Tips for Building a Successful Application Security Program

Reducing Software Risk

OWASP

The Open Web Application Security Project

Denver Chapter

January 16, 2013

About Me

OWASP
The Open Web Application Security Project

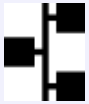
- Dave Ferguson
  - Veracode Solutions Architect
  - Certifications: CISSP, CSSLP
- In the Past
  - Principal Consultant with FishNet Security
  - Software Applications Developer
- Industry Involvement
  - OWASP
    - Member, Contributor, Former KC chapter leader
    - Forgot Password Cheat Sheet
  - Blog
    - http://appsecnotes.blogspot.com

VERACODE
Securing the Software That Runs the World

State of Software Security?

applications

end points



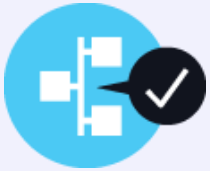
network

data

applications

end points

$35 billion spent in 2011



network

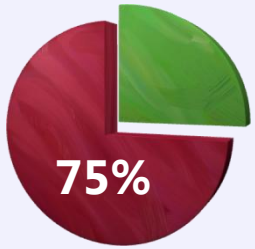
data

Applications Are Complex

OWASP
The Open Web Application Security Project

Why Is Software a Target?

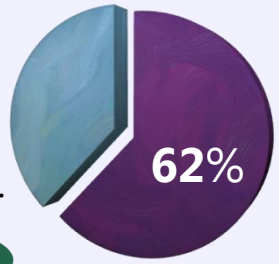
- $300 billion in software produced or sold each year
- One of the world's largest manufacturing industries
- No uniform standards or insight into security risk or liability of the final product



75% percent of attacks take place at the application layer.

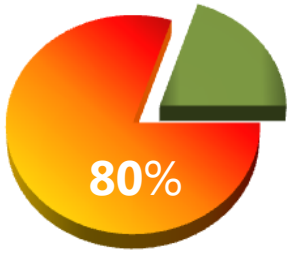
Gartner

62% of companies experienced security breaches in critical applications within the last year.



FORRESTER

58%



Upon first test, 8 in 10 applications contained XSS and/or SQL injection flaws.

VERACODE

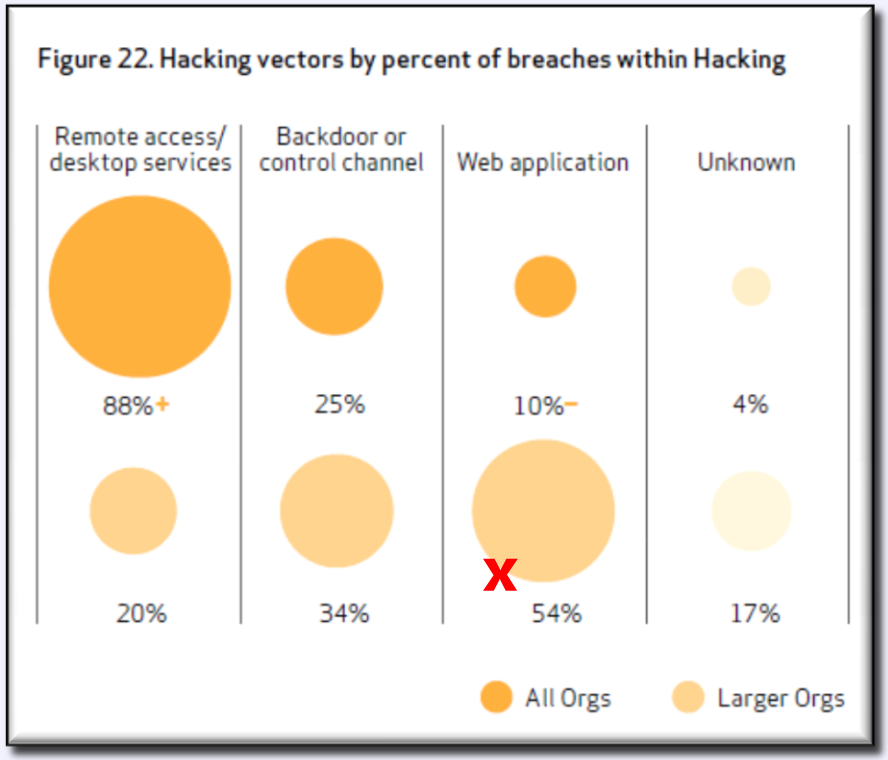
OWASP
The Open Web Application Security Project

"Developers and defenders need to be right every time. An attacker only has to be right once."

OWASP
The Open Web Application Security Project

Big Companies Are Hacked via Web Apps

HOW DO BREACHES OCCUR?	
81%	utilized some form of hacking (+31%)
69%	incorporated malware (+20%)
10%	involved physical attacks (-19%)
7%	employed social tactics (-4%)
5%	resulted from privilege misuse (-12%)

