FAA Secure Application Development and SDLC

NIST FCSM

April 12, 2011

Terry Fletcher

FAA, AIS-300

Federal Aviation Administration

Background

- **Definitions**
  - Application Security:
    - Application security encompasses measures taken throughout the application's life-cycle to prevent exceptions in the security policy of an application or the underlying system (vulnerabilities) through flaws in the design, development, deployment, upgrade, or maintenance of the application. -Wikipedia
  - SDLC:
    - The **Systems Development Life Cycle (SDLC)**, is the process of creating or altering systems, and the models and methodologies that people use to develop these systems. The concept generally refers to computer or information systems. Systems Development Life Cycle (SDLC) is a logical process used by a systems analyst to develop an information system, including requirements, validation, training, and user (stakeholder) ownership. . -Wikipedia

March 30, 2011 GCN Article

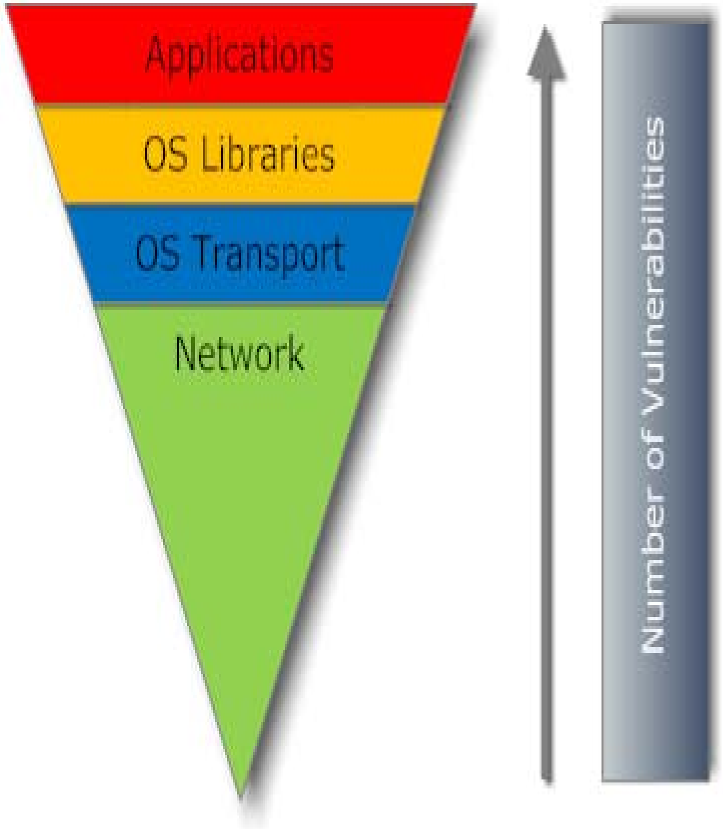
Information tops infrastructure as main target of cyber outlaws

"….retired Rear Adm. Betsy Hight, now vice president of Hewlett-Packard's cybersecurity practice. Hight said the network no longer is the primary target for attacks. ***Seventy percent of attacks now target applications*****,** and that is a shift in the threat landscape that has not been adequately dealt with **….."**

Vulnerabilities Exploitation Trends

Application Vulnerabilities Exceed OS Vulnerabilities

- ***During the last few years, the number of vulnerabilities being discovered in applications is far greater than the number of vulnerabilities discovered in operating systems. As a result, more exploitation attempts are recorded on application programs.*** *The most "popular" applications for exploitation tend to change over time since the rationale for targeting a particular application often depends on factors like prevalence or the inability to effectively patch. Due to the current trend of converting trusted web sites into malicious servers, browsers and client-side applications that can be invoked by browsers seem to be consistently targeted.*



