

Building Security Into The Software Life Cycle

A Business Case



Marco M. Morana
Senior Consultant
Foundstone Professional Services
a Division of McAfee
Email: marco.morana@foundstone.com

Outline

- » Glossary
- » Application Security Risks
- » Software Security and Application Security
- » Costs and Return Of Security Investment (ROSI)
- » Software Security Development Life Cycle (S-SDLC)
- » Process Models and Frameworks
- » Business Risks, Technical Risks and Strategies
- » Summary
- » Resources

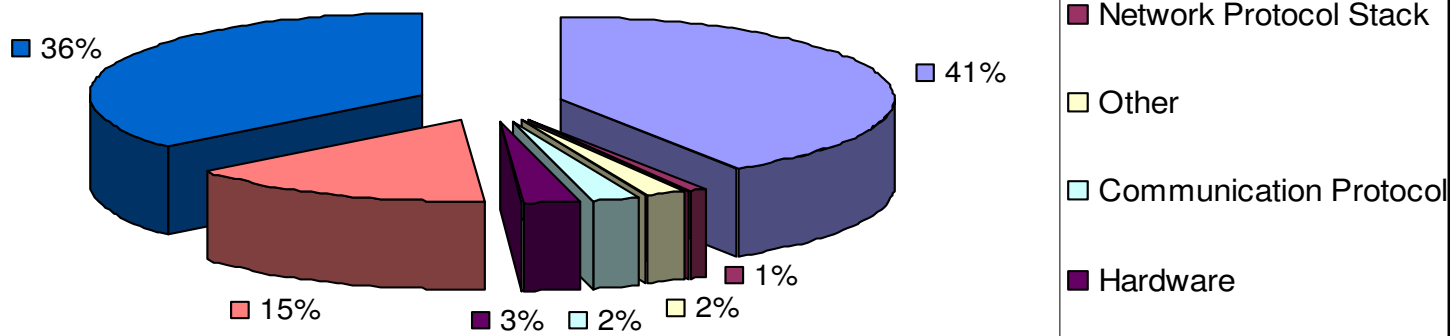
Glossary

- » **Information Security Risks:** *the probability that a particular threat-source will exercise a particular information system vulnerability and the resulting impact if this should occur (NIST publication 800-27)*
- » **Software Security:** *a way to defend against software exploits by building software to be secure (McGraw Exploiting Software)*
- » **Application Security:** *a way to defend against software exploits in a post-facto way after deployment is complete (McGraw Exploiting Software)*
- » **Return Of Security Investment in Security (ROSI):** *The total amount of money that an organization is expected to save in a year by implementing a security control (Microsoft Security Risk Management Guide)*

What is at risk?

Target Applications At Risk

92% of reported vulnerabilities are in applications not in networks



Source: NIST

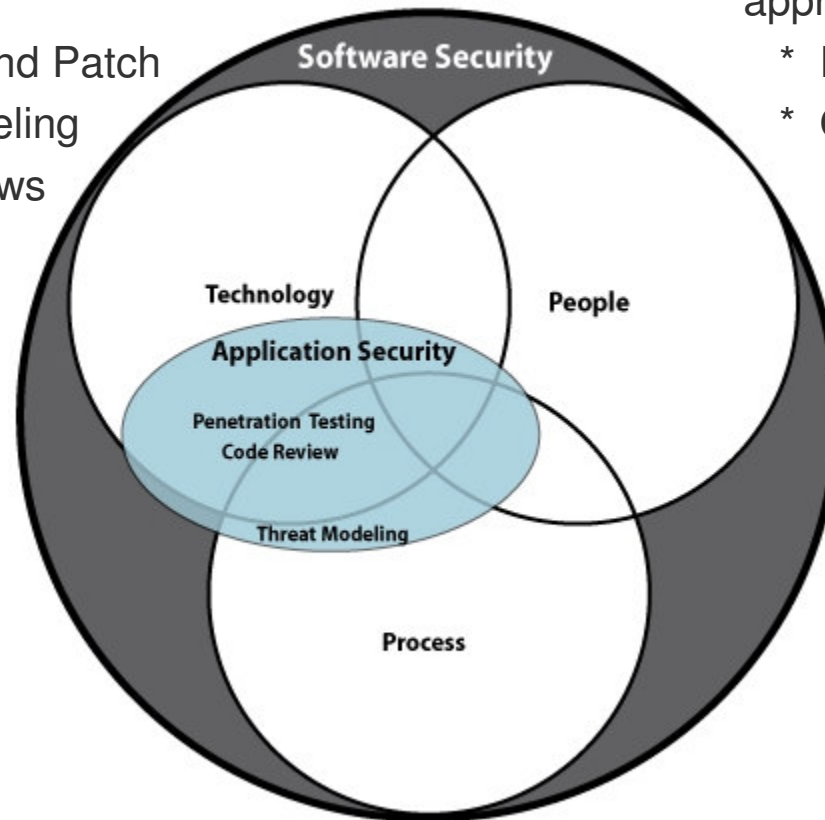
How we approach risk?