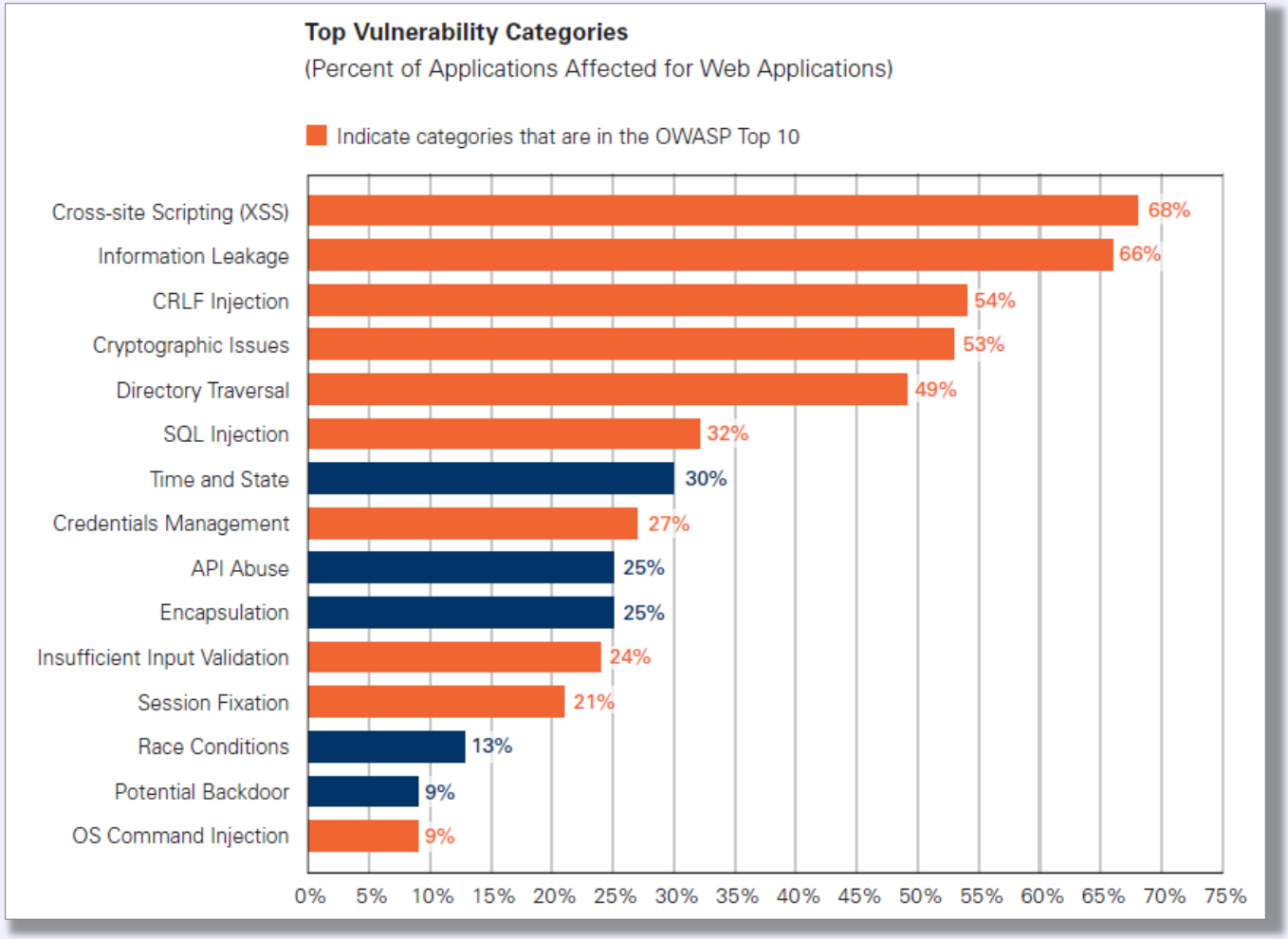


Source: Verizon Business 2012 Data Breach Investigations Report

OWASP
The Open Web Application Security Project

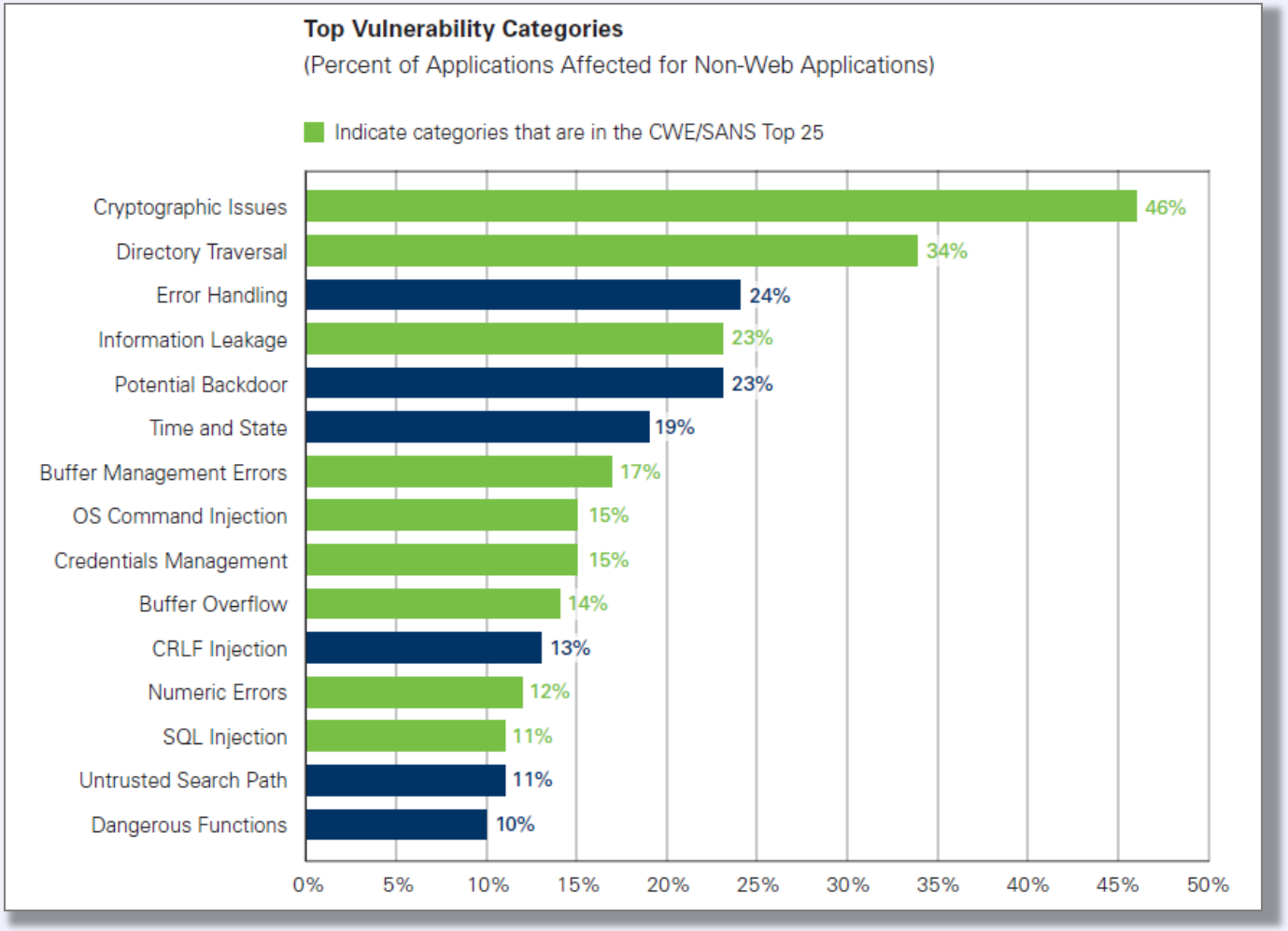
What kind of application vulnerabilities are out there?

Web Apps



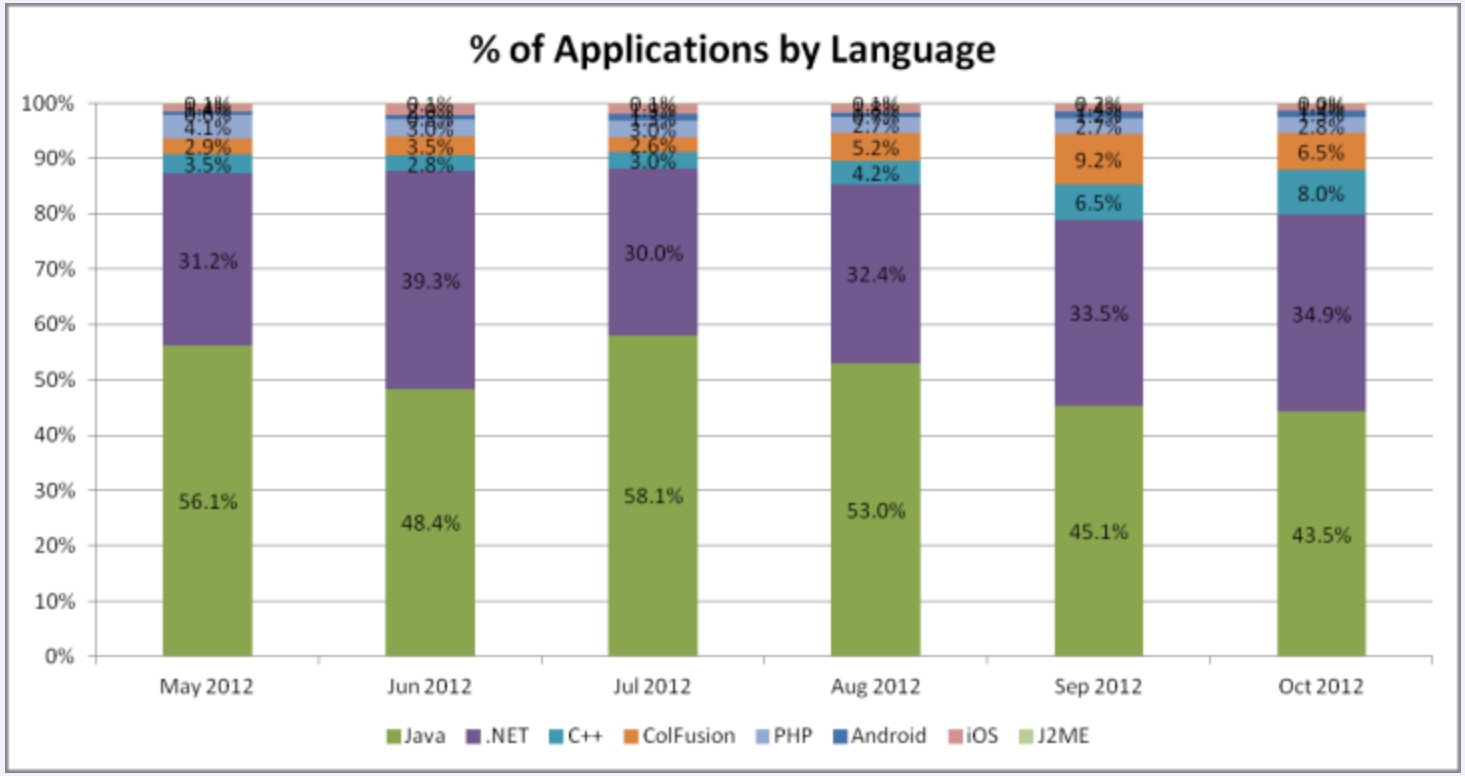
Source: Veracode State of Software Security Report, Volume 4

