FORTIFY

_security->starts here

Improving Security in the Application Development Life-cycle

Migchiel de Jong

Software Security Engineer

mdejong@fortifysoftware.com

March 9, 2006

General contact: Jurgen Teulings, 06-30072736

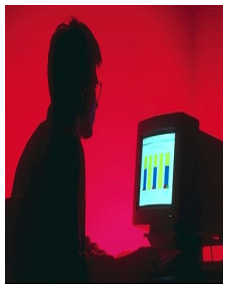
jteulings@fortifysoftware.com

Business Issues

- Staying off of the front page because of a hack
- Protect your customer assets and goodwill
- Risk with outsourced software
- Insecurity of third party and business partner software
- Protect against malicious insider activity
- Document and monitor a secure development lifecycle in support of regulatory compliance

The total cost impact on the financial services sector from an inadequate software-testing infrastructure is estimated to be $3.3 billion, according to the National Institute of Standards & Technology .

Security: Layered Solutions To Mitigate Risk



Hackers

Worms & Viruses

Malicious Insiders

Cisco Systems

Counterpane Internet Security

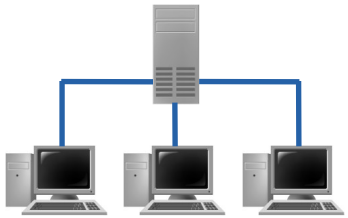
symantec.

Check Point Software Technologies Ltd.