Application Security

- Issue-based, short-term approach
 - * Penetrate and Patch
 - * Threat Modeling
 - * Code Reviews

Software Security

- Holistic, long-term approach
 - * Root Cause Analysis
 - * Organizational Change



What are the costs?

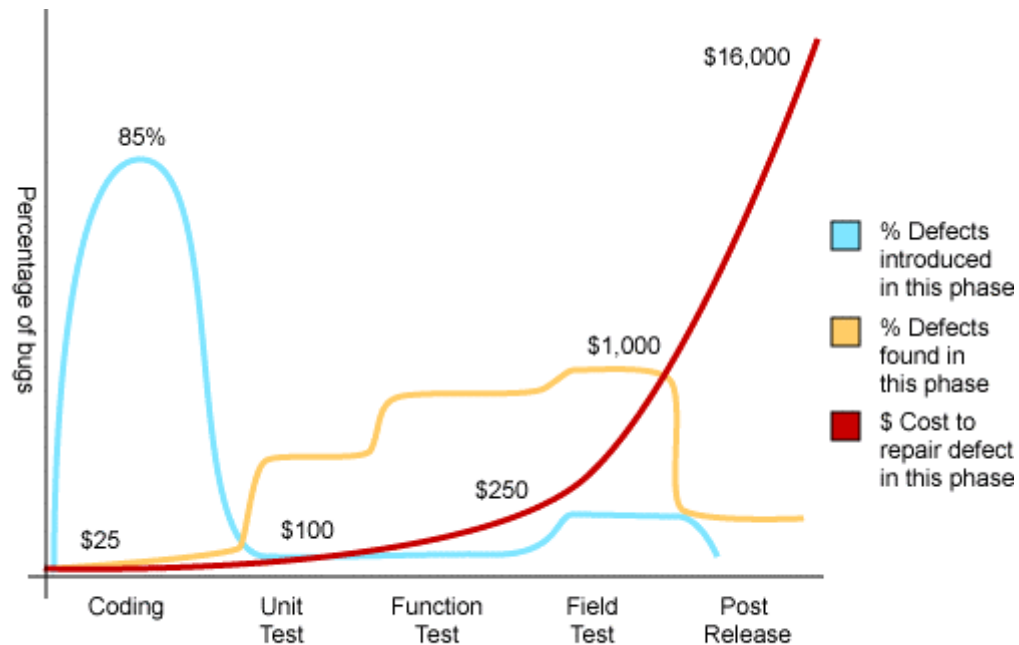
Application Security Costs:

- » Defect Management: 5 defects/KLOC, \$ 30,000/KLOC (Business week)
- » Patch Management: 1000 servers, \$ 300,000 to test and deploy a patch (Gartner)
- » Loss of productivity due of loss of service: \$ 500 ML lost from DoS attack (Microsoft)

Software Security Costs:

- » Unbudgeted time to fix security problems: 1000 man-hours (Microsoft)
- » Cost of training software developers in security: \$100 Million (Microsoft)
- » Inadequate software testing costs: \$3.3 billion (NIST)

When we do address the problem?