Non-Web Apps

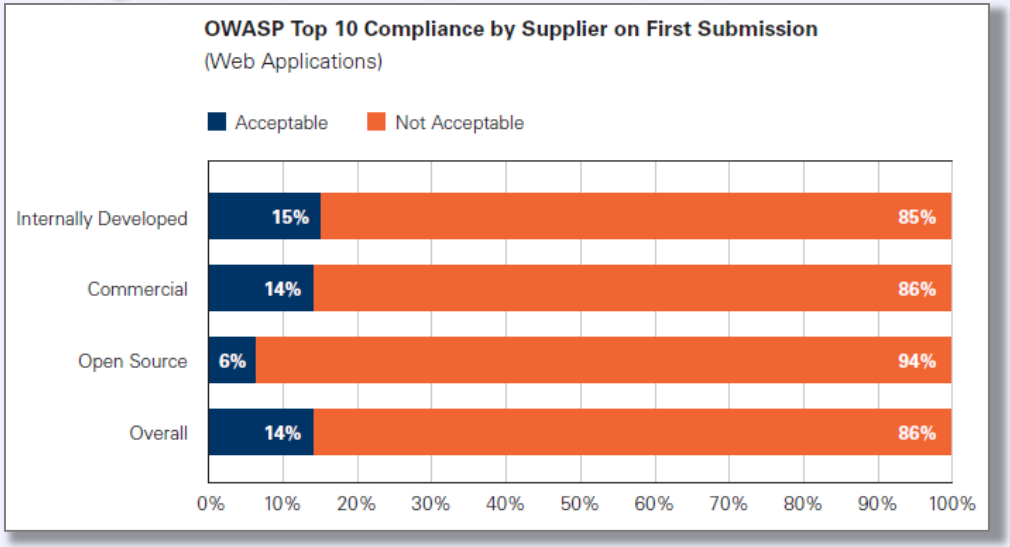


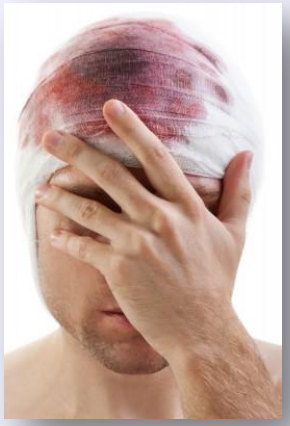
Source: Veracode State of Software Security Report, Volume 4

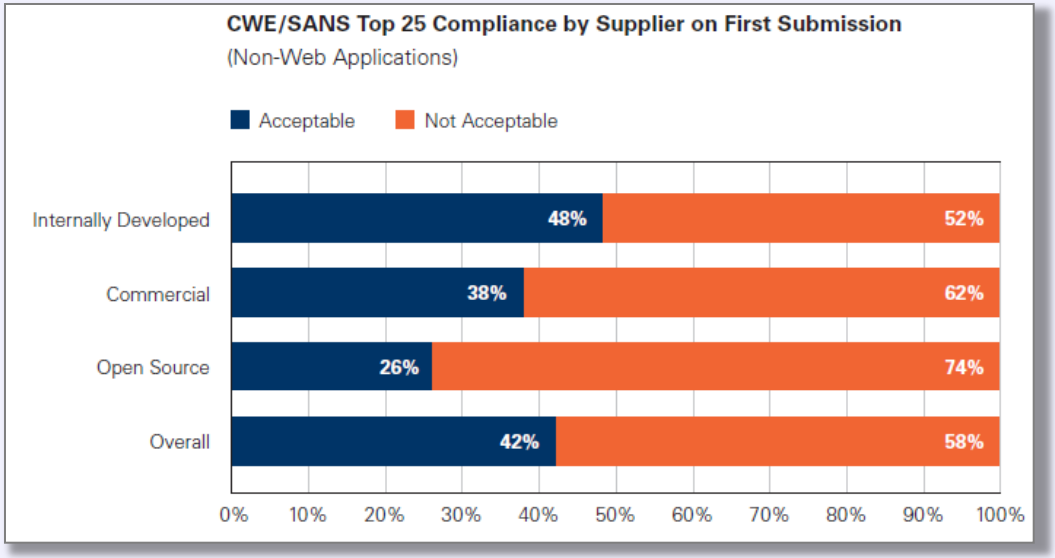
Programming Language Breakdown



OWASP
The Open Web Application Security Project







Source: Veracode State of Software Security Report, Volume 4

OWASP
The Open Web Application Security Project

Targeting the Software Supply Chain

NEW TREND

Lockheed says Cyber Attacks Up Sharply, Suppliers Targeted

By Andrea Shalal-Esa | Reuters – Mon, Nov 12, 2012

The Information Security Forum Announces Top Five Security Threats in 2013

Posted November 29, 2012

2. Supply Chain Security ←

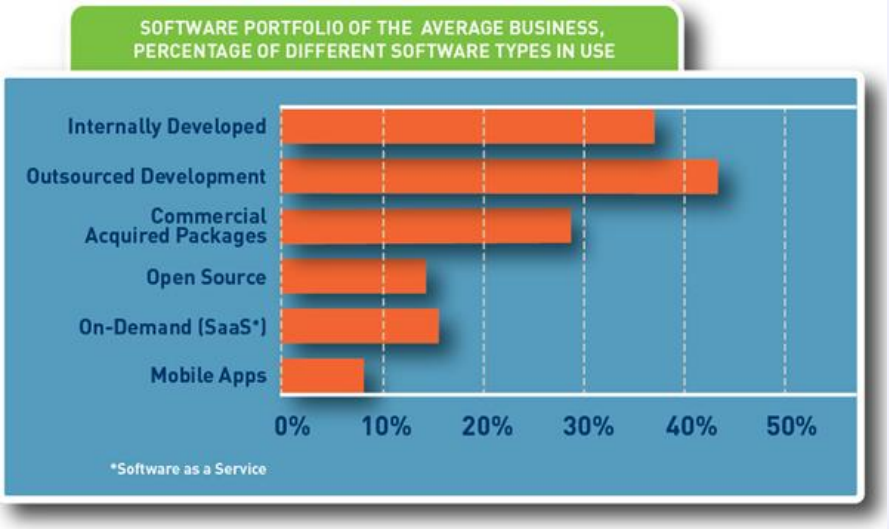
More organizations will fall victim to information security incidents at their suppliers.

Gartner Study Questions Integrity Of IT Supply Chain

By Ken Presti

October 18, 2012 8:04 PM ET

The Risks of Third-Party Software



"SOUP" : Software of Unknown Pedigree

Challenges

- ✓ Managing application sprawl
- ✓ Understanding criticality / business risk
- ✓ Choosing assessment technique
  - ✓ Static analysis?
  - ✓ Dynamic analysis?
  - ✓ Manual pen testing?
- ✓ Cost – tools & labor
- ✓ Tracking/measuring progress

And then there's...

- ✓ Lack of executive support
- ✓ Expertise required to run tools
- ✓ Turnover
- ✓ 3rd party / vendor apps
- ✓ Mobile apps
- ✓ Deciding which flaws to fix
- ✓ Friction between dev teams & security
- ✓ Communication inefficiencies
- ✓ Lack of secure coding skills
- ✓ Siloed development teams
- ✓ Remediation effort / re-testing

OWASP
The Open Web Application Security Project

The Solution?