Application Vulnerabilities Inevitable

- **Bugs are guaranteed in software and a certain portion *will be security related***
- **Threats to web applications are global**
  - Web applications constitute more than 60% of total internet attack attempts
    - SQL injection and Cross site scripting flaws account for >80% of vulnerabilities
- **2010 CWE/SANS Top 25 Most Dangerous Software Errors**
  - http://cwe.mitre.org/top25/

What Can We Do

- **Security must be considered from beginning of SDLC**
  - It is iterative process
- **Involve all stakeholders in business and security requirements definition**
  - Proper level of security assurance requires trade-offs between business needs and security
  - Security/Development teams meet regularly
- **Security review during architecture through development**
  - NSF study estimated that uncovering and correcting severe software problems during requirements and design is 100x less expensive than finding it in production
- **Final security audit before launch**
- **Commitment to ongoing security review**

How Do We Do This

- **Develop an Application Security Program**
  - – Select a framework
  - – Integrate into the SDLC
  - – Develop repeatable processes

Security Framework

- **Select a framework**
  - Consider your organizations needs
  - Consider regulatory requirements
  - Consider existing best practices
  - Consider your geographic region

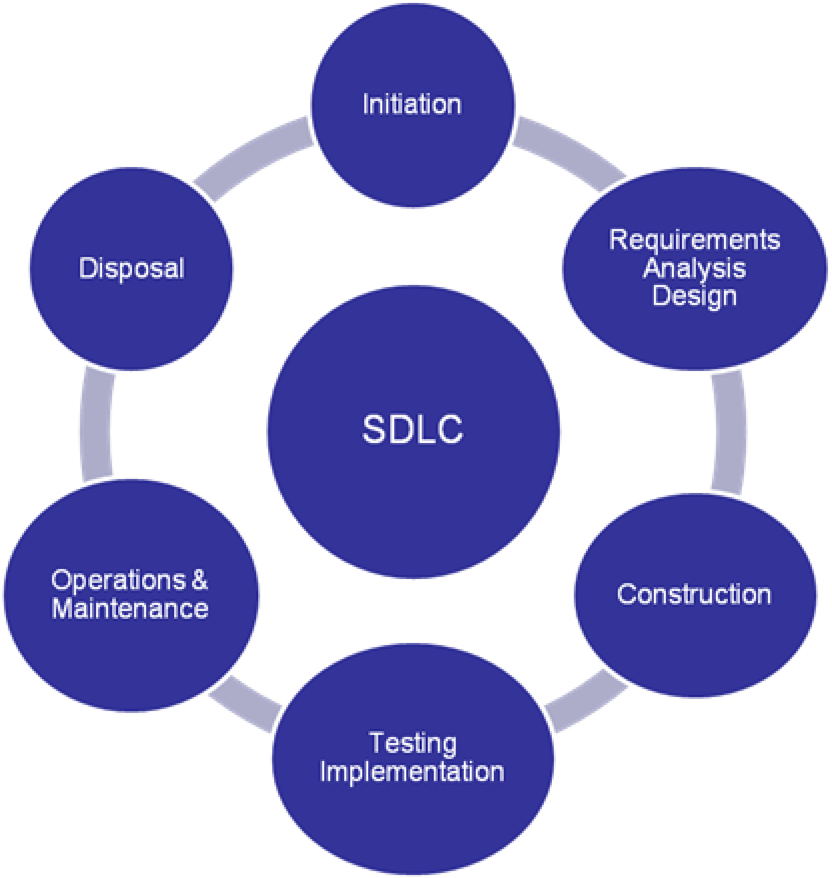
FISMA

PCI

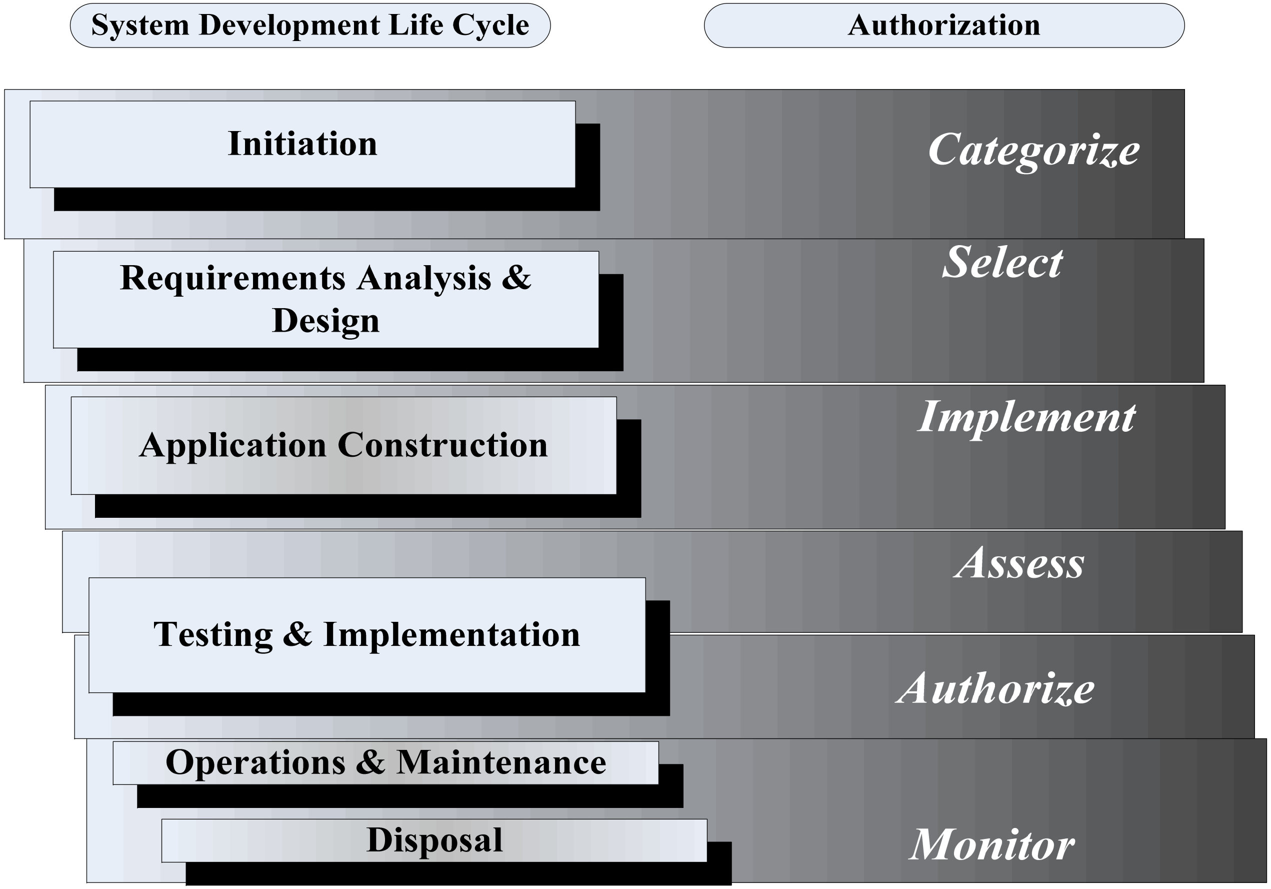
HIPAA

Integrate Into SDLC

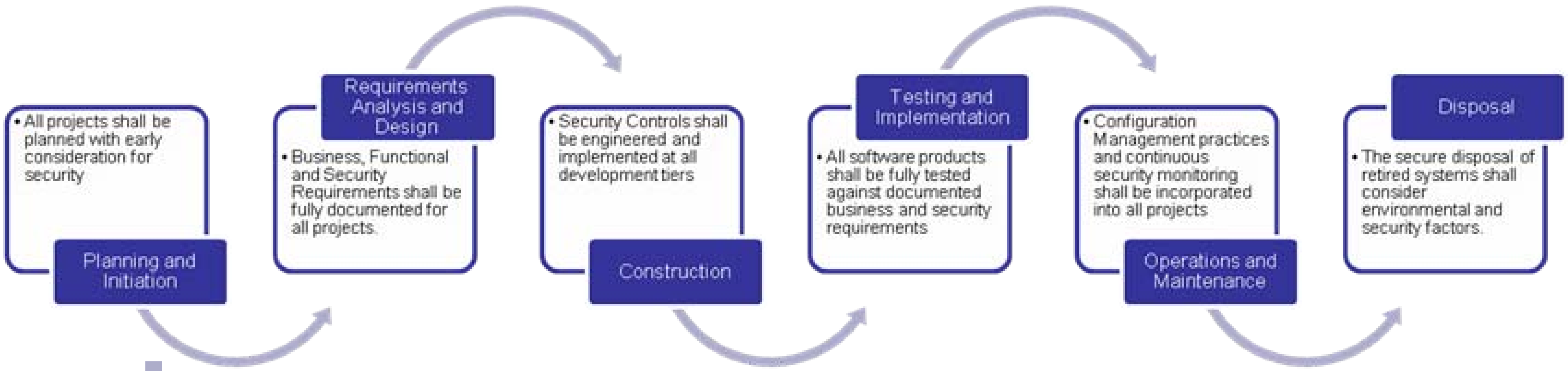
Minimum System Development Life Cycle (SDLC) Requirements for Building Secure Applications Version 1.0: Intent to ensure that FAA software development efforts adequately plan for security; properly identify, assess and mitigate risks; include applicable security controls in the software design; adhere to agency and federal IT policies and regulations, and continually monitor and assess security through system retirement.



FAA SDLC & RMF


System Development Life Cycle
Authorization
Initiation
Categorize
Requirements Analysis & Design
Select
Application Construction
Implement
Assess
Testing & Implementation
Authorize
Operations & Maintenance
Disposal
Monitor

Minimum SDLC Requirements



Planning and Initiation

- All projects shall be planned with early consideration for security

Requirements Analysis and Design

- Business, Functional and Security Requirements shall be fully documented for all projects.

Construction

