Foundstone®

OWASP
Open Web Application
Security Project

Building a Software Security Program

Software Security Maturity Assessment Services

TM

Kuai Hinojosa
Global Lead Software Security Engineer

August 20, 2014

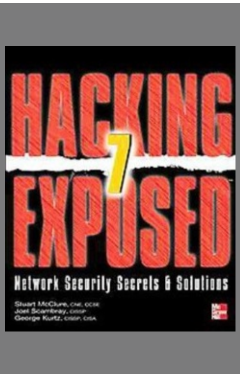
Agenda

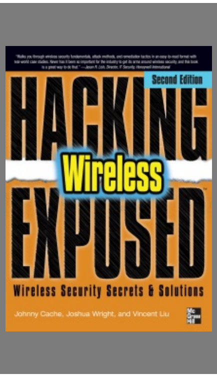
Building a Software Security Program

- Foundstone’s Software Security Maturity Assessment Services
- Case Study
- Summary
- Questions

Thought Leadership

Contributing authors to all editions of Hacking Exposed




Second Edition
HACKING
Wireless
EXPOSED
Wireless Security Secrets & Solutions
Johnny Cache, Joshua Wright, and Vincent Liu

OPEN SECURITY RESEARCH

sponsored by Foundstone

blog.opensecurityresearch.com

NYU-poly NYIT CAL POLY

POLYTECHNIC INSTITUTE OF NYU

Professors and Lecturers

CYBERPATRIOT CSAW

Competition Judges/Mentors

Foundstone

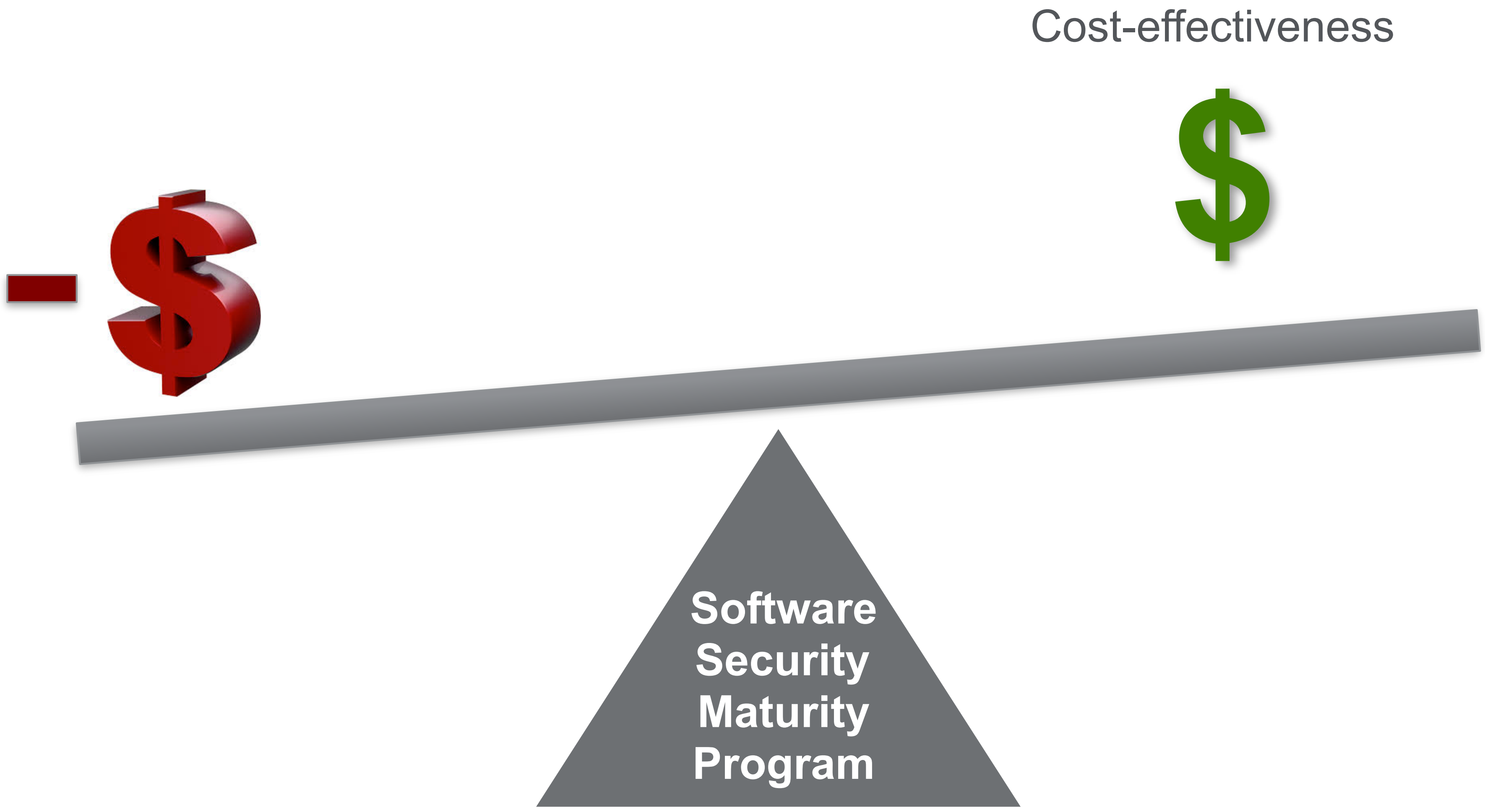
intel Security

Common Challenges

Building a Software Security Program

Common Challenges

Building a Software Security Program



Software Assurance Maturity Model

Building a Software Security Program

Governance	Architecture	Verification	Operations
Strategy & Metrics (SM)	Threat Assessment (TA)	Design Review (DR)	Data Leakage Prevention (DP)
Policy & Compliance (PC)	Data Modeling (DM)	Data Audit (DA)	Cryptography & Hardening (CH)
Education & Guidance (EG)	Security Requirements (SR)	Access Certification (AC)	Operational Security (OS)