Source: Applied Software Measurement, Capers Jones, 1996

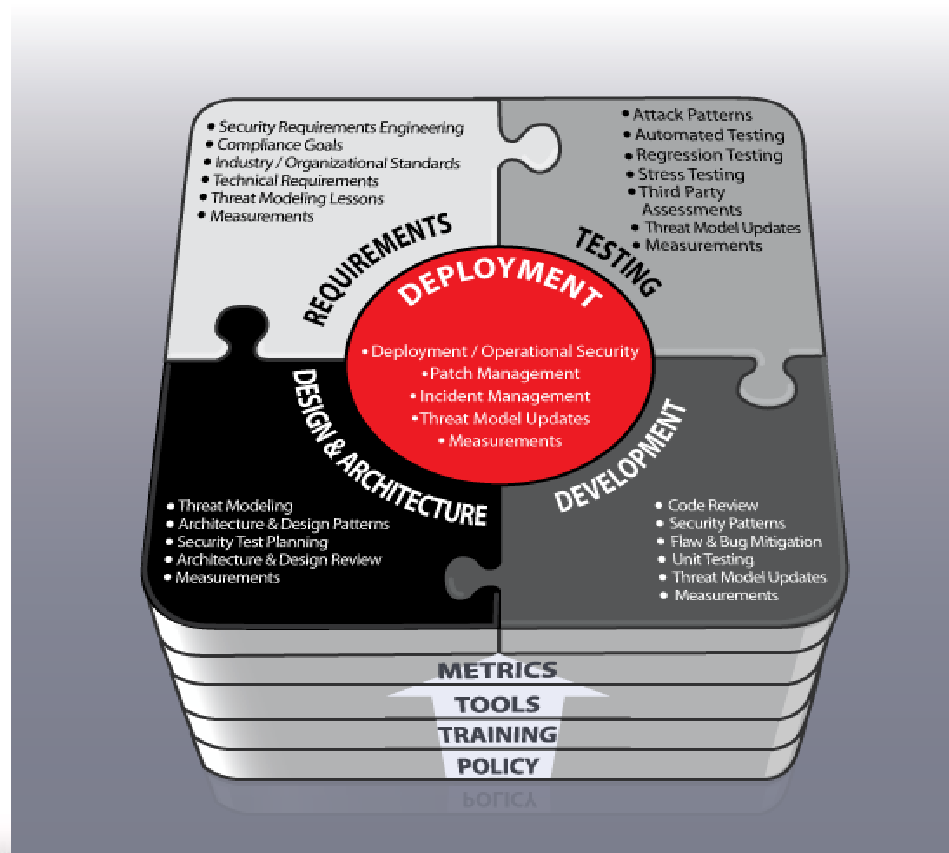
» Today most people test after software is built!

When is more cost effective to build security in?

- » **Assume the following data from a study (IBM):**
 - Secure Software Engineering Expense Per Phase
 - Number of Security Defects found Per Phase
 - Percentage of Vulnerabilities Fixed

- » **The Return Of Security Investment (ROSI) in dollar savings for every \$ 100,000 spent is:**
 - \$ 21,000 when defects are fixed and identified during design
 - \$ 15,000 when defects are fixed during implementation
 - \$ 12,000 when defects are fixed during tests

Software Risk Management and Secure Software Development Life Cycles (S-SDLC)



How do we get there?

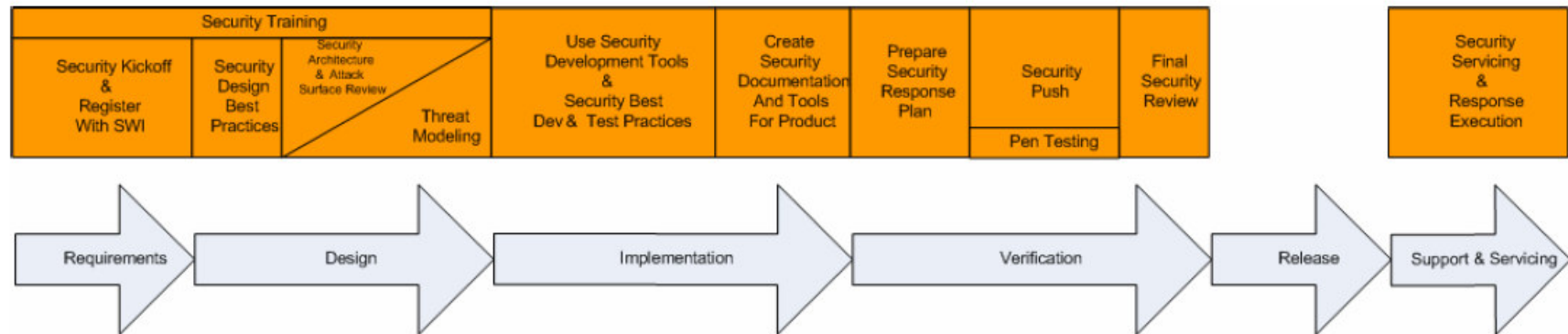
1. Adopt an activity driven approach
2. Document security activities derived by best practices
3. Define dependencies and prerequisites
4. Define entry scenarios for the activities
5. Define strategic and tactical tracks for the activities
6. Define the roadmaps for software security
7. Position the activities with respect to different SDLC methodologies