- Security Controls shall be engineered and implemented at all development tiers

Testing and Implementation

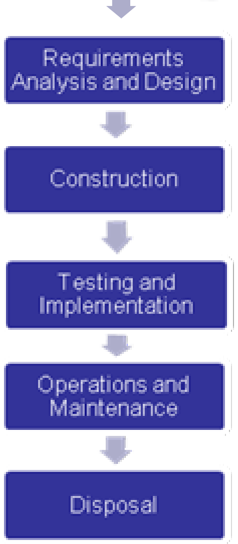
- All software products shall be fully tested against documented business and security requirements

Operations and Maintenance

- Configuration Management practices and continuous security monitoring shall be incorporated into all projects

Disposal

- The secure disposal of retired systems shall consider environmental and security factors.



Requirements Analysis and Design

Construction

Testing and Implementation

Operations and Maintenance

Disposal

These minimum SDLC requirements must be included in all FAA software system development efforts, regardless of the platform or the LOB/SO’s preferred life cycle methodology (waterfall, agile, iterative, etc.)

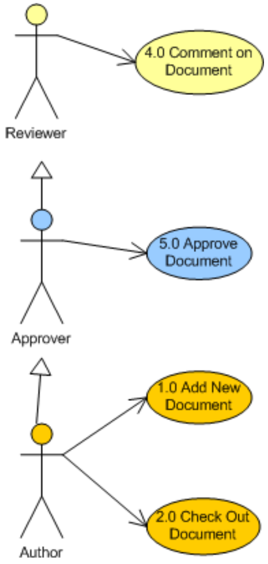
Initiation & Planning

- **Project Sponsorship -The Project Sponsor acquires the appropriate management support to secure funding and sponsorship for the project prior to initiation**
- **Project Planning - All software development projects planned with early consideration for security**
- **Risk Management – Risk management planned to identify and mitigate risks continuously throughout the life cycle**

