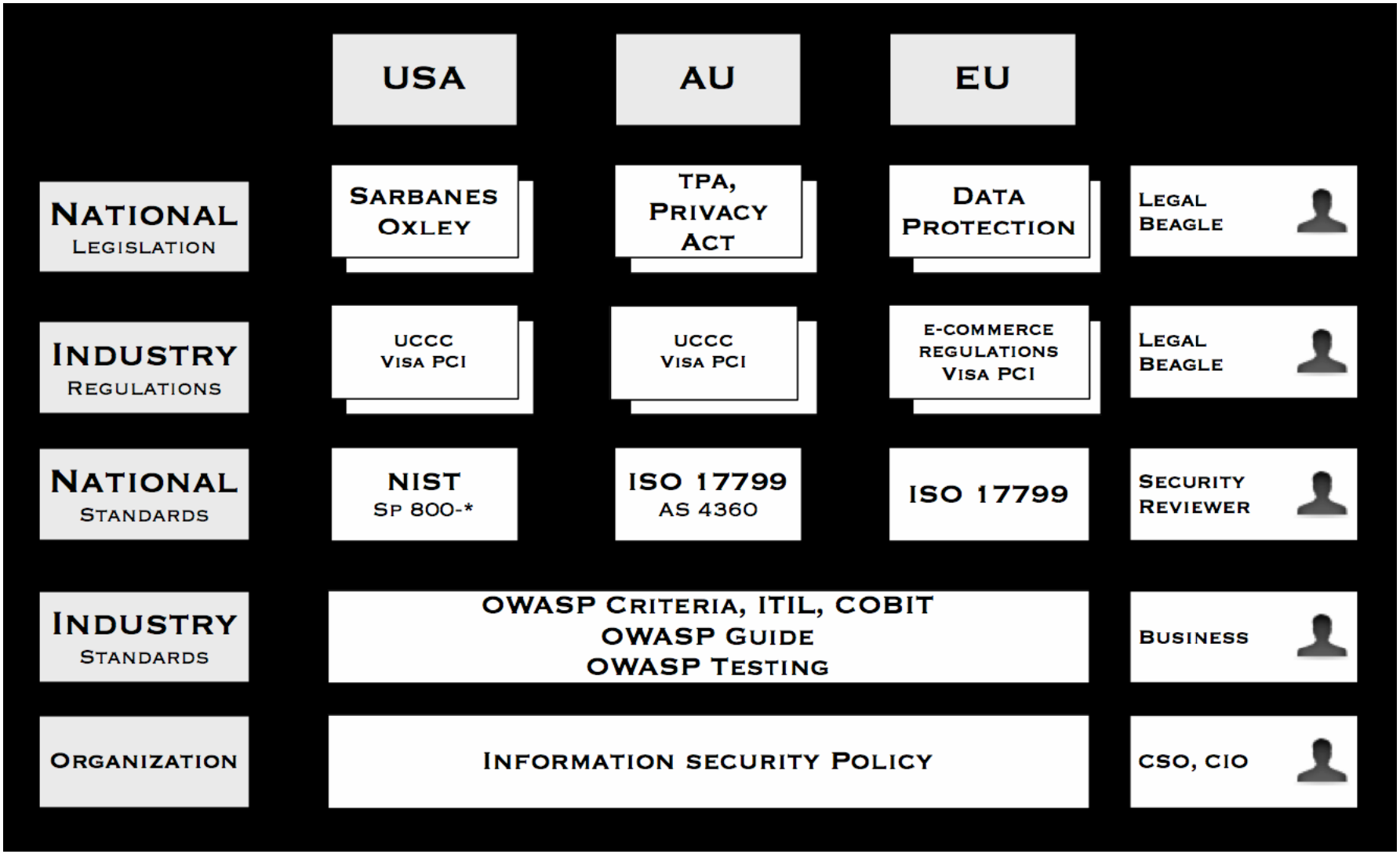
Concept Map


- SDLC Foundation
- SDLC Stages
- Security Additions
Industry Requirements
Customer Requirements
Security Requirements
Functional Requirements
Non-Functional Requirements
Privacy Requirements
Government Requirements
Design Principles
Requirements
Change Control
Threat Modeling
Architecture & Design
Maintenance
Maintenance Procedures
Development Methodology
Application Security Review
Detection & Response
Coding
Deployment
Coding Standards
Monitoring & Logging
Testing & QA
Secure Coding Training
Security Certifications
Source Code Review