Juniper Networks

McAfee

Traditional network/perimeter defenses



Critical Software Automation Of Key Operational Processes

FORTIFY _security->starts here

Software Apps Cut Through The Layers

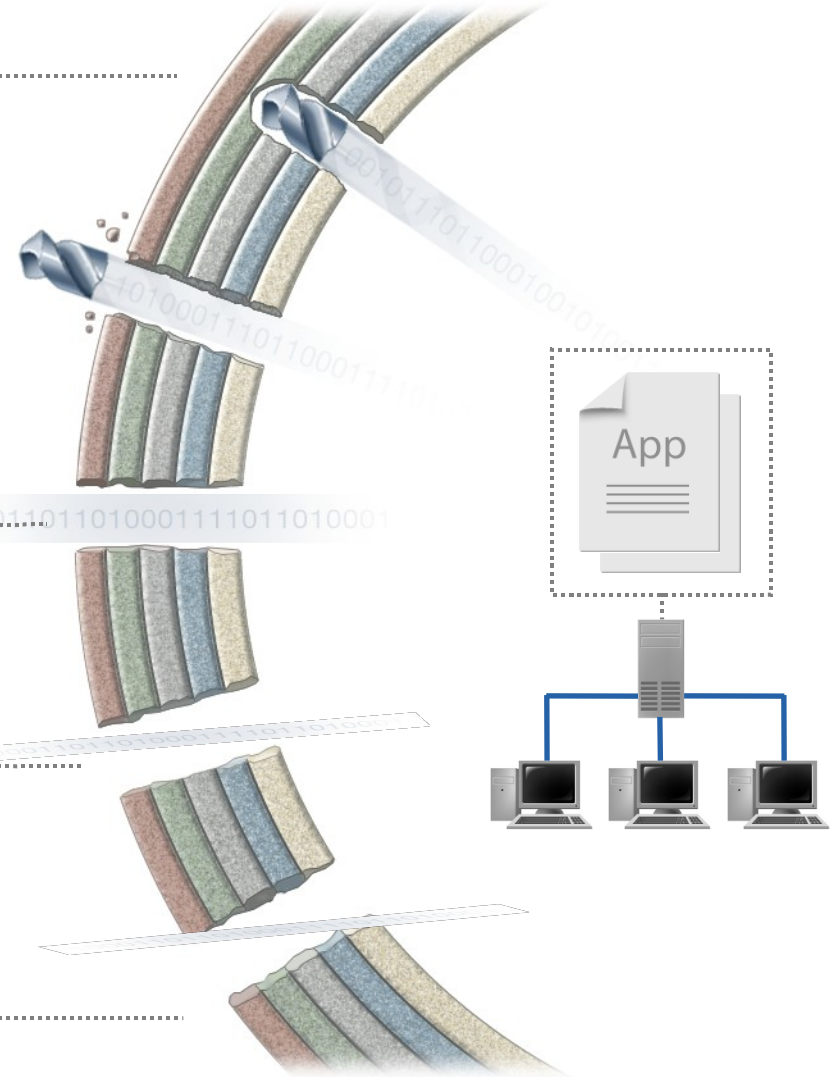
Outsourcing

Legacy App Integration

Web Facing Applications

Employee Self-Service

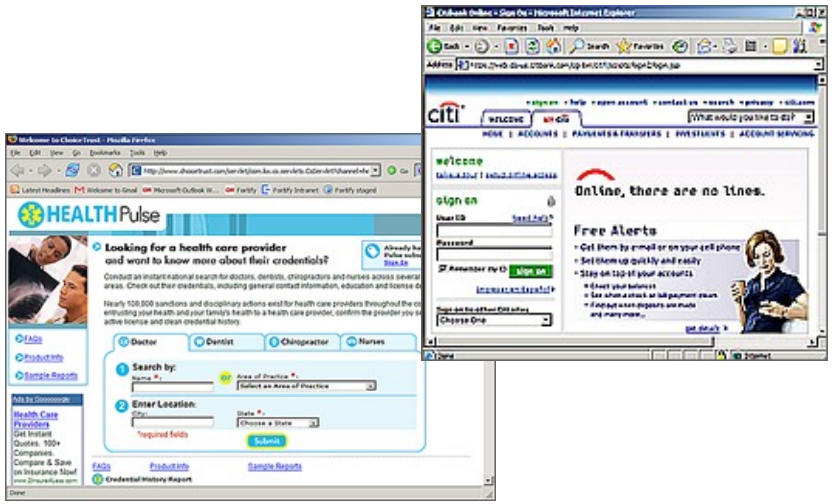
Connectivity w/ Partners & Suppliers

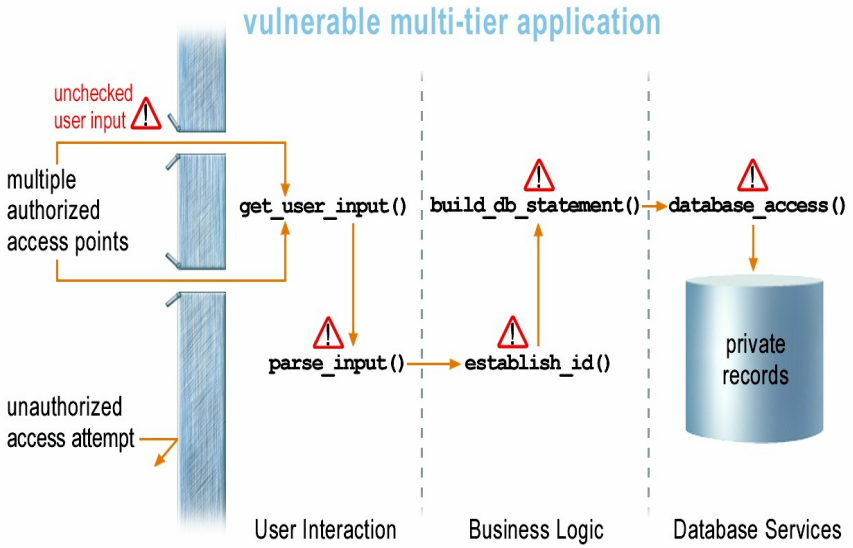


Today Hackers (and Insiders) Use The Software

There is an emerging and robust science of “hacking” software:

- SQL Injections
- Buffer Overflows
- Information Leakage
- Numerous Other Categories