Requirements Analysis & Design

- **Requirements Specification – Develop and document business, security, privacy, and other regulatory requirements**
- **Detail Requirements – Develop testable requirements to ensure the construction of correct solutions**
  - Map security requirements to use cases
- **Requirements Analysis – Requirements analyzed and security assessed as part of application testing**

SDLC Example - Requirements Management



Requirements Traceability Matrix: Maps the high-level business requirements to system use cases

- Each high-level requirement shall correspond to a single, sequentially numbered and uniquely named use case.
- Each low-level requirement shall correspond to a single, sequentially numbered and uniquely named test case.
- Each use case shall contain one or more functional test cases.

Release	Feature	Business Requirement	Functional Role	Use Case No.	Use Case Name	Depends On	Implements
1.0	**Documents**	**Users must be able to add new documents to the FTS.**	**Author**	**1.0**	**Add New Document**		**1.1**
1.0	**Documents**	**Users must be able to select organizations that can view the document.**	**Author**	**1.1**	**Specify Document Access**	**1.0**	
1.0	**Documents**	**Users must be able to check out a document for editing**	**Author**	**2.0**	**Check Out Document**	**1.0**	
1.0	**Reports**	**Users must be provided a complete history on who has accessed a document.**	**Author, Reviewer, Approver**	**3.0**	**Document History Report**	**1.0**	

Test Case No.	Test Case Name	Functional Requirement	Functional Role	Depends On	Implements
1.0.A	**Designate Document Authors**	**The user entering the document shall be listed as an anuthor of the document and he/she can designate additonal authors who will have edit rights to the document.**	**Author**		
1.0.B	**Upload Document**	**When adding the document the system shall enable the user to upload the document directly from his/her personal computer**	**Author**		
2.0.A	**Lock Document**	**When a user has checked out a document for editing the system shall prevent any additional authors from checking out the document.**	**Author**		

Application Construction

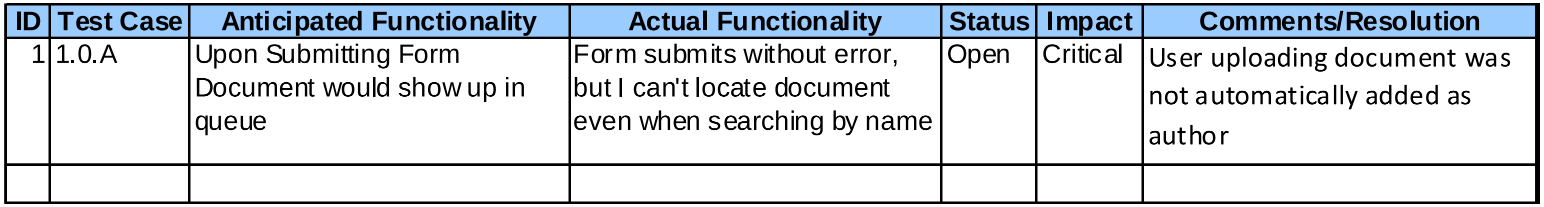
- **Database, Program and User Interface Development - The development of database and software components incorporate security controls**

- **Unit Testing - Software developers test their specific components to ensure compliance with functional and security requirements.**
  - Scan results should be available to developer to address in system test

- **Newly identified risks, including security related risks, must be communicated and mitigated in a timely manner**

Testing and Implementation

- **Systems/ Regression Testing – Test new and existing software functionality to identify software defects and vulnerabilities**
  - Vulnerability and integrity scanning
  - Complete test report includes all relevant test cases covered in scenarios
    - Only failed test cases should have issues recorded



ID	Test Case	Anticipated Functionality	Actual Functionality	Status	Impact	Comments/Resolution
1	1.0.A	Upon Submitting Form Document would show up in queue	Form submits without error, but I can't locate document even when searching by name	Open	Critical	User uploading document was not automatically added as author