Requirements

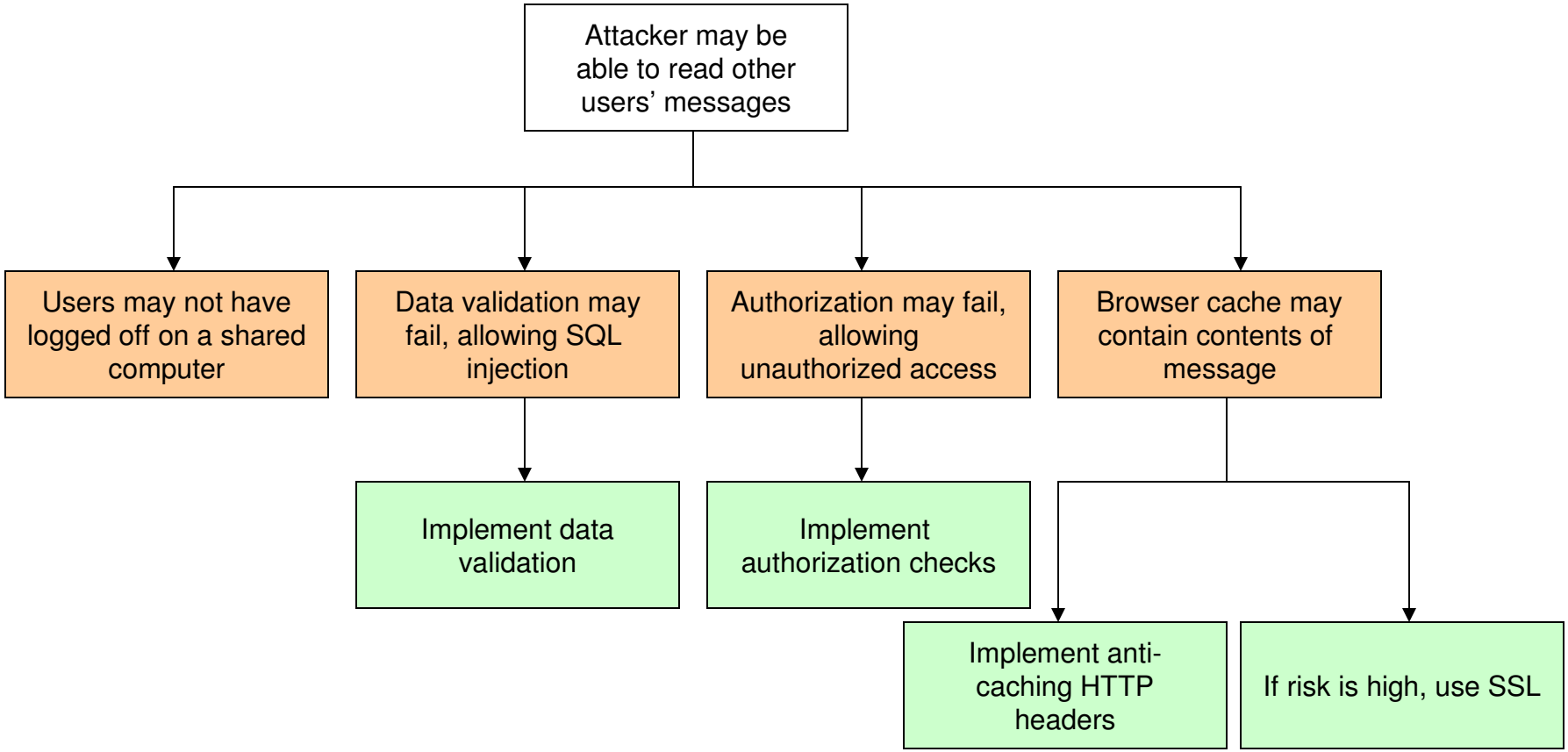
- Solicit business requirements for security
- Ask customers what they expect from security
  - Requirements should include security expectation
- Identify Security Objectives
  - What level of risk is the organization willing to absorb?
  - Service Level Agreement (SLAs)
  - Privacy & Data Protection
  - Compliance

Policy Frameworks



Source: http://www.owasp.org/index.php/Category:OWASP_Guide_Project

Threat Modeling



Source: http://www.owasp.org/index.php/Category:OWASP_Guide_Project

Key Points to Threat Modeling

- Ask the Right Question
- Threats Categories
  - Accidental discovery
  - Automated Malware
  - Curious Attacker
  - Script Kiddies
  - Motivated Attacker
  - Organized Crime
- Participation
  - User participation
  - Developer participation

Threat Modeling Methodologies

- Microsoft's Threat Modeling
  - STRIDE – threat taxonomy
  - DREAD – rating risks
- Trike
- AS / NZS 4360:2004 Risk Management
- CVSS
- OCTAVE

Source: http://www.owasp.org/index.php/Category:OWASP_Guide_Project

Secure Coding

- Secure Code Training
- Source Code Review
- Security Certification for Developers

Secure Coding

1. Authentication
2. Authorization
3. Session Management
4. Data Validation
5. Interpreter Injection
6. Canonicalization, locale, and Unicode
7. Error Handling, Auditing, and Logging
8. File System
9. Buffer Overflows
10. Administrative Interfaces
11. Cryptography

Source: http://www.owasp.org/index.php/Category:OWASP_Guide_Project

Application Security Reviews

- Planning
- Reconnaissance
- Infrastructure
- Input validation
- Denial of Service (DoS)
- Authentication & Authorization
- Information Disclosure
- Code Review
- Reporting