Security-Enhancing Lifecycle Process Models

1. Enhance security through a repeatable and measurable process
2. Provide guidance on secure software activities
3. Provide secure software development reviews
4. Include tactical resources
5. Provision the use of automation tools
6. Suggest roles for conducting the activities
7. Integrate with foundational software development activities

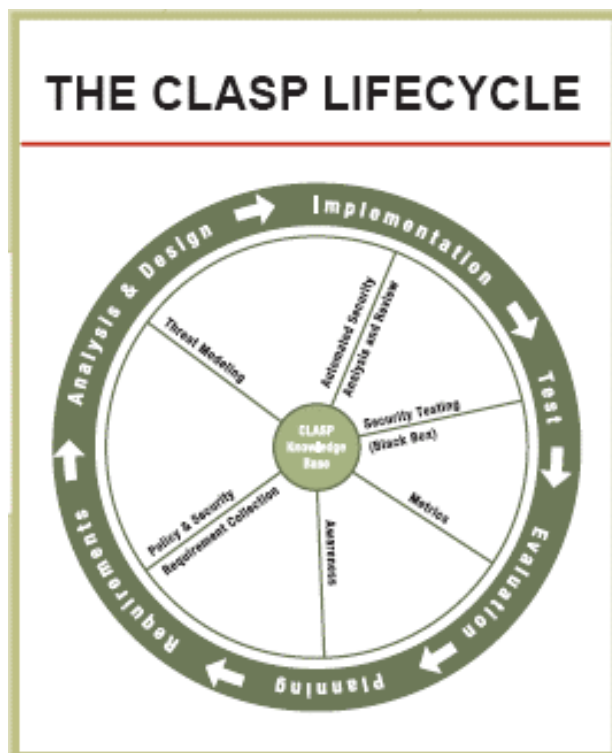
Security Enhancing Process Models