- **Operational Readiness/Security Review- The information system is fully tested and accepted by the system owner and undergoes all required operational and security assessments, includes vulnerability scanning and mitigation prior to being authorized for production deployment and implementation**

Operations & Maintenance

- **Perform Configuration Management and Control - The "System" is under and follows configuration management and change control processes, evaluating security risks for requested change**

- **Conduct Continuous Monitoring - The "System" follows LOB-defined continuous monitoring strategy to maintain reliability, data integrity, and appropriate level of security**

Disposal

- **Ensure Information Preservation- Data contained within the information system that is subject to FOIA, and other records retention guidelines are properly archived to ensure information preservation.**
- **Sanitize Media - Sensitive, confidential and personally identifiable information must be completely cleaned from all storage devices and other media.**
- **Dispose of Hardware and Software- Excess hardware and software is properly surplused or disposed of in compliance with environmental and property management regulations**
- **Closure of System - Retired systems updated accordingly in the FAA Application/System Inventory and properly shut down to ensure proper access restrictions.**

Develop Repeatable Processes

- **Without Process we have no idea how to get there**
  - Clearly define the processes
    - *Minimum System Development Life Cycle (SDLC) Requirements for Building Secure Applications Version 1.0*
  - Clearly document the procedures
  - Require every application system to go through the program
  - Educate, educate, educate
    - FAA ISS Security Conference

Software Assurance Policy

- **FAA Order 1370.109 *Software Assurance Policy***
  - Defines software assurance assessment activities that must be conducted as part of software development lifecycle
  - Addresses assurance requirements for each stage of SDLC
  - Identifies roles and responsibilities within FAA for software assurance

Summary Examples

- **Tactical Voice Switch – SPAWAR security agent**
  - Security involved with requirements definition, review of design documents, testing
- **2 COCO’s with PII – FAA**
  - Discovered Authentication issue as part of review of “C&A” submittal for System 1
    - Resolution delayed operations by 7 months
  - System 1 owner brought authentication requirement to attention of development team for System 2
    - Addressed as part of development and no delays
- **GPS Upgrade – Joint Service**
  - Integrated but separate

Resources

- http://www.sans.org/top25-software-errors/
  - **What Errors Are Included in the Top 25 Software Errors?**
    - Version 2.0 Updated February 16, 2010
- http://www.mitre.org/news/digest/defense_intelligence/02_09/errors.html
  - Drawn from more than 700 errors listed in the Common Weakness Enumeration (CWE) based on frequency & severity
  - see http://cwe.mitre.org/top25/)
    - Updated March 29, 2010
- **http://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project**
  - The Open Web Application Security Project (OWASP) Top 10 Web Application Security Risks for 2010

Resource: Contract Reference

http://www.sans.org/appseccontract/

- **Application Security Procurement Language**

This document helps enable buyers of custom software to make code writers responsible for checking the code and for fixing security flaws before software is delivered.

Proactive Software Assurance: http://www.sans.org/whatworks/

- Single most effective step in thwarting attacks is to design applications and develop code with fewer security flaws and stronger security features. Security tools that can find vulnerabilities and reduce the time spent mitigating those weaknesses.
  - 1.1 Source Code and Binary Code Testing Tools and Services
    - These tools search through code with the goal of finding potential vulnerabilities and other security weaknesses. Since they don't require a complete software system, these tools can be used to test code during development or integration.
  - 1.2 Application Security Scanners (Black Box Scanners)
    - These tools detect common programming errors in Web-based applications. While tools should be part of the solution, skilled humans are the key to finding lower level vulnerabilities that more targeted attacks will exploit.
  - 1.3 Application Security Skills Assessment & Certification
    - Application security managers can ensure that programmers are able to identify and eliminate common security flaws from code by using assessment tools and having outsourced programmers prove their knowledge through certification.