Maturity Level Per Practice (+ = between levels)

0	Largely Absent	1	Adapt	2	Sustain	3	Master and Scale

Software Assurance Maturity Model

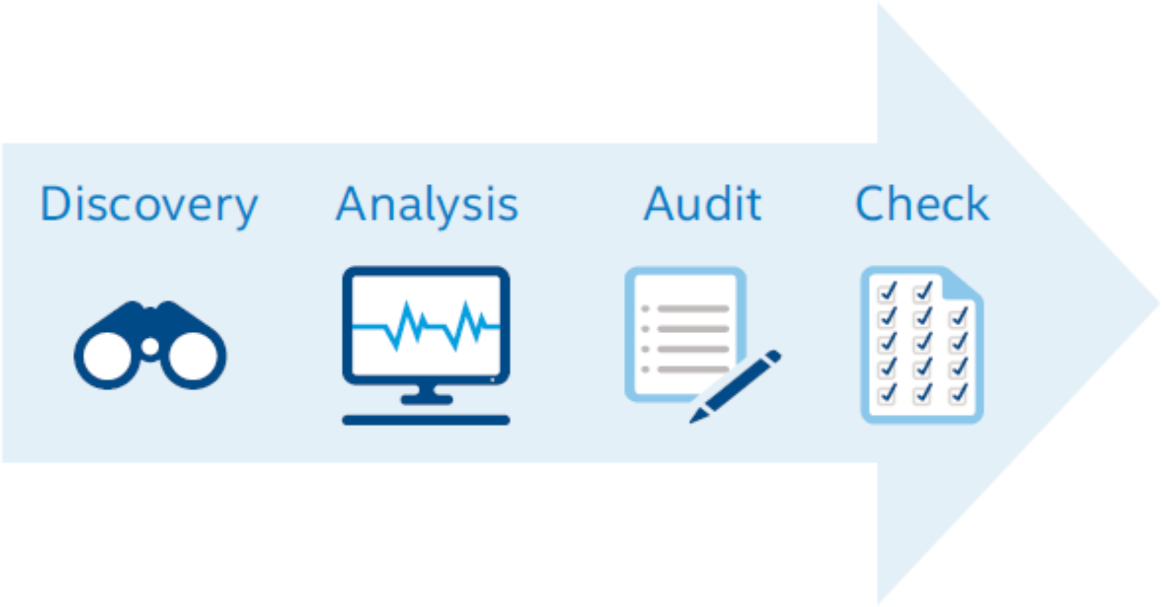
Building a Software Security Program

Strategy & Metrics

	1.	2.	3.
Objective	Establish unified strategic roadmap for software security within the organization	Measure relative value of data and software assets and choose risk tolerance	Align security expenditure with relevant business indicators and asset value
Activities	A. Estimate business risk profile derived from secure development compliance goals B. Build and maintain a PCI centric software security program roadmap	A. Classify data and software applications handling or storing credit card information based on business risk B. Establish and measure per classification security goals	A. Conduct periodic industry wide cost comparisons of compliance efforts related to secure software development B. Collect metrics for historic security spending
Results	Concrete list of the most critical business-level risks caused by software within PCI scope Tailored roadmap that addresses the security needs for your organization with minimal overhead Organization-wide understanding of how the assurance program will grow over time	Customized compliance focused assurance plan per project Organization wide understanding of security relevance of data and software applications Better informed stakeholders with respect to compliance efforts and risk acceptance	Information to make informed decisions on compliance related expenditures Estimates of past financial loss linked to security issues and compliance Per project consideration of compliance efforts and security expenses

SSMA – Phase 1 (Assessment)

Building a Software Security Program



Key Benefits

- Maps current security practices against recommendations by the maturity model
- Highlights gaps in SDLC
- Gathers supporting evidence thought risk base testing approach
- Offers a head start to improve an organization’s software security posture

SSMA – Key Findings

Building a Software Security Program

People Gaps

- Secure software development training program
- Security strategy aligned with external compliance driver

Process Gaps

- Guidance implementing a SDL such as;
    - Security Architecture Practice
    - Design Review Practice
    - Code Review Practice
    - Security Testing Practice
    - Vulnerability Management Practice
- Standardize Web server and DB server build processes
- Security & change control

Technology Gaps

- Development tools integrating with security tools
- Tools for automation of processes