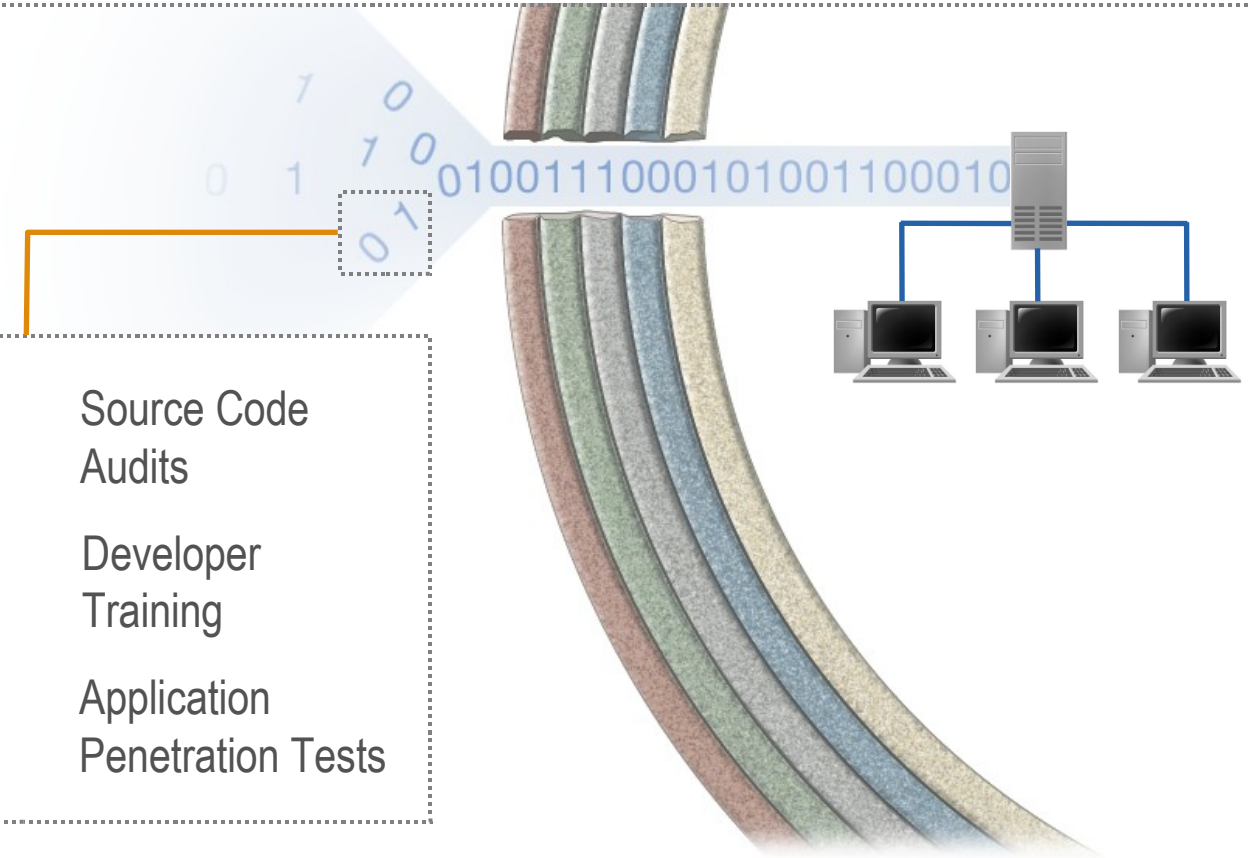




According to Gartner Group, “Over 70 percent of security vulnerabilities exist at the application layer, not the network layer.” It’s not just operating systems or web browsers, but all types of applications—particularly applications that automate key business processes.

Leading Organizations Starting To Address The Problem

JPMorgan Fidelity Investments Wells Fargo eBay PayPal Microsoft
Oracle Time Inc. Bank of America Higher Standards Goldman Sachs



- Security Design Reviews
- Network-based Solutions
- Source Code Audits
- Developer Training
- Application Penetration Tests

FORTIFY
_security->starts here

Steps to take to deliver more secure Code

- Evaluate and Plan
- Specify the Risk and Threats to the Software
- Review the Code
- Test and Verify the Code
- Build a Gate
- Measure
- Educate