OWASP
The Open Web Application Security Project

Application Security Program

- ✓ Centralized
- ✓ Policy-Driven
- ✓ Comprehensive

The Four Pillars

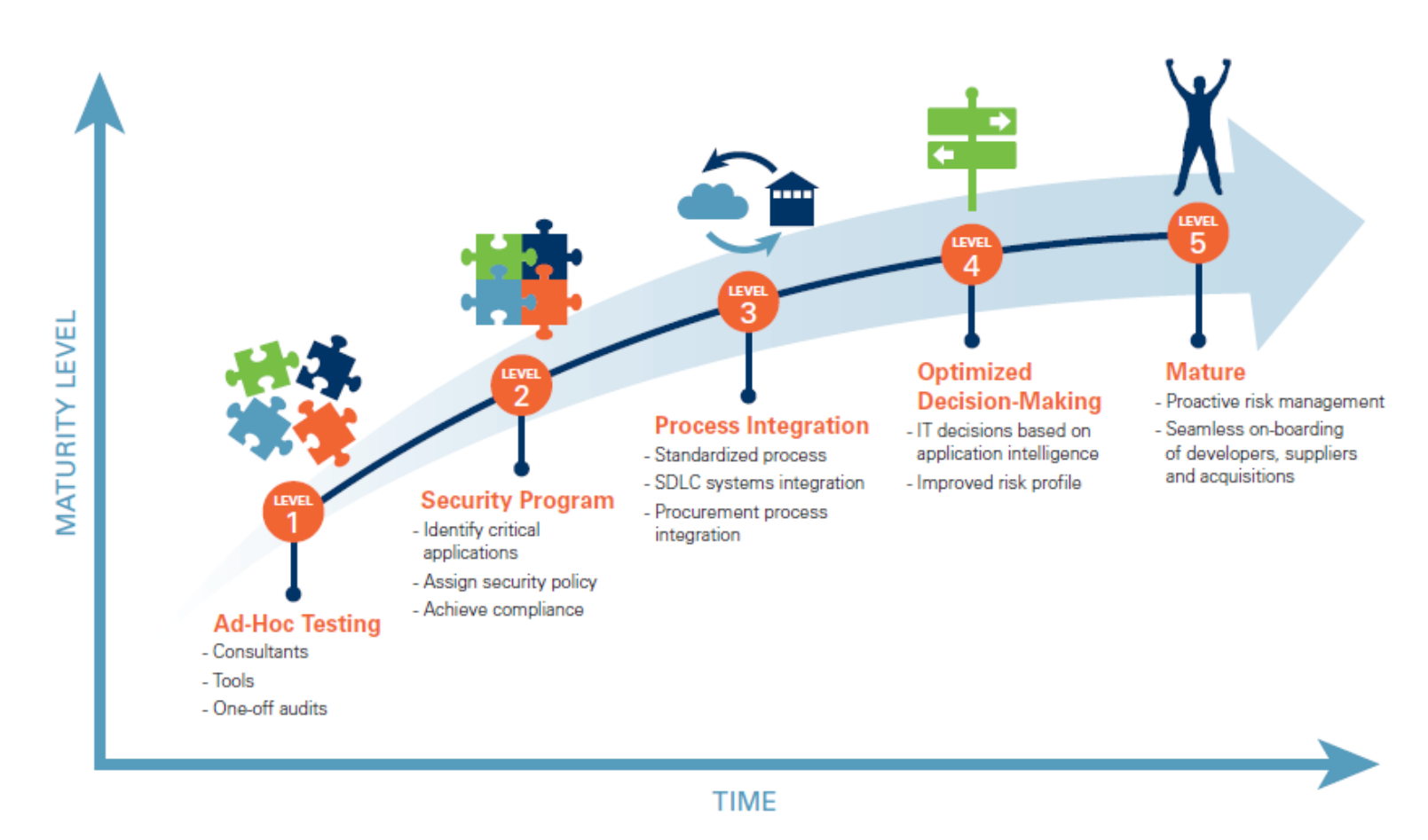
Policies, Management

Scans, Pen Testing, Remediation

Developer Training

Metrics

Phased Implementation



Tips for a Successful Program

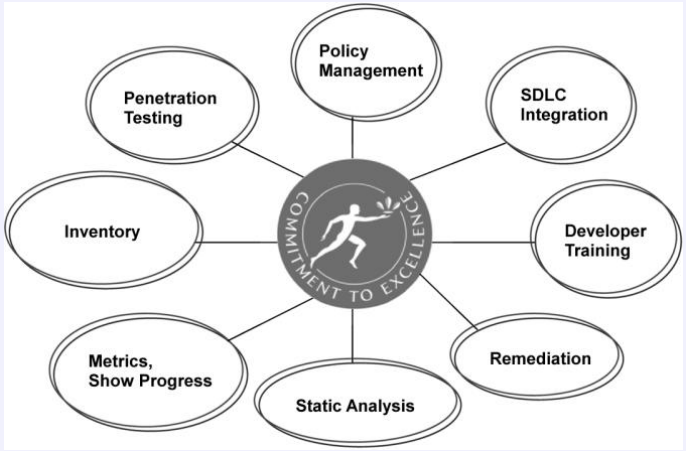
- Executive sponsorship
- Center of Excellence approach
- Application inventory and classification
- Use of security policies
- Have defined roles
- Automated scans (static & dynamic)
- Manual penetration testing
- Metrics & reporting
- Automation / integration into SDLC
- Remediation
- Validate 3rd-party & outsourced software
- Developer training
- Collaboration with devs (avoid scan & scold)

Executive Support

- ✓ Essential for a successful program
- ✓ Approves corporate resources
- ✓ Sets software security as a priority
- ✓ Validates the business risks
- ✓ Assures that acceptable security is achieved within release timelines

Center of Excellence

- ✓ Mission: To establish a standard for ensuring software security across the organization
- ✓ Formalize elements of the program
- ✓ Consistent approach across risk domains
  - ✓ Internally-developed
  - ✓ Outsourced
  - ✓ Purchased
  - ✓ Open source
  - ✓ Mobile apps



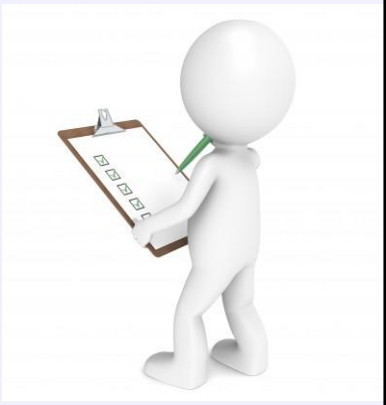
Establish Security Policies

- ✓ Assign different policies for different business risk
- ✓ Remediation requirements naturally fall out
- ✓ Don't require perfect code!
  - Consider standards such as OWASP, SANS Top 25, or PCI
  - Consider severity rating or specific CWEs
- ✓ Specify required testing techniques and test frequency

Enforce Policies

- ✓ Instill accountability for policy compliance with application owners
- ✓ Define escalation paths

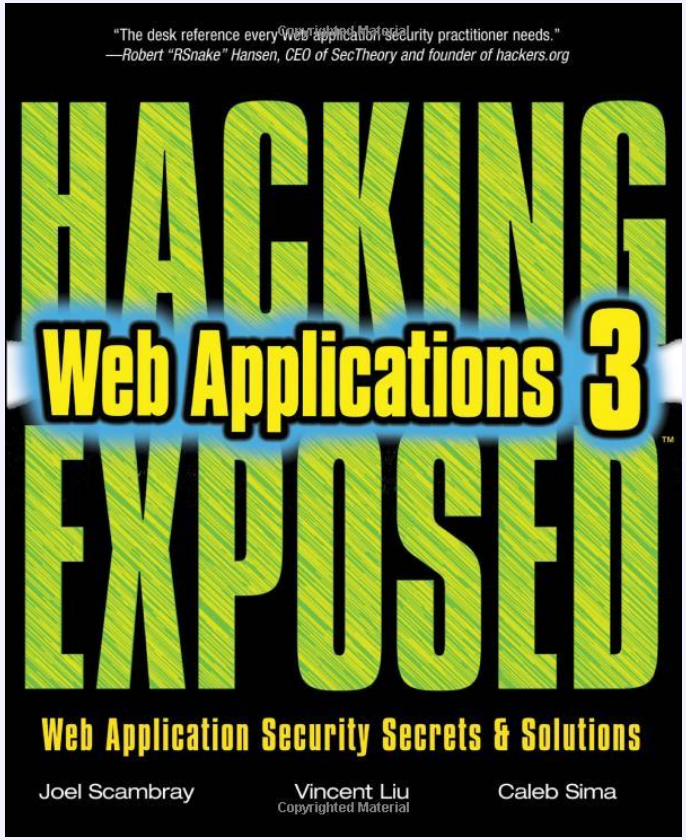
Define Roles



- ✓ Need clear distinction between management and operational roles
- ✓ Who establishes appropriate security policies?
- ✓ Who determines the business criticality of the apps?
- ✓ Who decides the proper policy to assign to an app?
- ✓ Who is responsible for testing the app?
- ✓ How are conflicts resolved?
- ✓ Who monitors the program and reports to executive management?