SSMA – Sample Score Card & Check Point

Building a Software Security Program

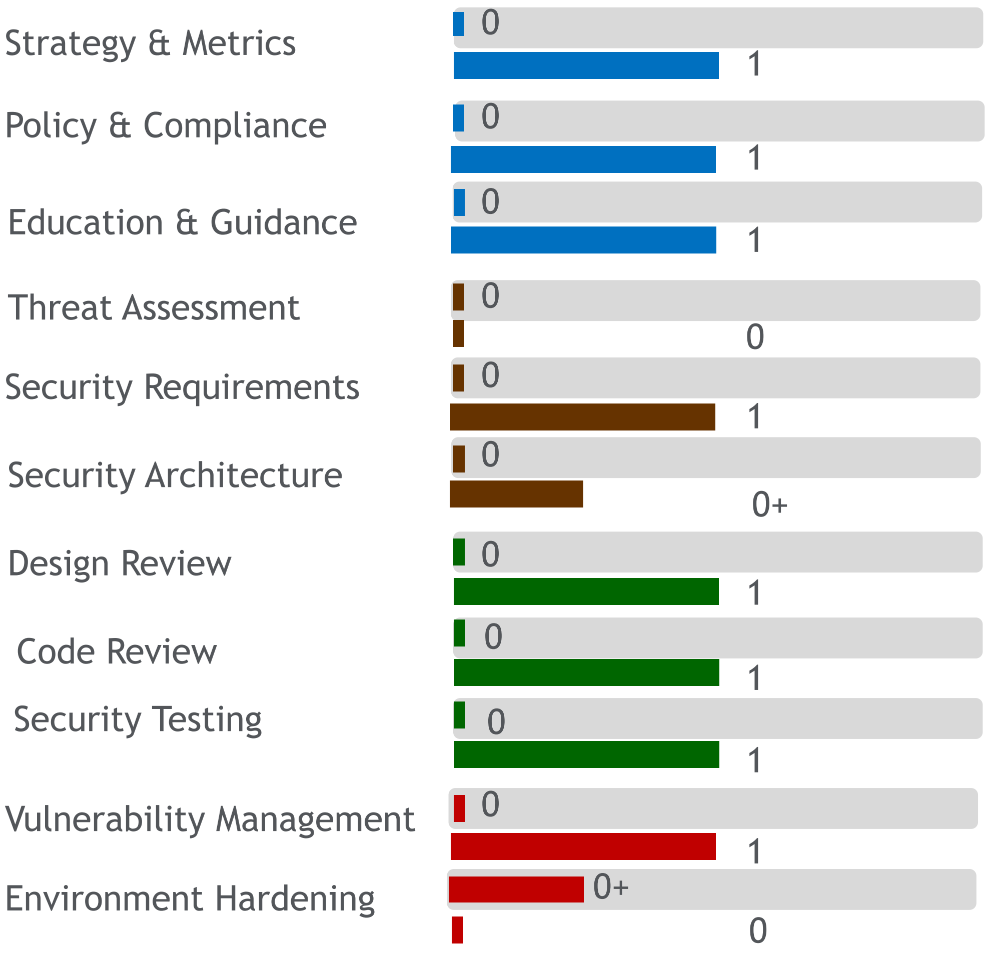
Current & Future State

Security Practices/Phase	Start	One	Two	Three	Four
Strategy & metrics	0	1	2	2	3
Policy & compliance	0	0	0+	1	2
Education & guidance	0	1	2	2	3
Threat assessment	0	0	1	2	2
Security requirements	0	1	1	2	3
Secure architecture	0	0	0+	1	1
Design analysis	0	0+	1	2	2
Code review	0	1	2	2	3
Security testing	0+	1	2	2	2
Vulnerability management	0	1	1	2	3
Environment hardening	0	0	0+	0	0
Operational enablement	0	0	1	2	3

Maturity Level Per Practice (+ = between levels)

0	Largely Absent	1	Adapt	2	Sustain	3	Master and Scale

Current State – Check Point



Software Security Maturity Assessment Services

Building a Software Security Program

Discovery

- Application Threat Assessment
- App Risk Portfolio
- Business Risk Profile
- Reporting

Planning & Awareness

- Build maturity roadmap
- Build project plan
- Socialize plan
- Awareness 101

Training & Testing

- Role based training
- Security Testing Practice
- Remediation Guidance

Infrastructure & Architecture

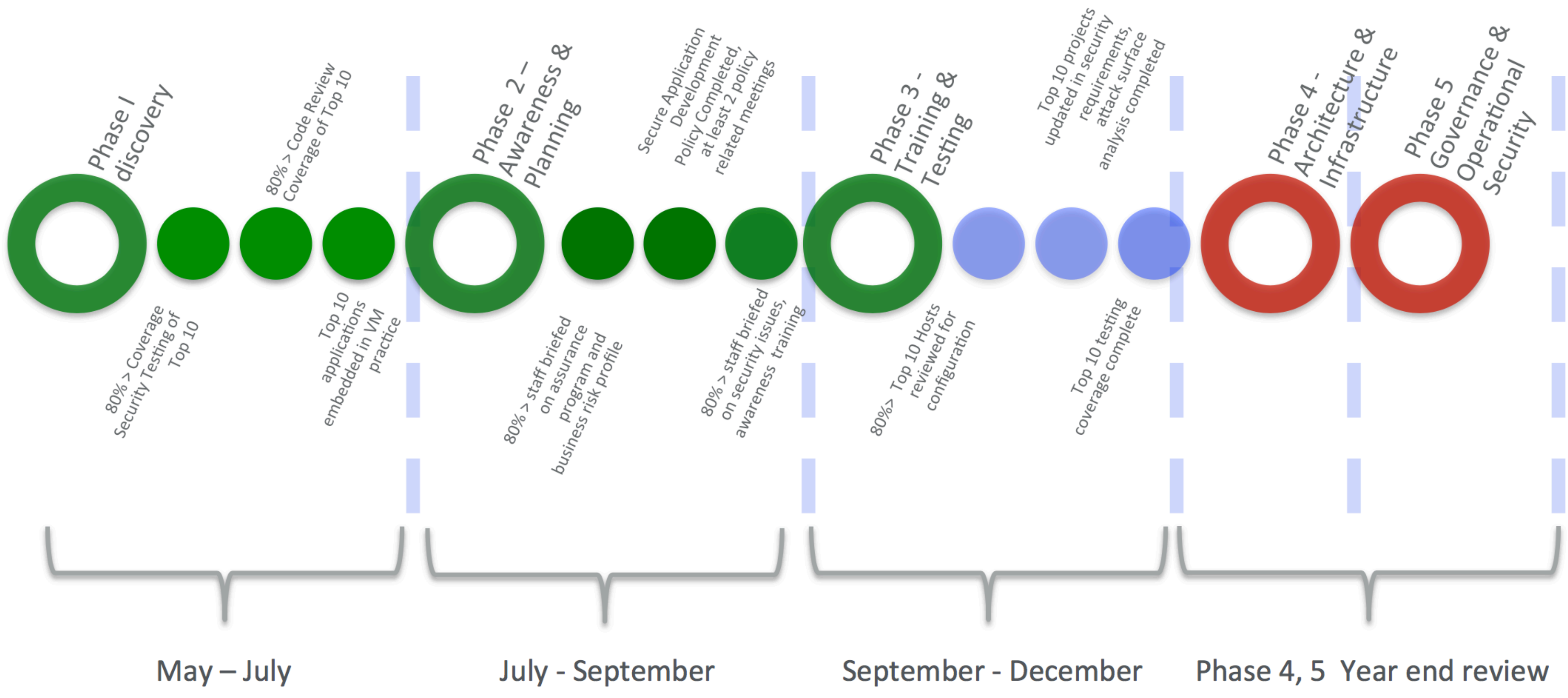
- Role based infra security training
- Security Testing Practice (inter/ ext)
- Secure architecture remediation Guidance