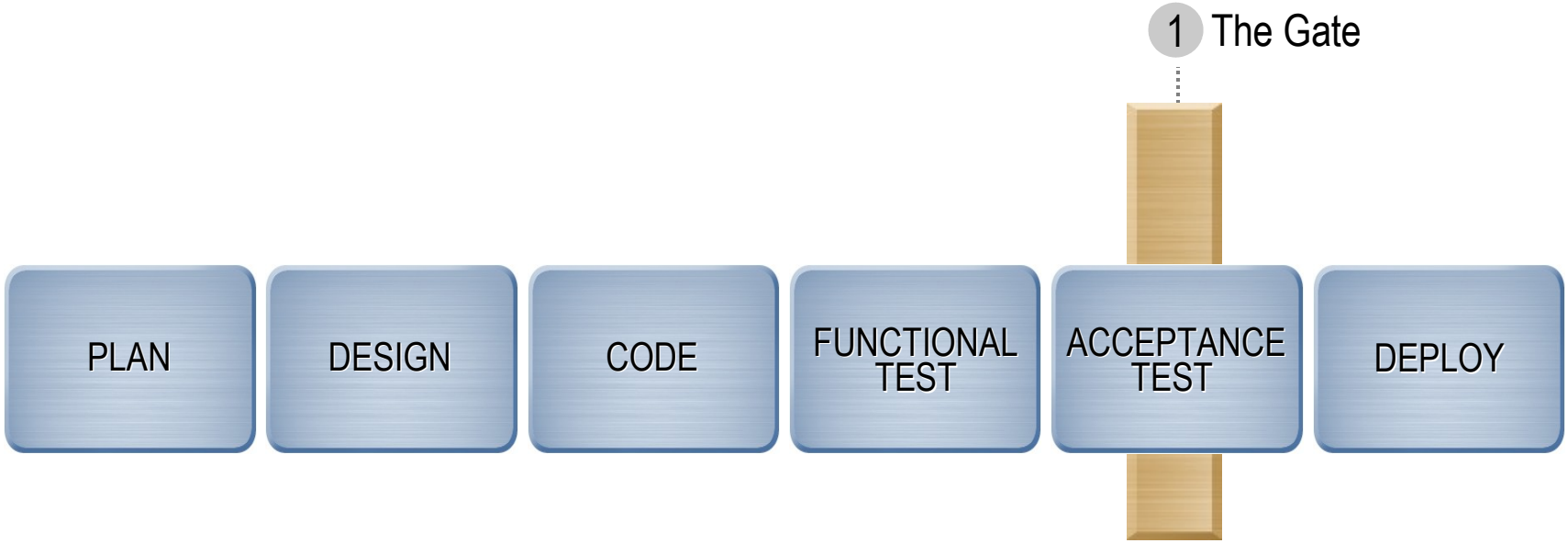
A simple Check List

- Internal Security experts exist and have been identified — 1 2 3 4 5
- Threat analysis completed for every project — 1 2 3 4 5
- Education programs for security occur regularly — 1 2 3 4 5
- Specific tools and resources have been acquired and allocated — 1 2 3 4 5
- Post-deployment security if fully integrated in the overall process — 1 2 3 4 5
- Vulnerability response maximizes benefits and prevents repeats — 1 2 3 4 5