Test Third-Party Applications

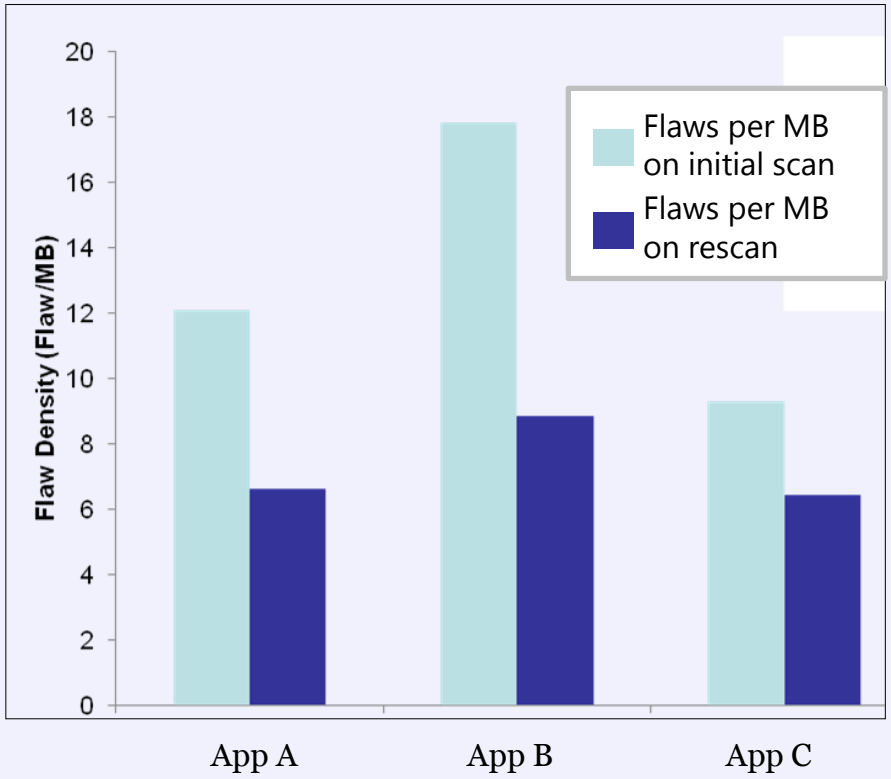
- ✓ Hold vendors, contractors, and outsourcers to the same standards as internal development teams
- ✓ Employ independent verification
- ✓ Pre-define security policy so all parties understand acceptance criteria

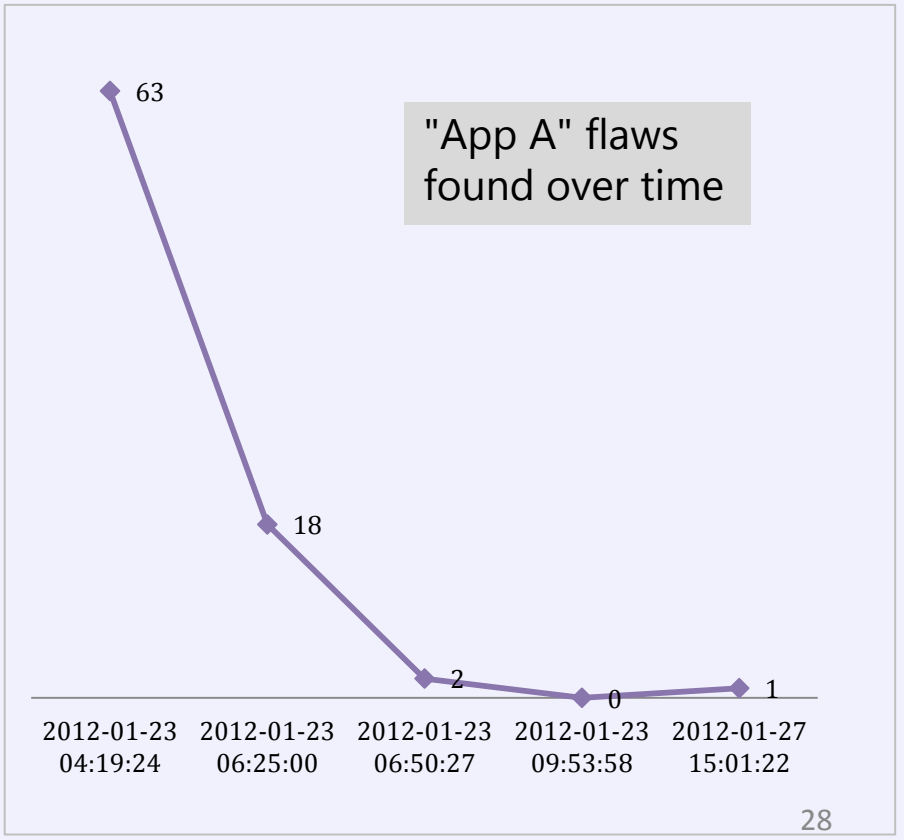


http://www.amazon.com/HACKING-EXPOSED-WEB-APPLICATIONS-Edition/dp/0071740643/

OWASP
The Open Web Application Security Project

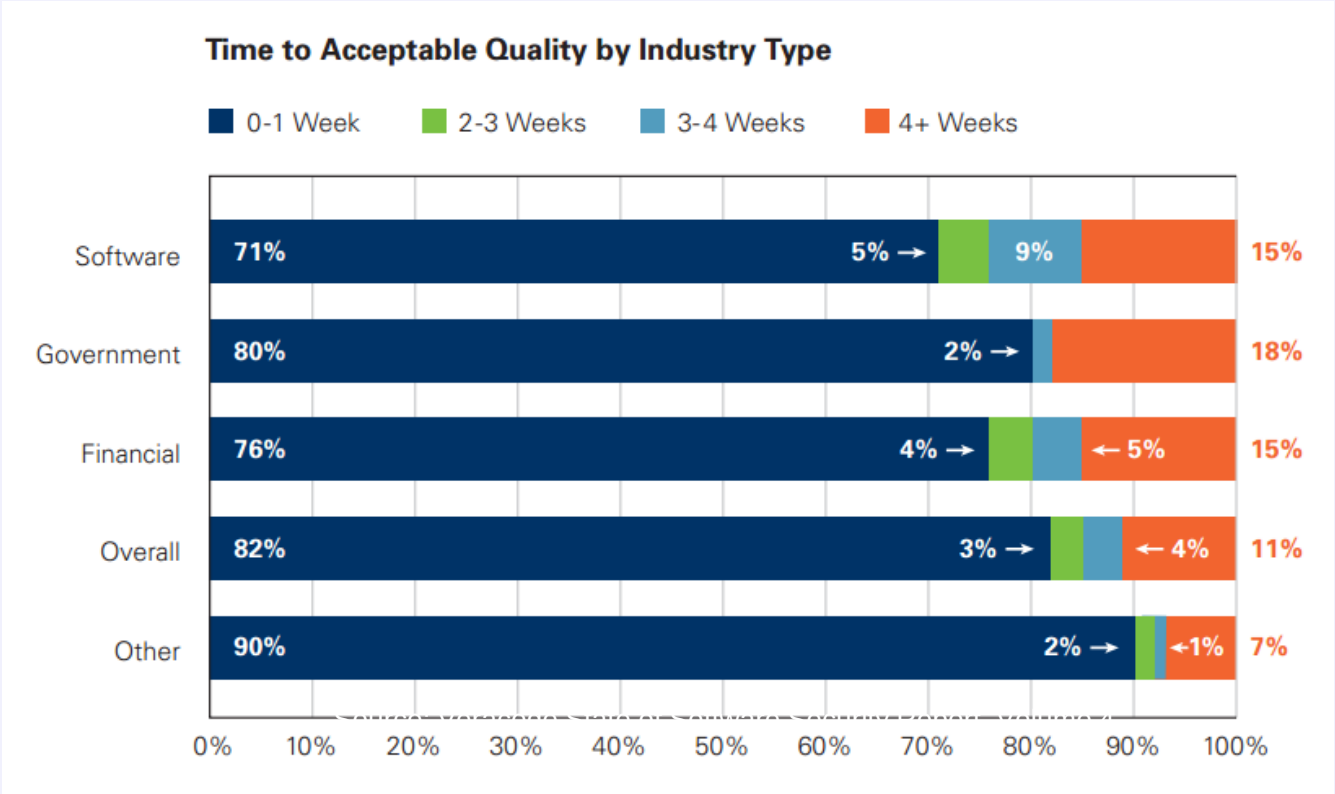
Case Study:
Metrics show rapid improvement