Governance & Security Operations

- Review policies and compliance
- Review strategy and metrics
- Change management control, DevOps

SSMA – Sample Score Card & Check Point

Building a Software Security Program


Phase I discovery
80% > Coverage Security Testing of Top 10
80% > Code Review Coverage of Top 10
Top 10 applications embedded in VM practice
Phase 2 – Awareness & Planning
80% > staff briefed on assurance program and business risk profile
Secure Application Development Policy Completed, at least 2 policy related meetings
80% > staff briefed on security issues, awareness training
Phase 3 - Training & Testing
80%> Top 10 Hosts reviewed for configuration
Top 10 projects updated in security requirements, attack surface analysis completed
Top 10 testing coverage complete
Phase 4 - Architecture & Infrastructure
Phase 5 Governance & Operational Security
May – July
July - September
September - December
Phase 4, 5 Year end review

● In Progress ● Success Metrics Achieved ● Success Metrics To Achieve for next phase ▬ Project checkpoint Dashboard update

Phase 2 – Awareness & Planning

Building a Software Security Program

Education & Guidance | Policy & Compliance | Strategy & Metrics

- Establish and share strategic software security roadmap
- Deliver 15 minute Security Brown Bags
  - Delivered by groups (Builder, Breakers, Defenders)
  - Sample topics:
    - ❑ Application Security Risks 101
    - ❑ PCI & The OWASP Top 10
    - ❑ PCI & SANS Top 25
    - ❑ The Secure Development Lifecycle
- Build SharePoint like knowledge base or repository to support security guidance
- Build Standards, policies (Secure Development Policy)
- Establish Project Audit Practice

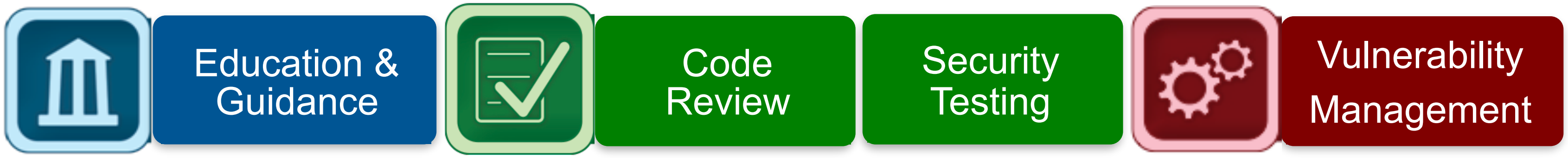
Phase 2 – Awareness & Planning

Building a Software Security Program

	Governance			Construction			Verification			Deployment		
	Strategy & Metrics	Policy & Compliance	Education & Guidance	Threat Assessment	Security Requirements	Secure Architecture	Design Review	Code Review	Security Testing	Vulnerability Management	Environment Hardening	Operational Enablement
PCI DSS 3.0	**SM2**	**PC2**	**EG2**	**TA1**	**SR2**	**SA2**	**DR3**	**CR2**	**ST2**	**VM3**	**EH3**	**OE3**
2.2.X		PC2			SR1						EH2	OE2
2.3		PC2									EH2	
3.X					SR1							
4.1		PC2									EH2	
4.2					SR1							
5.X		PC2			SR1						EH2	
6.1		PC2								VM1	EH1	
6.2		PC2									EH1	
6.3	SM2	PC2	EG1	TA1	SR1	SA2	DR3	CR2	ST2			OE2
6.3.1		PC2						CR1	ST1			
6.3.2		PC2						CR1	ST1			
6.4.X		PC2						CR1	ST1			OE2
6.5.X	SM2	PC2	EG1		SR1	SA1	DR3	CR2	ST2	VM1		
6.6									ST2	VM1		
6.7		PC2	EG1									
7.X		PC2			SR2							
8.1.X		PC2			SR1							
8.2.X		PC2			SR1							
10.1		PC2			SR2							
10.2.X		PC2		TA1	SR2						EH3	OE3
10.3.X		PC2			SR2						EH1	OE1
10.4.X		PC2									EH2	OE2
10.5.X		PC2			SR2						EH2	OE2
10.6.X		PC2									EH3	OE3
10.7.X		PC2			SR2						EH1	
10.8		PC2	EG1									
11.2.X		PC2									EH3	
11.3.X		PC2		TA1					ST2	VM2	EH3	
11.4										VM2	EH2	
11.5										VM2		OE2
11.6		PC2	EG1									
12.1		PC2	EG2									
12.2	SM1											
12.6			EG2									
12.10										VM3		