- » Microsoft's Trustworthy Computing Security Development Lifecycle



Security Enhancing Process Models

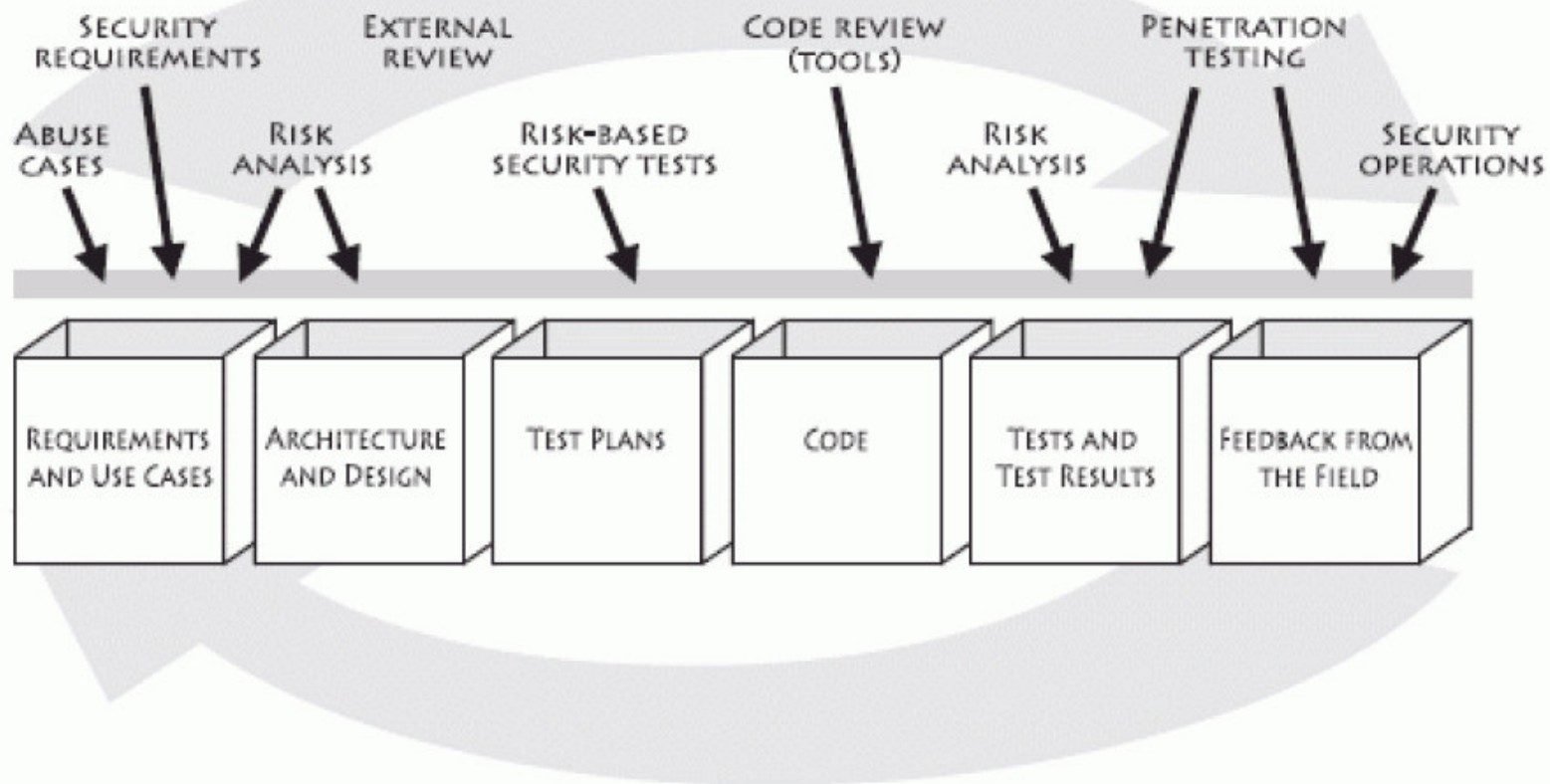
- » Comprehensive Lightweight Application Security Process (CLASP)



- CLASP
BEST PRACTICES**
- 1) Institute awareness programs
 - 2) Perform application assessments
 - 3) Capture security requirements
 - 4) Implement secure development practices
 - 5) Build vulnerability remediation procedures
 - 6) Define & monitor metrics
 - 7) Publish operational security guidelines