Source: OWASP 'Application Security Reviews' Presentation by David Byrne

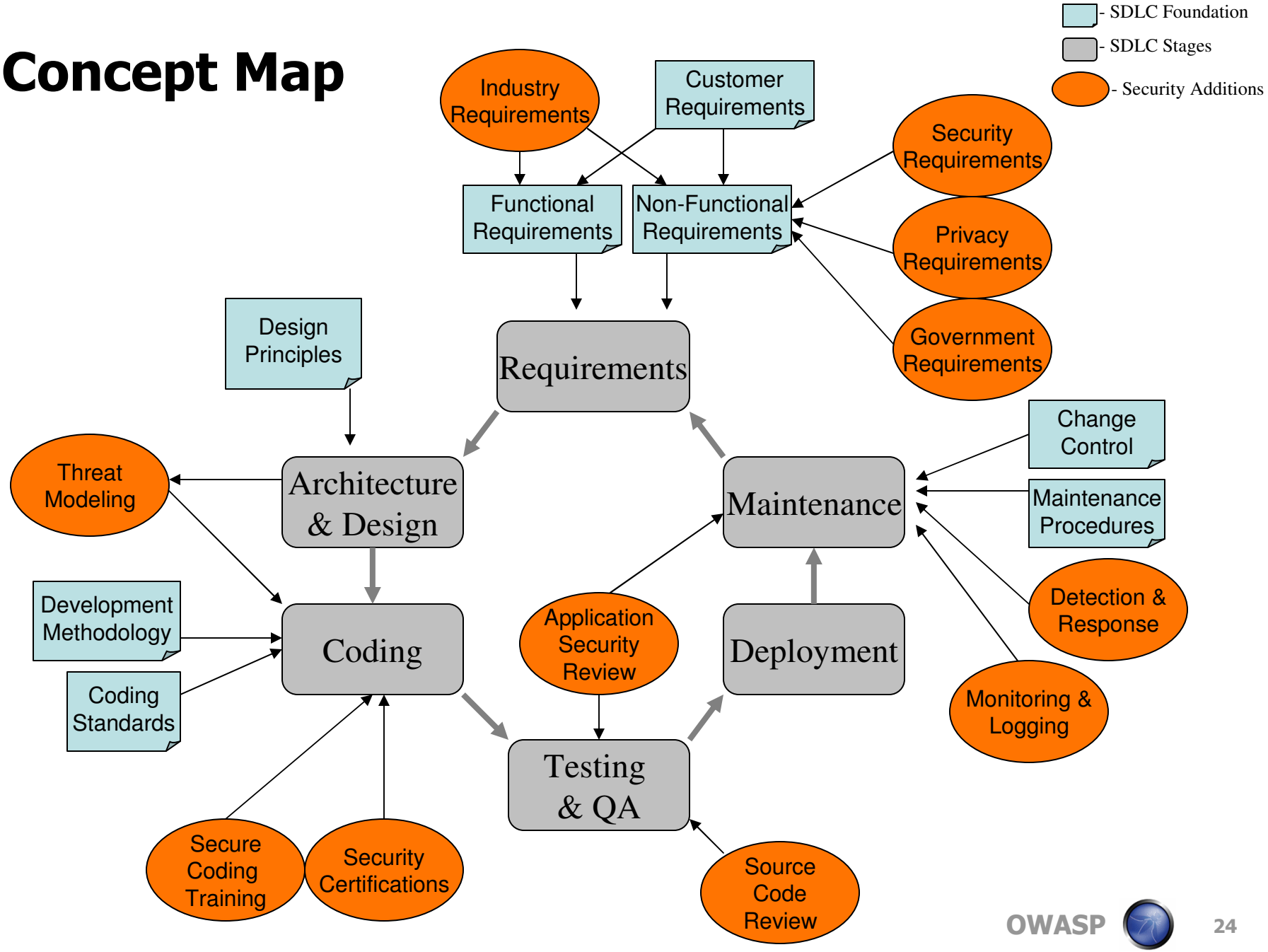
Secure Maintenance of Web Application

- Failure Analysis – How often does what fail?
- Change Control
- Monitoring & Logging
- Detection & Response

Achieving Long Term Success

- Demonstrating Leadership and Commitment is the best way to create last change in the SDLC
- *Long-term success is not technological it is cultural*
- "Good to Great" by Jim Collins
  - Good is the Enemy of Great
  - Level 5 Leadership
  - First Who…Then What
  - Confront the Brutal Facts
  - Hedgehog Concept – 'Keep it Simple'
  - Culture of Discipline

Concept Map


- SDLC Foundation
- SDLC Stages
- Security Additions
Industry Requirements
Customer Requirements
Security Requirements
Functional Requirements
Non-Functional Requirements
Privacy Requirements
Government Requirements
Design Principles
Requirements
Change Control
Threat Modeling
Architecture & Design
Maintenance
Maintenance Procedures
Development Methodology
Application Security Review
Detection & Response
Coding
Deployment
Coding Standards
Monitoring & Logging
Testing & QA
Secure Coding Training
Security Certifications
Source Code Review

Building Security into the Software Development Life Cycle

Michael Walter, CISSP
Security Consultant
waltermi@gmail.com

OWASP

January 17, 2007

The OWASP Foundation

http://www.owasp.org