Phase 3 - Training & Testing

Building a Software Security Program


Education & Guidance

Code Review

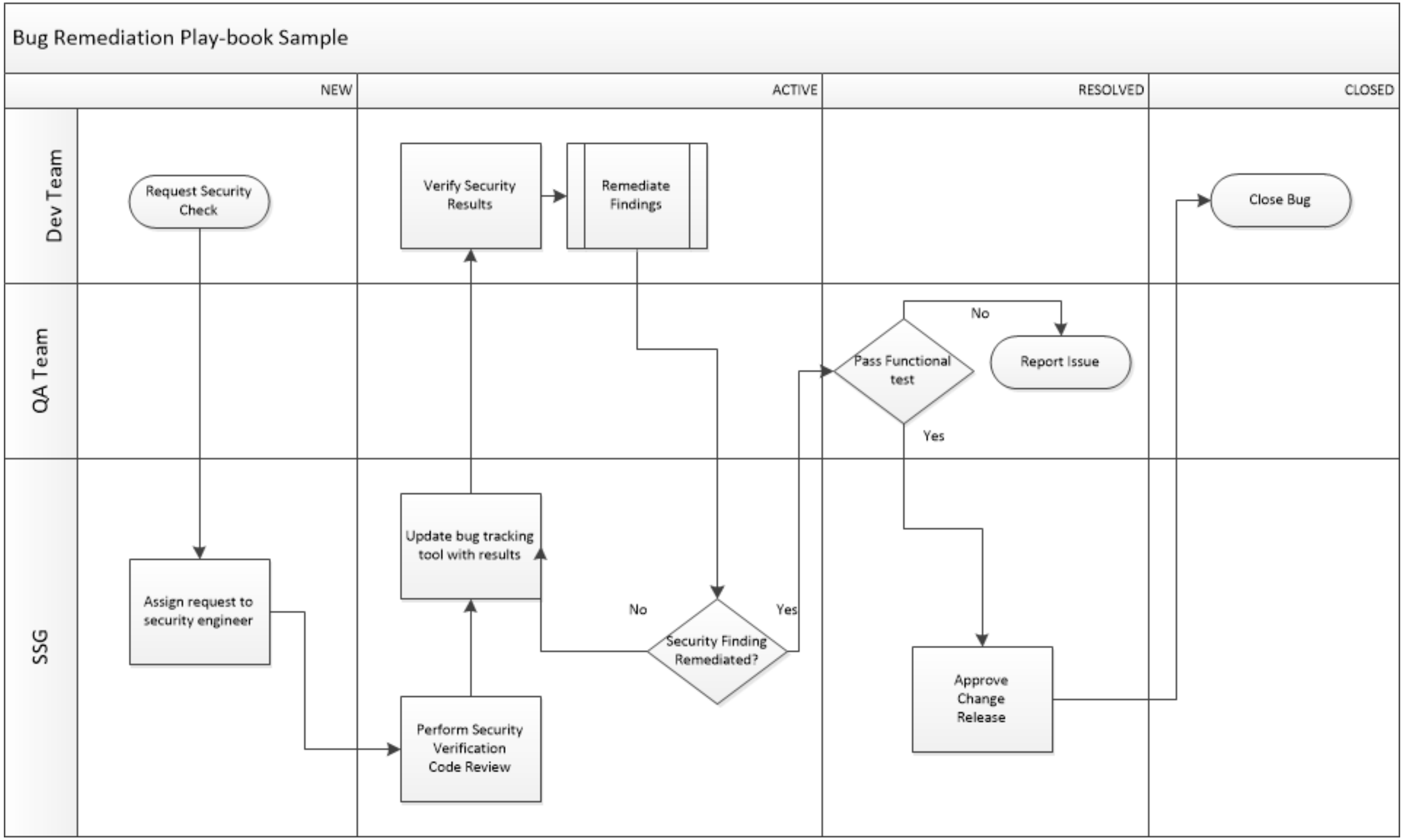
Security Testing

Vulnerability Management

- Continue Security Brown Bags
- Conduct role base “hands on” technical training
- Enhance remediation guidance
  - Testing Checklist (CR, WAPT, HCR)
  - Guidelines (WSC:.NET Cheat Sheets, Hardening Guides)
- Conduct Security Code Reviews of applications within application risk portfolio
- Conduct security tests of applications
- Establish point of contact and informal response team

Playbook

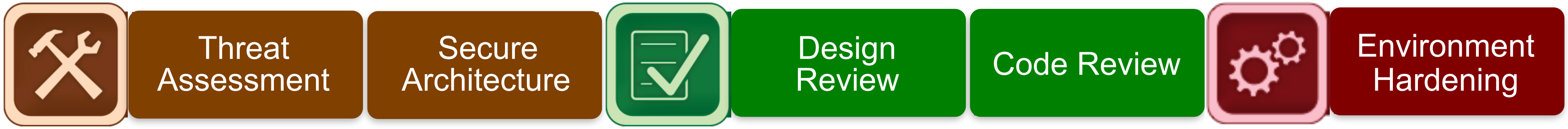
Taking a Strategic Approach to Enterprise Security



