Secure SDLC: The Good, The Bad, and The Ugly

Joey Peloquin, Director, Application Security
FishNet Security
• Secure Development Programs
  – The Good, The Bad, and The Ugly
• QSA Perspectives
  – Application Security in a *PCI World*
• Secure SDLC
  – The Essential Elements & Where to Start
• Post-Mortem
  – A Flawed “AppSec” Program Made Right
• Q & A
Secure Development Programs
THE GOOD
• Top -> Down Support
• Clearly Defined Processes
• Focus on Training and Education
• Security is a Function of Quality Management
• Properly Leveraging Technology
• Third-party Partnerships
• Go – No-Go Authority
• Working **Smarter**, Not **Harder**
The Bad
- Insufficient Support from Management
- Reactive Security Posture
- Check-in-the-box Mentality
- Insufficient Vulnerability Management
- No Developer Training
- Lack of Application Security Awareness
- Insufficient Standardization
- Development Silos
Complete Lack of Management Support

Devoid of Security Awareness

“Wow, there’s organizations devoted to Application Security that offer free information, tools, and standards?”

Complete Lack of Vulnerability Management

Little Standardization

No Quality Management

Pattern of Denial
“I’m concerned that as long as the payment card industry is writing the standards, we’ll never see a more secure system. We in Congress must consider whether we can continue to rely on industry-created standards, particularly if they’re inadequate to address the ongoing threat.”

- Rep. Bennie Thompson
Elements of a PCI Compliant Program

• Security Throughout the Lifecycle
  – Requirements, checkpoints, accreditation, testing
  – No concept of OWASP, inability to examine code for common defects, no peer reviews, etc.

• Well-documented and Maintained SDLC
  – I’m from Missouri…

• Knowledgeable Developers
  – Coding examples, processes

• Peer Reviews
  – Someone other than the dev; examine comments
- Homegrown Encryption
  - Publically available, commercial/open source
- Code Reviews
  - No, you can’t review your own…
- Look at the Pretty WAF!
  - Yes, it has to actually be configured to block, /sigh
- “We have a WAF, so we don’t need to fix our code.”
- “Our IPS can totally block SQLi and XSS!”
Section 6.6 Compliance

- **WAF**
  - Network diagrams
  - Configuration
  - Logging

- **Code Reviews**
  - Documented policy, process, methodologies
  - Reports
  - Internal or third-party?
  - Tester’s role
  - Tester’s credentials
ALL OF THESE PARENTHESES AND SEMICOLONS TOTALLY RUIN THE FENG SHUI OF THE CODE.

SALLY LEARNS AN IMPORTANT LESSON ABOUT INVITING THE RIGHT PEOPLE TO HER CODE REVIEWS.
Essential Elements

- Executive Champion
- Mid-level Support
- Support of The Business
- People
- Process
- Technology
- …and unfortunately;
  - Time & Money help a great deal
The only valid measurement of code quality: WTFs/minute

Good code.

Bad code.
Where to Start?

- Assess your current maturity level
- Identify Business and Security Objectives
- Plan your work and work your plan!
- Document your approach
  - Who, what, when, where, how?
- Dr. McGraw’s Touchpoints:
  - Code Reviews (Static Analysis)
  - Risk Analysis
  - *Skills Assessment and Training*
  - Penetration Testing (Dynamic Analysis)
Scale of Maturity

Increasing Maturity

Security Unaware
- No documented Application Security practices
- No internal testing, merely annual penetration test
- No application security awareness or developer training

Reactive Security
- Standards-based internal processes lead to a basic level of awareness
- Some manual testing, looking into automation
- Recognize need for application security, but don’t know where to start

Proactive Security
- Champion and stakeholders identified
- Policies, standards & processes established
- Tools evaluated and purchased
- Automated and manual internal testing
- Developer training and awareness

Security Fitness
- Security baked into SDLC, discussed during design phase
- Security checkpoints defined and enforced
- Centralized, reusable resources for developers
- Centralized testing and remediation tracking
- Development mentors identified and trained

Sustained Maturity
- Centralized People, Processes and Technology
- Application security integrated seamlessly into quality lifecycle, becoming third pillar
- Application security team has Enterprise influence
- Security addressed throughout SDLC and applied retroactively to legacy applications

Decreasing Overall Development Cost
Post-Mortem: A Flawed Attempt at Building Security In...
Mistakes / Issues *(Opportunities?!)*

- Lost executive champion
- Lack of mid-level support
- Staff Reorganization
- No business support
- No defined processes
- Not enough expertise
- Development silos
- Shelfware
Putting the Pieces Back Together

- Educate *The Business*
- Security Requirements
- Define Standards
- Define Processes
- Development Mentors
- HP AMP – SaaS
- Offensive Security
  - License to Pen-test