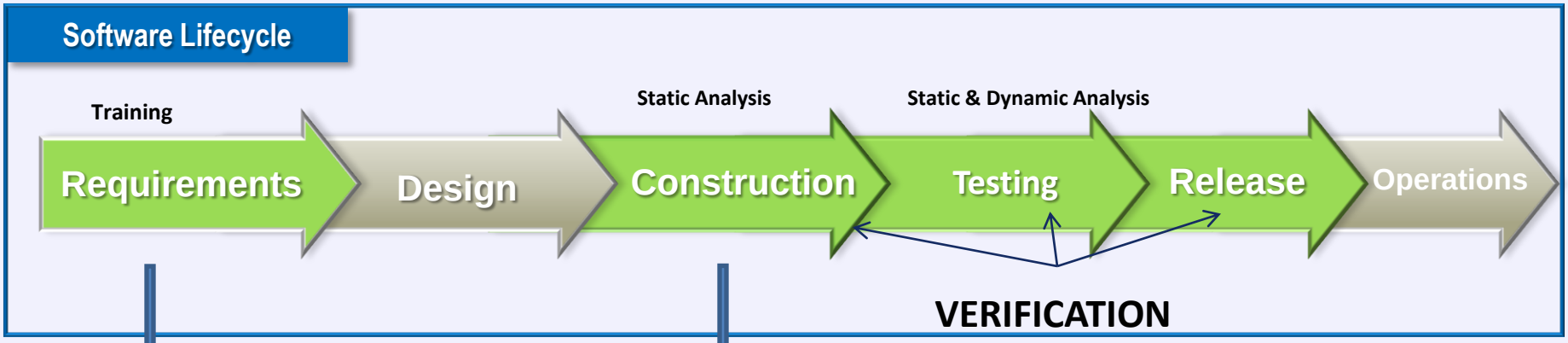


Remediation is Achievable

➔ Over **80% of applications** that were remediated to a satisfactory level did so in 1 week or less



Injecting Security into the SDLC



Training to help developers learn about the secure coding practices. Also provides CPE credit for certifications like CISSP.

Static Analysis
Submit scan through build server

Scanning engine analyzes the application code and returns results.

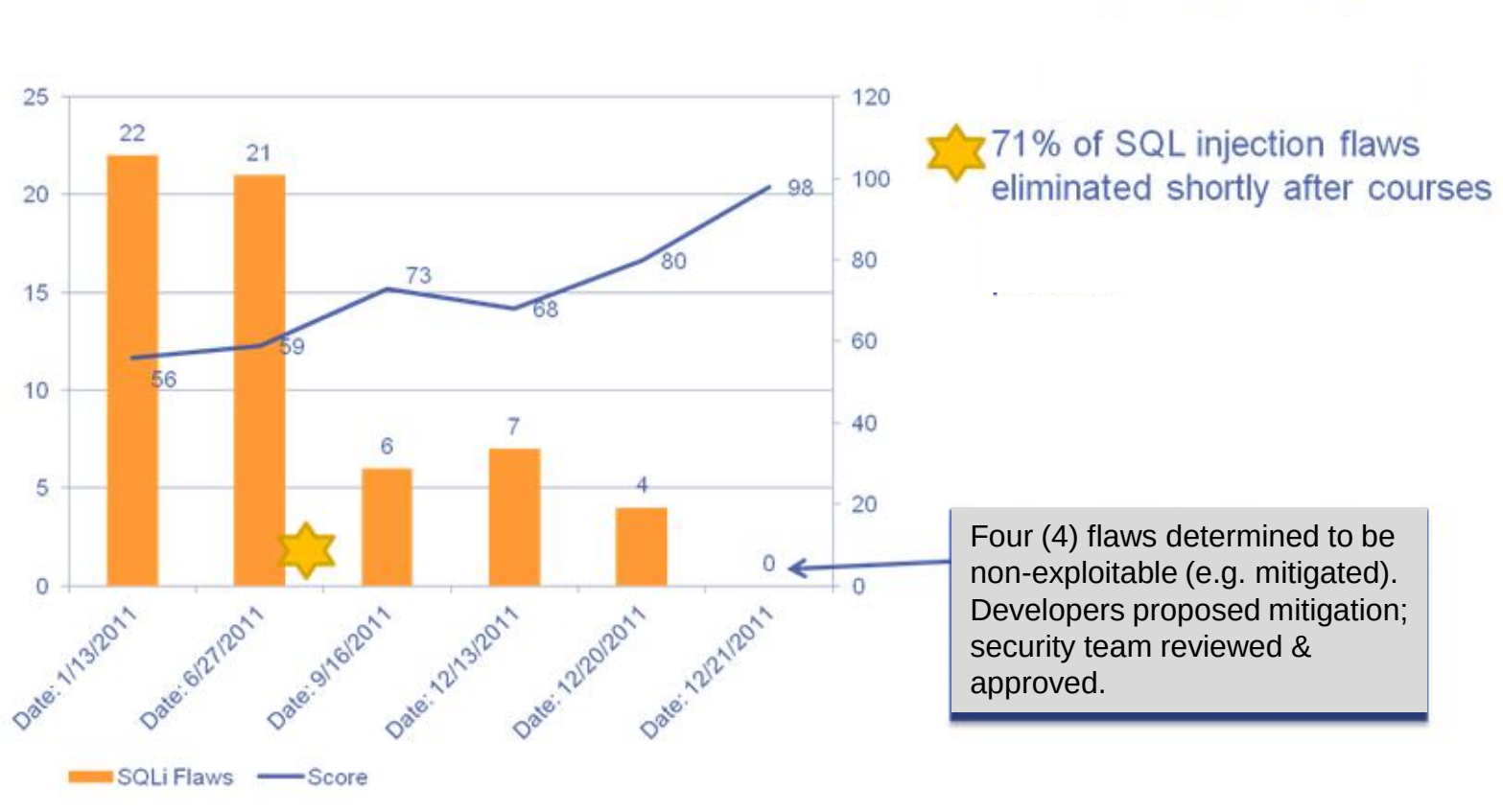
Static & Dynamic Analysis
Set up dynamic scanning (if web app) during the testing phase. Continue static scanning on reworked modules of the application.