Bug Remediation Play-book Sample

Phase 4 - Infrastructure & Architecture

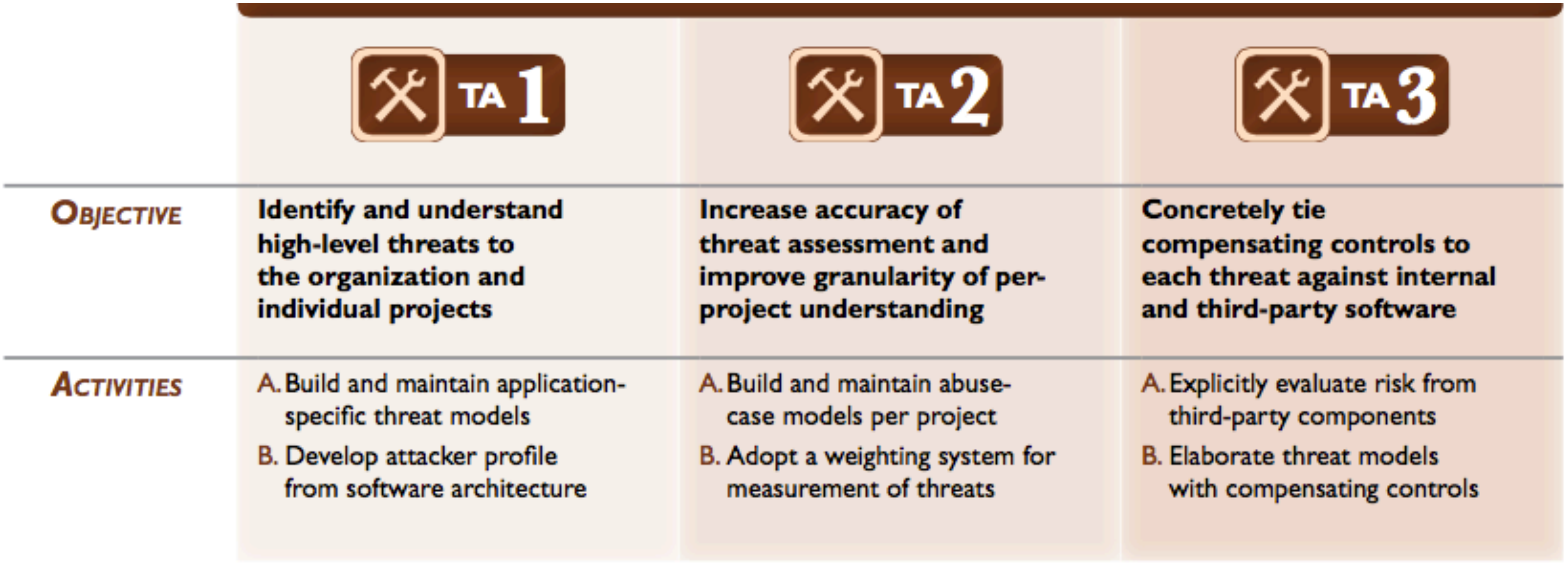
Building a Software Security Program



- Build Threat Assessment practice
- Conduct Threat Assessments per project base
- Provide secure architecture design guidance and support
- Document and align security requirements per project code base
- Build Design Review Practice per project code base
- Expand and continue Code Review and WAPT practice

What does maturity look like?

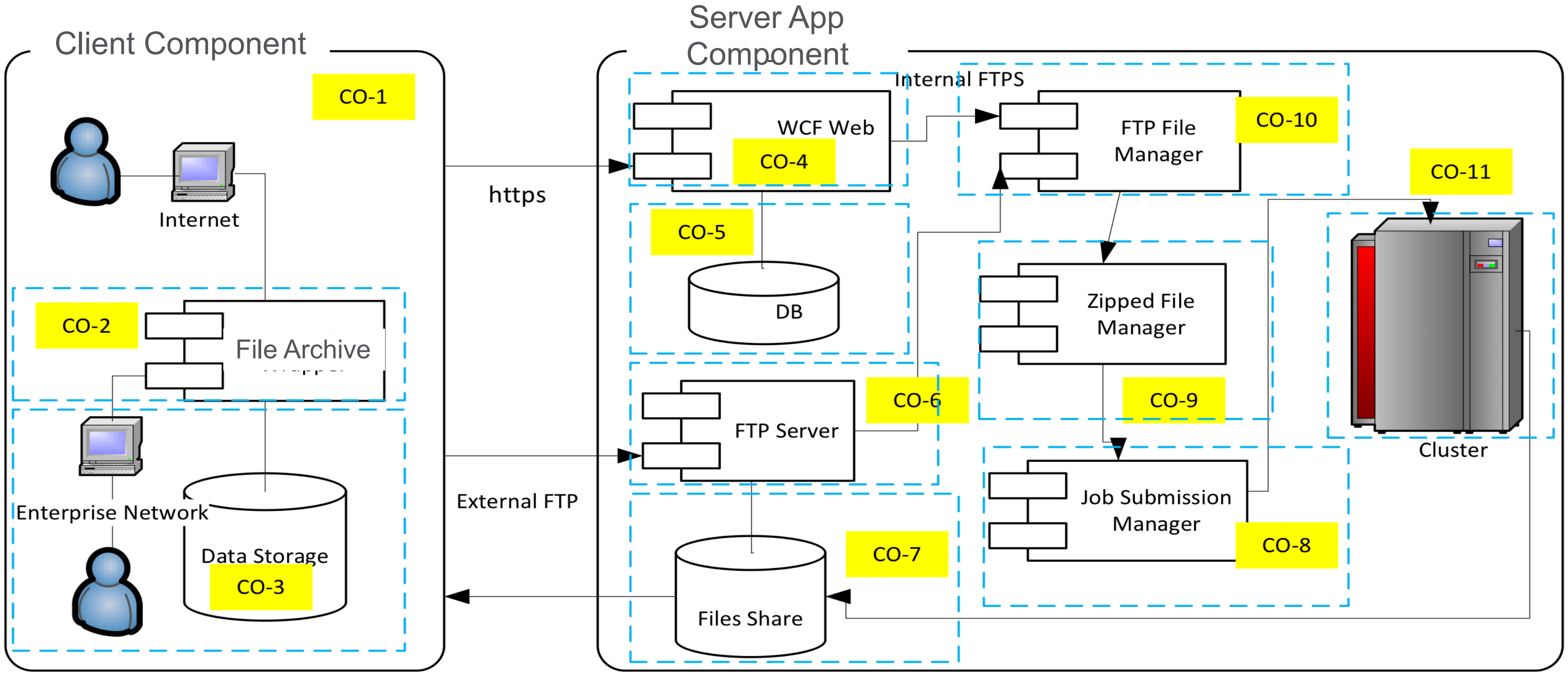
Taking a Strategic Approach to Enterprise Security