First Step

Software Security as a Gate

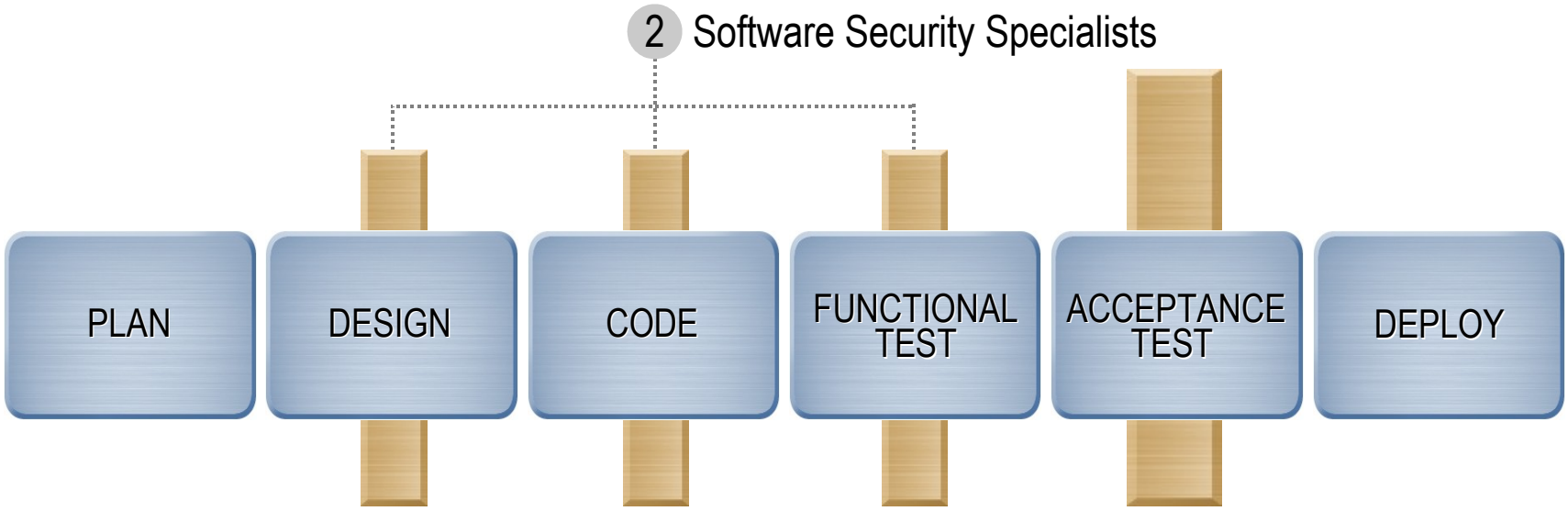
- Used as final check pre-deployment
- Acceptance of outsourced software
- A first step in a comprehensive program



Second Step

Software Security With Specialists At Discrete Process Points

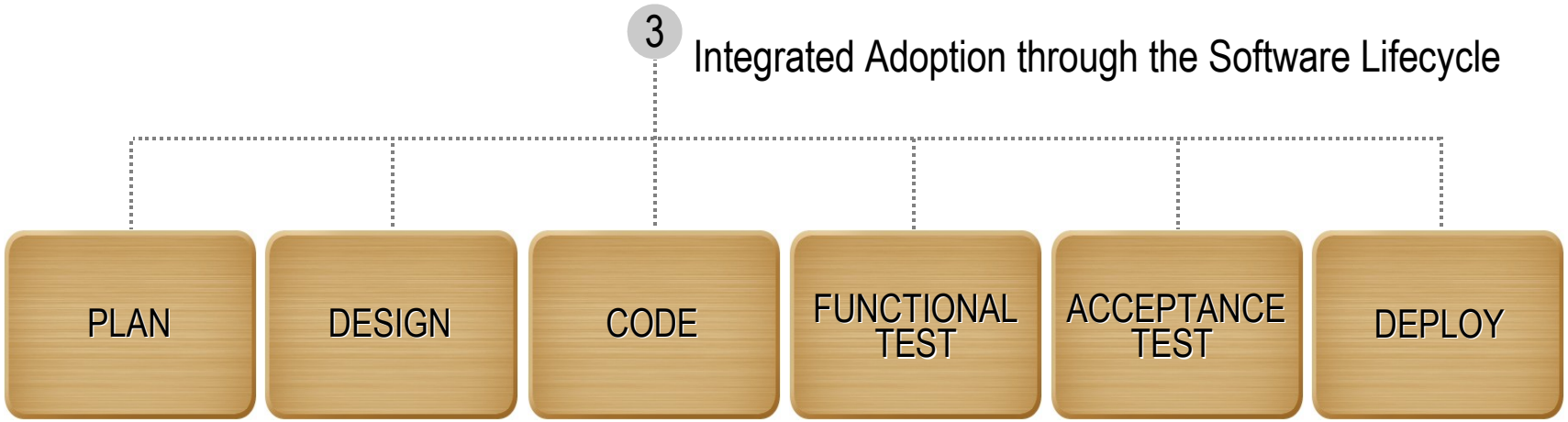
- Automation make specialists more productive
- Central repository makes comparisons possible
- Build awareness of security within development process



