Building Bridges between Dev and AppSec

SANS AppSec 2011
The Problem

- Building secure software is hard
- Development has to take ownership for writing secure code
- Developers need help to do this – from the security community
- But there are serious gaps between Dev and AppSec
- How can we bridge these gaps?
Why I’m Here

- 20+ years in software development and managing development teams
- Specialist in financial trading platforms for exchanges and investment banks
- I manage small teams with big customers
- I want to find better ways for small teams to build good, reliable, secure software

- Blog in my spare time about the challenges that small teams face building real software
The Gaps

- **Priority Gap**
  - Features
  - Cost/Time to Market
  - Connectivity and Integration
  - If we succeed: performance, reliability, compliance, ...
  - Security?

- **Knowledge Gap**

- **Language Gap**
  - XSRF, reflected/persistent XSS, XST, XSSI, 0-day exploit, XPath injection, pwning, session hijacking, SQL smuggling... attacks and hacker-talk

- **Quality Gap – the Attacker’s Advantage**

- **Ownership Gap and Capability Gap**
Independent Communities
AppSec has to be Simpler

- Education and awareness: developers can’t prevent or fix what they don’t understand
  
  OWASP 2011 Summit: “Developers don’t know shit about security”

- Break security problems down

- Pragmatic practices: fast, simple, work under pressure – for small Agile teams

- Make it harder to write bad code
We need better tools

- It’s too easy to make security mistakes
- Frameworks and languages aren’t secure
- Instead of lists of do this, don’t do that:
  Ivan Ristic: “Design platforms, libraries and components in such a way that vulnerabilities cannot exist. Then use them.”
- Better tools to find problems in legacy code
  - Too many dups, too many FPs, too many FNs
- Too expensive – to license, to get working
- We need a stronger Open Source community
- WAFs aren’t the answer either
- Nothing is good enough, today…
Software Security is 50/50

- “Soft” problems – developers
  - Data validation
  - Exceptions and error handling
  - Race conditions, locking, concurrency
  - Bad coding, stupid mistakes: overflows, null pointers, leaks, arithmetic errors, debugging code..
  - Good layered architecture

- “Hard” problems – security experts
  - Secure design: authentication, authorization, …
  - Crypto and secure protocols
  - Technology-specific security problems, research…
Who Owns the Security Problem?

I discovered a hole in our internet security. What?!

Good grief, man! How could you put a hole in our internet? I didn't put it there. I found it... and it's not.

It's your job to fix that hole. I want you to work 24-7!

Actually, that's not my job. But I'll inform our network management group.

Passing the buck!!! You're a buck passer!!!

Forget it! There's no hole! It got better!

That's more like it.

I fixed the internet.
Taking Ownership Together

- Developers aren’t victims
  - We can and should own problems
  - But we need help… from AppSec
  - We have to bridge the gaps

- Like Dev and QA communities 10+ years ago
  - Independent test groups, walls and gates
  - Better results from opposed interests – objective
  - Expensive tools, manual testing, checklists
  - XP and Agile changed all this
Building Bridges

- Look at DevOps
  - Community leaders taking on problems together
  - Writing better tools and using them
  - Hold conferences together: Velocity…
  - Lightweight, practical, collaborative – every level

- OWASP Developer Outreach
  - OWASP is trying to connect with developers
  - Builders and Defenders communities