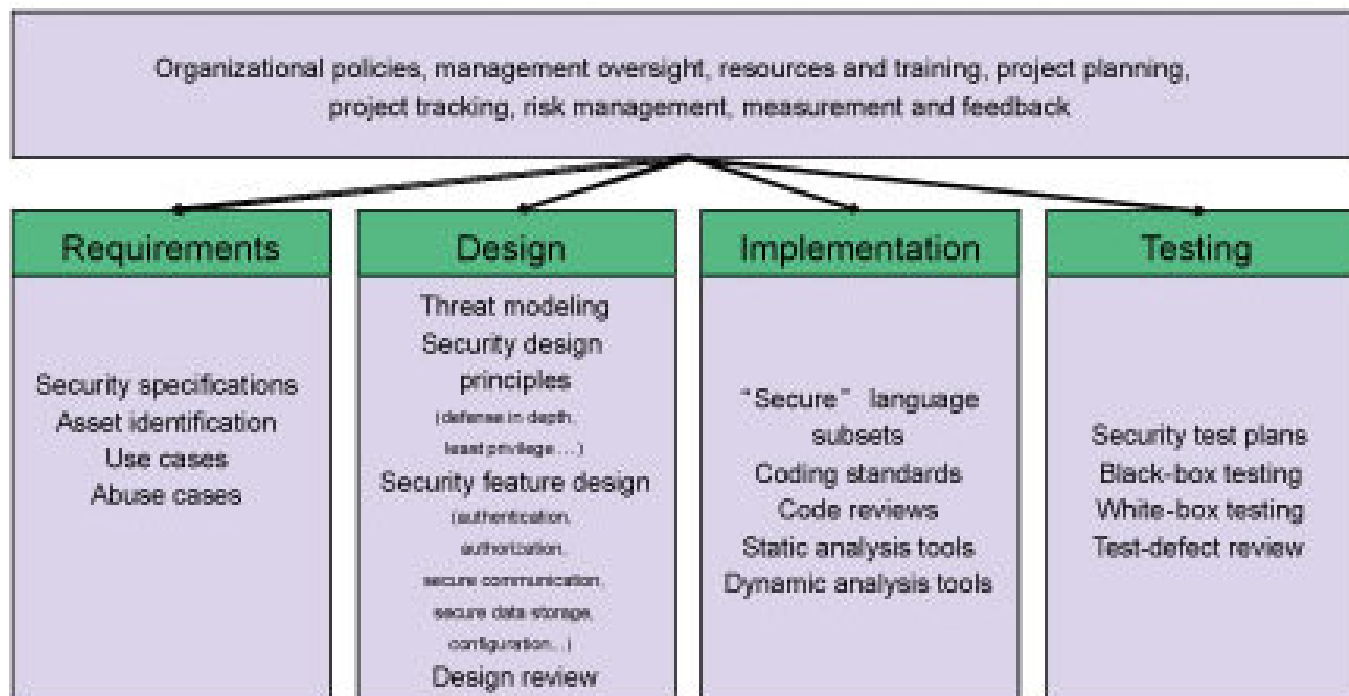
Security Enhancing Process Models

» Gary McGraw Touch-Point Model



Security Enhancing Process Models

- » SEI Team Software Process for Secure Software Development



Software Security Frameworks

SDLC Phases	Requirements		Design	Development	Testing	Deployment and Operations	
Secure Software Best Practices	Preliminary Software Risk Analysis	Security Requirements Engineering	Security Risk-Driven Design	Secure Code Implementation	Security Tests	Security Configuration & Deployment	Secure Operations
On Going S-SDLC Activities	Metrics and Measurements, Training and Awareness						
S-SDLC Activities	Define Use & Misuse Cases	Define Security Requirements	Secure Architecture & Design Patterns Threat Modeling Security Test Planning Security Architecture Review	Peer Code Review Automated Static and Dynamic Code Review Security Unit Tests	Functional Test Risk Driven Tests System Tests White Box Testing Black Box Testing	Secure Configuration Secure Deployment	
Other Disciplines	High Level Risk Assessments		Technical Risk Assessment				Incident Management Patch Management
Other On Going Disciplines	Information Risk Management, Defect Management, Change Management, Vulnerability Management						

Business Risks, Technical Risks and Strategies

» Business Risk Factors:

- Business impact
- Value of the assets
- Information risk management

» Technical Risks Factors:

- Technical impacts
- Value of data
- Software risks associated to threats and vulnerabilities

» Risk Remediation Strategies:

- Cost to fix vulnerabilities vs. cost of exploitation
- Translate technical risks to business risks
- Assess, evaluate and prioritize by business impact

In Summary

- 1. Make the initial business case**
 - Costs
 - Return Of Security Investment (ROSI)
- 2. Adopt a formal process to build security into the SDLC**
 - Security Enhancing Process Models
 - Software Security Frameworks
- 3. Have a plan for the implementation**
 - Tactical and strategic plans
 - Roadmaps: short term and long term
- 4. Integrate Software Security with Information Security Risks**
 - Assess business impacts
 - Factor technical and business impacts in overall risks
- 5. Review the business case and commit to it**
 - Measure overall risk and decide strategies
 - Commit people, process and technology



Questions?

Foundstone Links

- » Foundstone Software Application Security Services (SASS)
www.foundstone.com/sass
- » Foundstone Resources
www.foundstone.com/resources.overview.htm

Thank you for listening!