Developer Training

- Instructor Led
  - Allows hands-on hacking
  - Allows for interaction
  - Costly
  - Limited bandwidth
- CBT / eLearning
  - More flexible / higher bandwidth
  - Customized curricula
  - Less costly per student
  - Retention of content may be less
- Both:
  - Secure code examples
  - Assessments & quizzes

Case Study:

Secure code training measurably reduces SQL injection



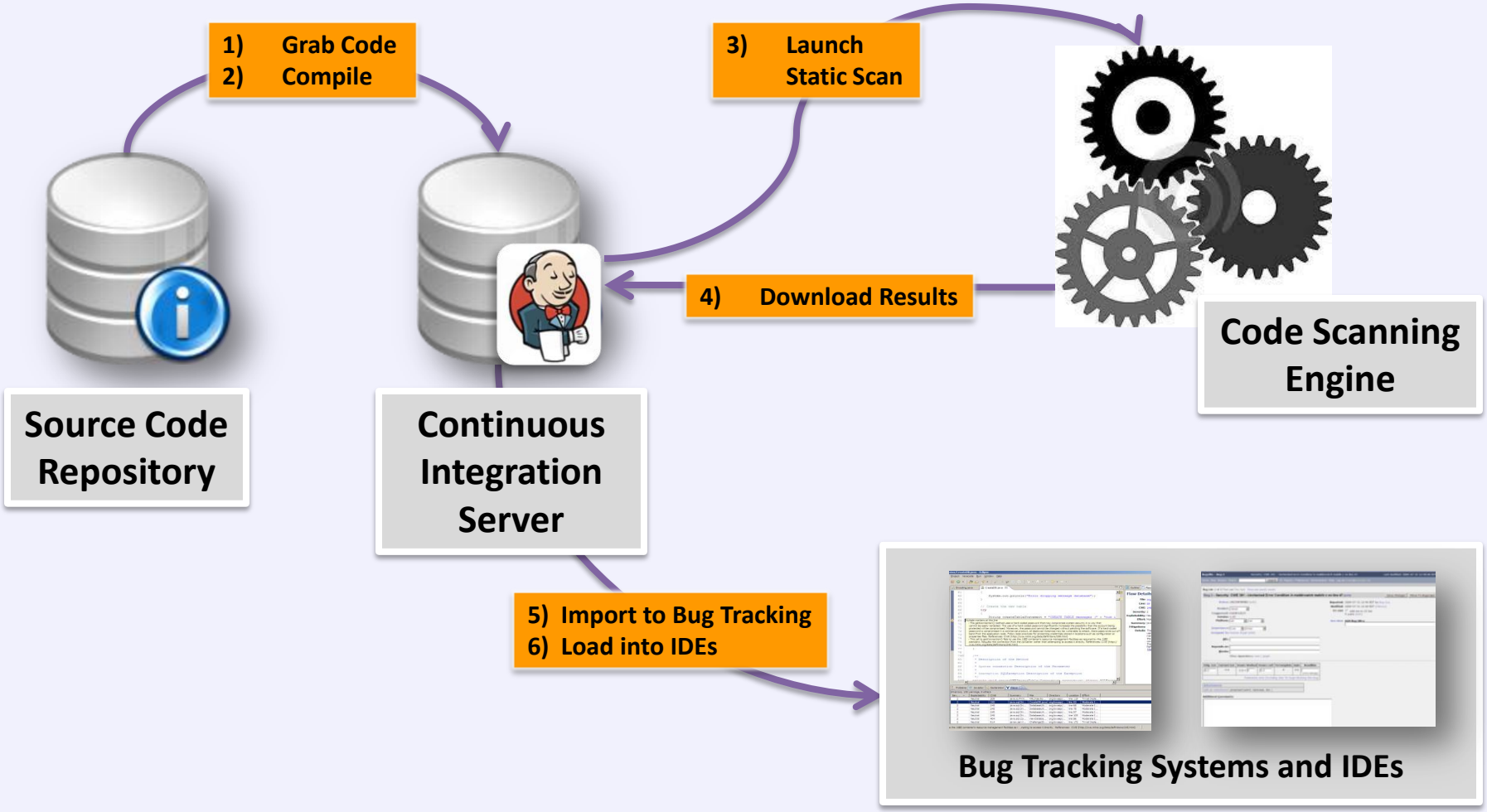
OWASP
The Open Web Application Security Project

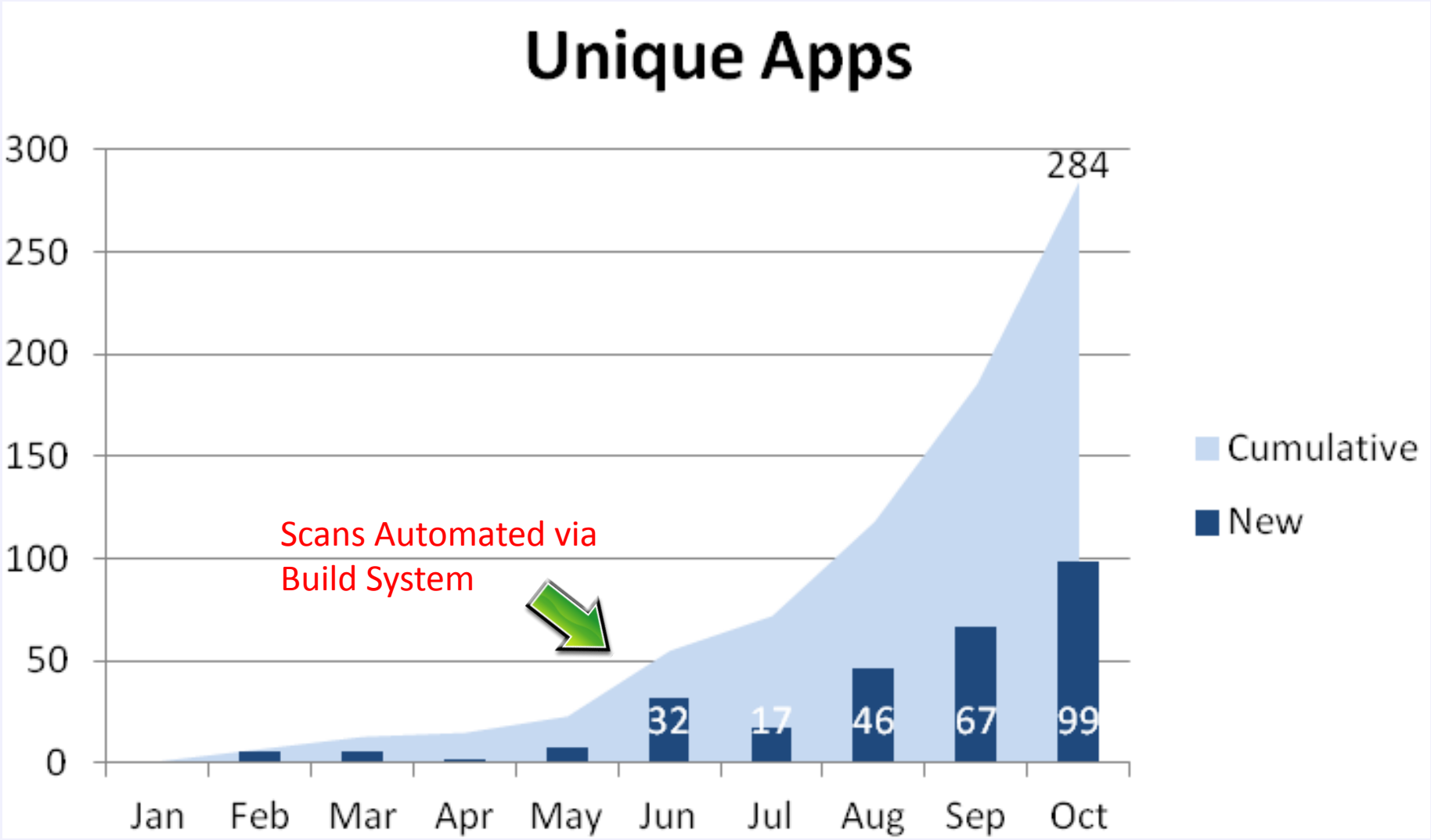
Continuous Integration / Build Systems

- Ant
- Maven
- Hudson
- Jenkins
- Microsoft TFS
- IBM Rational Build Forge
- Collabnet TeamForge
- CruiseControl
- QuickBuild
- AntHillPro