	TA 1	TA 2	TA 3
OBJECTIVE	**Identify and understand high-level threats to the organization and individual projects**	**Increase accuracy of threat assessment and improve granularity of per-project understanding**	**Concretely tie compensating controls to each threat against internal and third-party software**
ACTIVITIES	A. Build and maintain application-specific threat models B. Develop attacker profile from software architecture	A. Build and maintain abuse-case models per project B. Adopt a weighting system for measurement of threats	A. Explicitly evaluate risk from third-party components B. Elaborate threat models with compensating controls

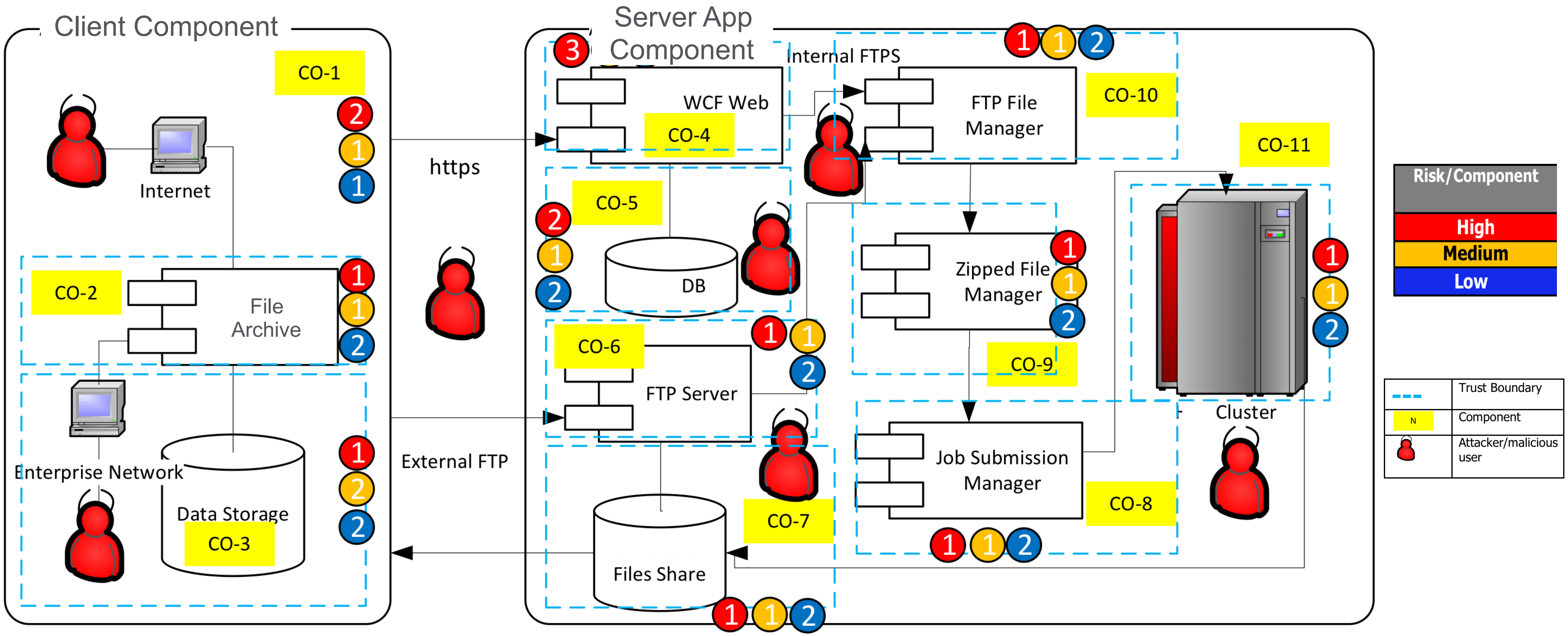
Threat Assessment Practice

Taking a Strategic Approach to Enterprise Security



Threat Assessment Practice

Taking a Strategic Approach to Enterprise Security



Phase 5 - Governance & Security Operations

Building a Software Security Program

Strategy & Metrics

Operational Enablement

- Document metrics for security expenditure
- Conduct industry wide cost comparisons
- Coordinate and enhance code release and relevant change manage procedures
- Maintain formal operational security guides

Some Success Metrics

Building a Software Security Program

- 80% of applications in compliance with policies and standards
- 80% of staff knowledgeable about policies and standards

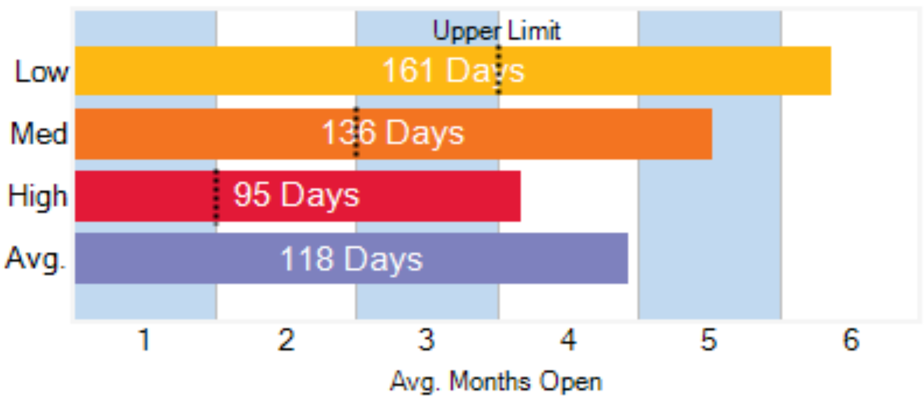
- 80% CR code coverage for Top 10 software applications
- 85% of projects updated with security requirements and design analysis