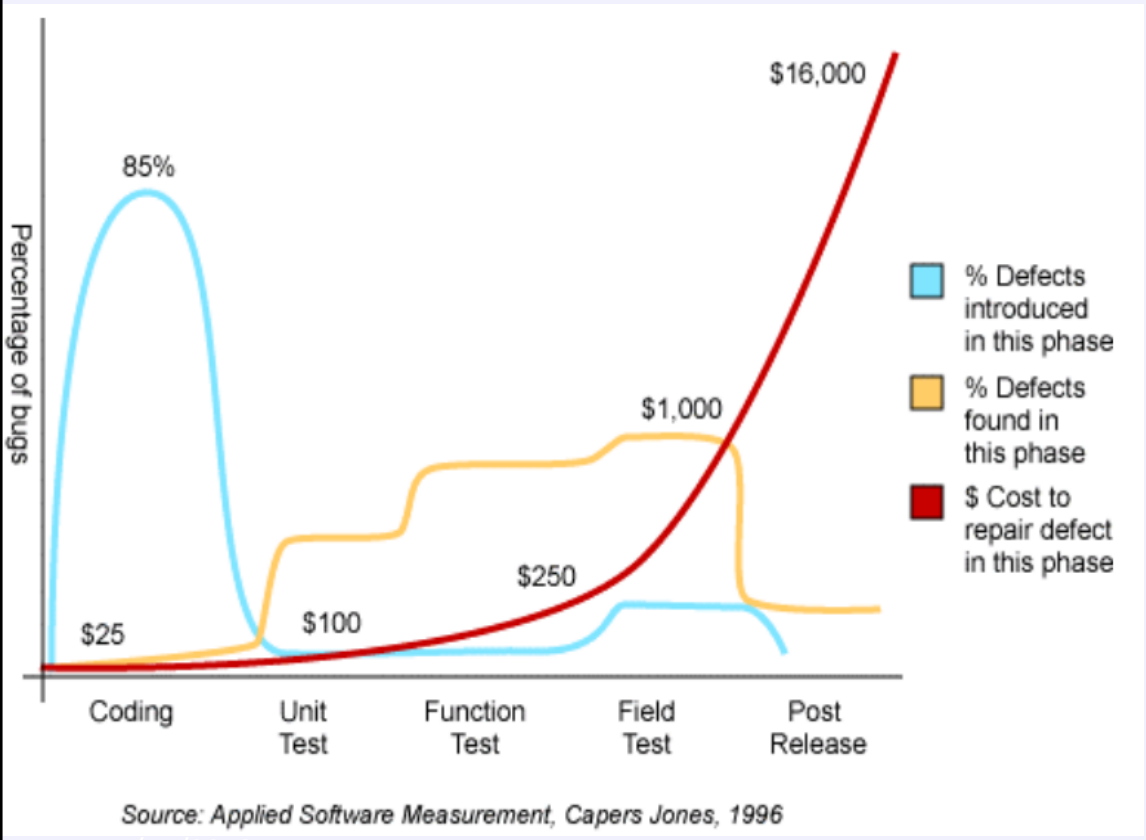
OWASP
The Open Web Application Security Project

Automating Static Analysis




Unique Apps
300
250
200
150
100
50
0
284
Scans Automated via
Build System
32
17
46
67
99
Jan
Feb
Mar
Apr
May
Jun
Jul
Aug
Sep
Oct
Cumulative
New

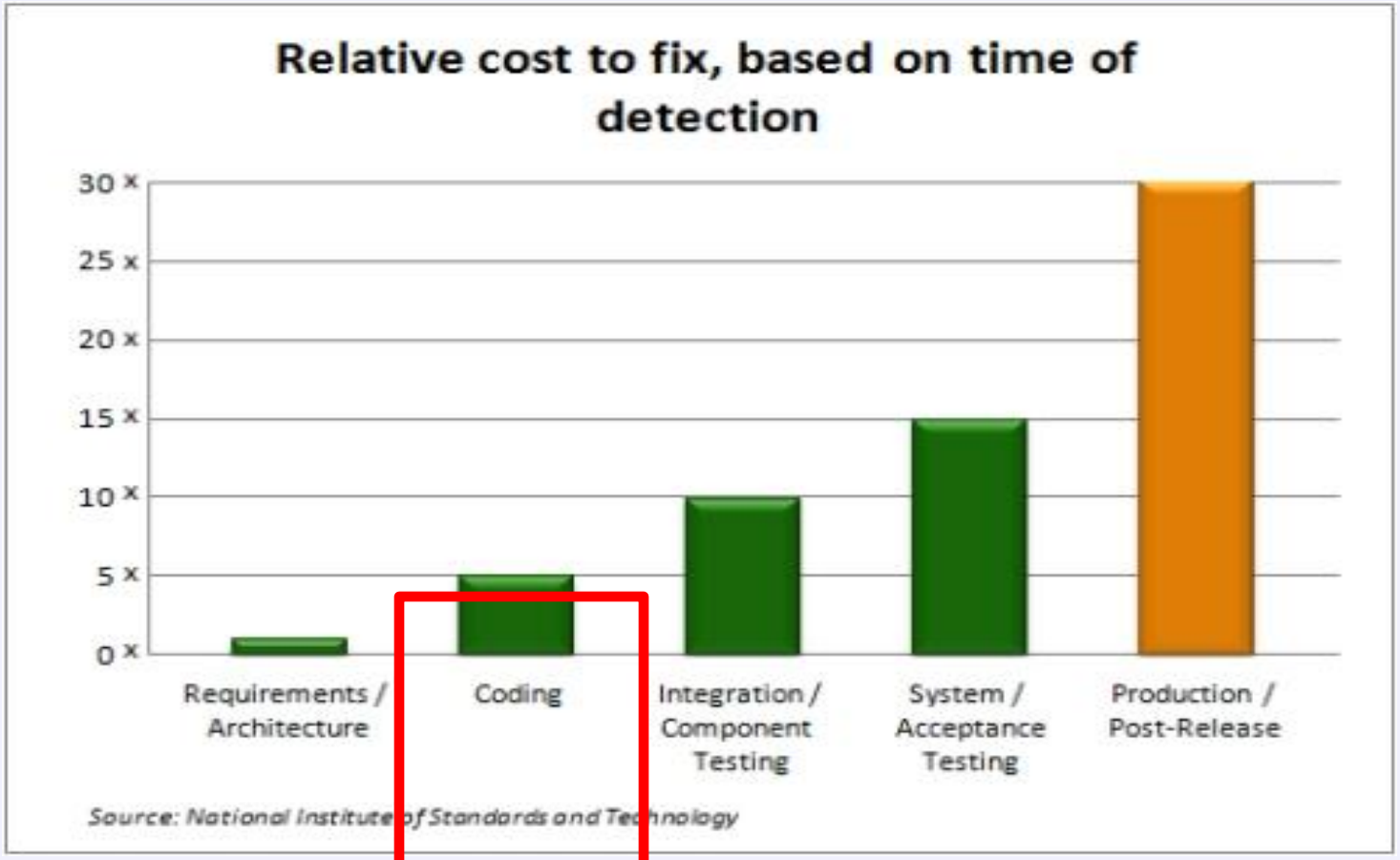
Advantages of Early Flaw Detection



Source: Applied Software Measurement, Capers Jones, 1996

"The National Institute of Standards and Technology (NIST) estimates that code fixes performed after release can result in 30 times the cost of fixes performed during the coding/development phase."

Source:
http://www.nist.gov/director/planning/upload/report02-3.pdf



Scan Early and Often!

OWASP
The Open Web Application Security Project

Summary: Building a Successful App Sec Program

- Critical to Success: Executive Support/Sponsorship
- Center of Excellence Approach
- Application inventory and classification
- Defined security policies
- Technology for automated testing
- Manual pen testing
- Integration of static code analysis with internal SDLC
- Assessment of 3rd party/purchased software
- Remediation guidance
- Developer Training
- Metrics

OWASP
The Open Web Application Security Project

Thank You

Contact info:
dferguson[at]veracode.com