- 80% of stakeholders aware of threats per project code base
- 80% of code base projects covered by security requirements
- 80% Vendors briefed on security requirements and agreements

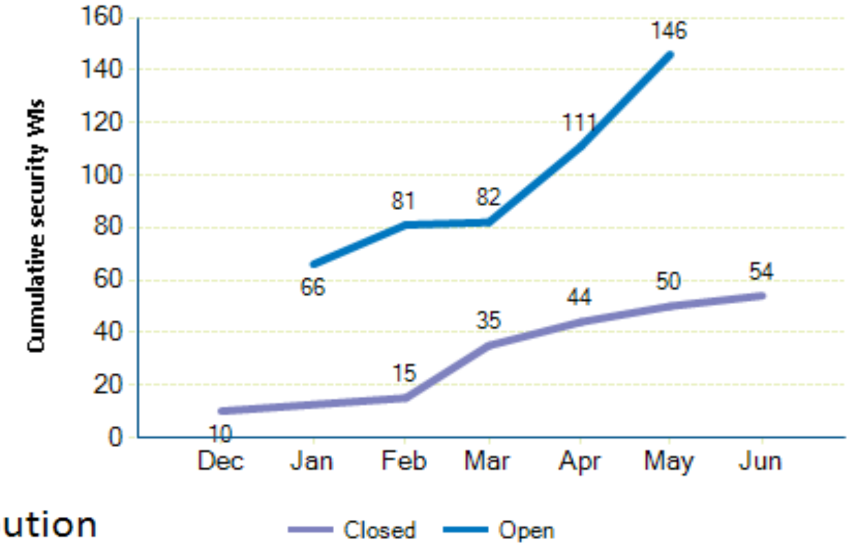
Some Success Metrics

Building a Software Security Program

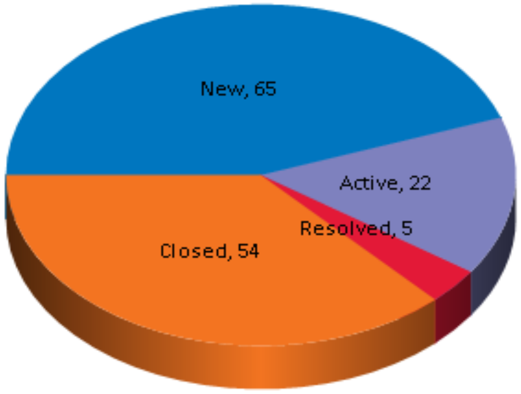
Security Bug Latency



Security Bugs - Total vs. Closed



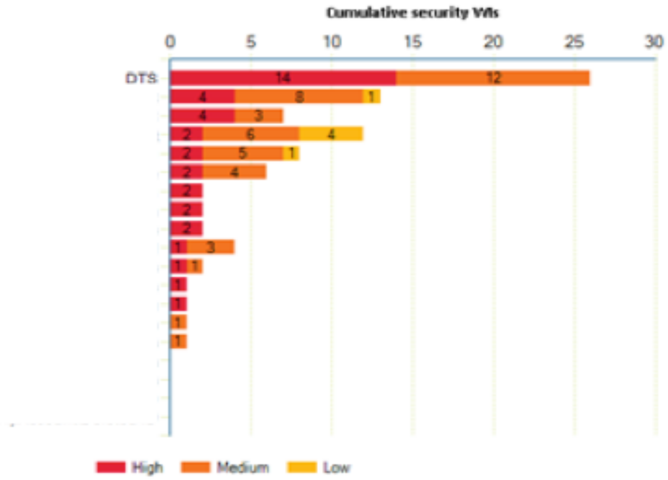
Security Bugs Status Distribution



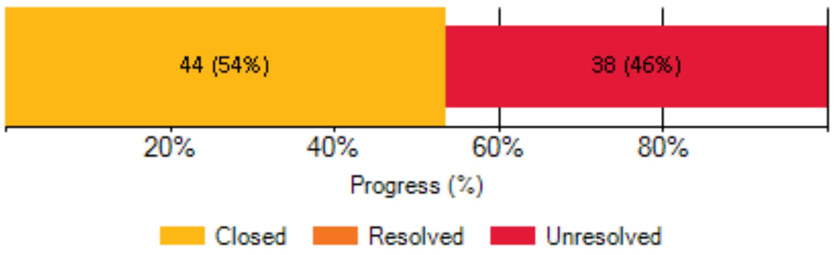
Some Success Metrics

Building a Software Security Program

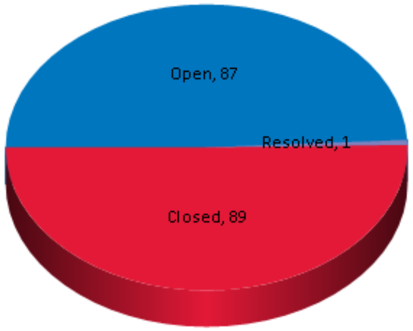
Security Bugs By Product



PCI Compliance (DSS 3.0)



Security Bugs Distribution (2014)



Summary

Building a Software Security Program

- SSMA Methodology
  - Governance, Construction, Verification and Deployment
  - 3 maturity levels
  - SDL Gap Analysis followed by in depth audit
- Case Study (SSM Execution)
  - Awareness & Planning
  - Training & Testing
  - Infrastructure & Architecture
  - Governance & Operational Security
- SSMA Key Benefits
  - Comparison of current SDL activities vs. best practices
  - Cost effective guided approach supported by check points to ensure positive direction
  - A flexible plan to apply

Foundstone®

intel® Security