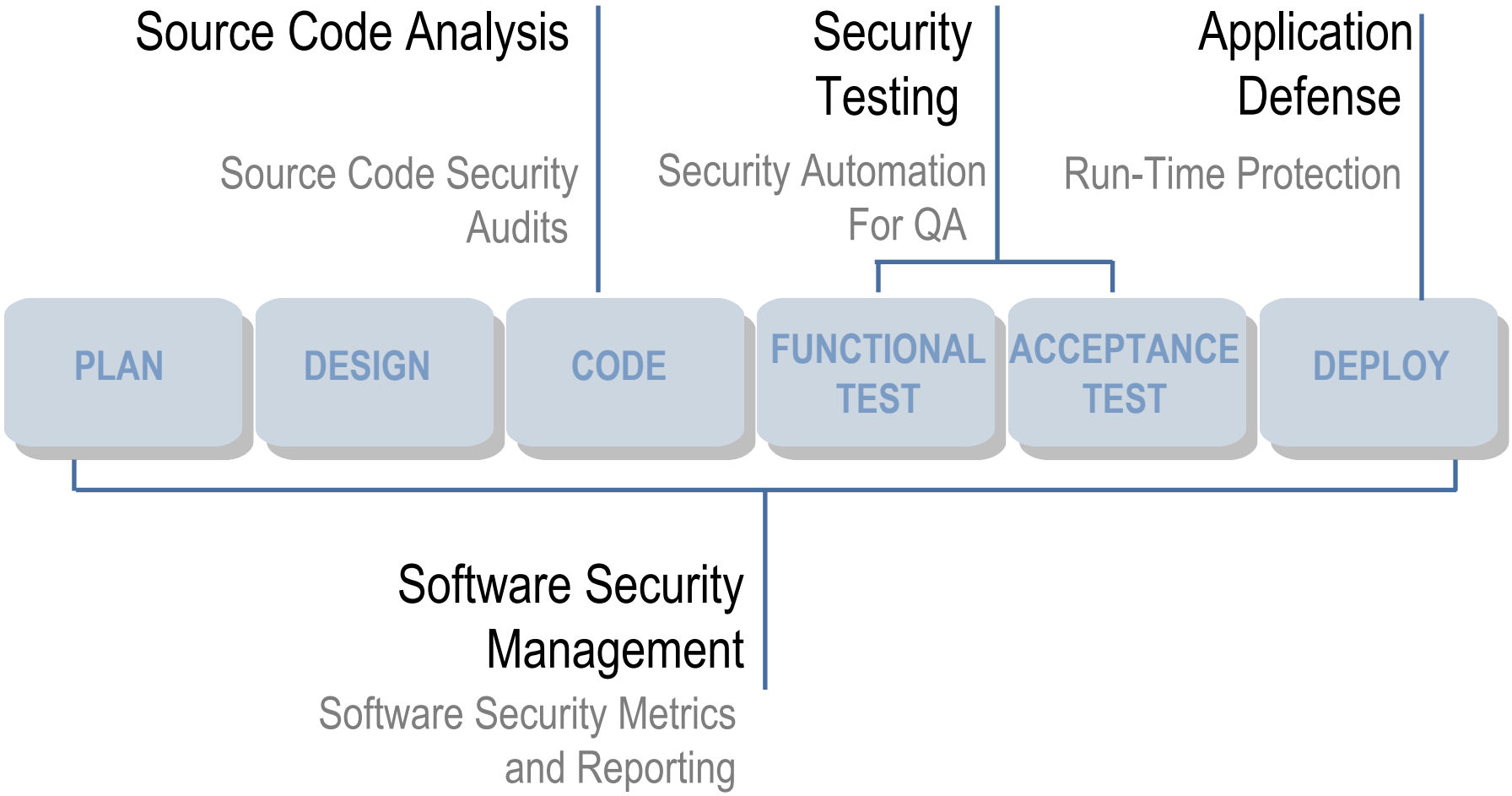
Ultimate Goal

Software Security Throughout Development Process

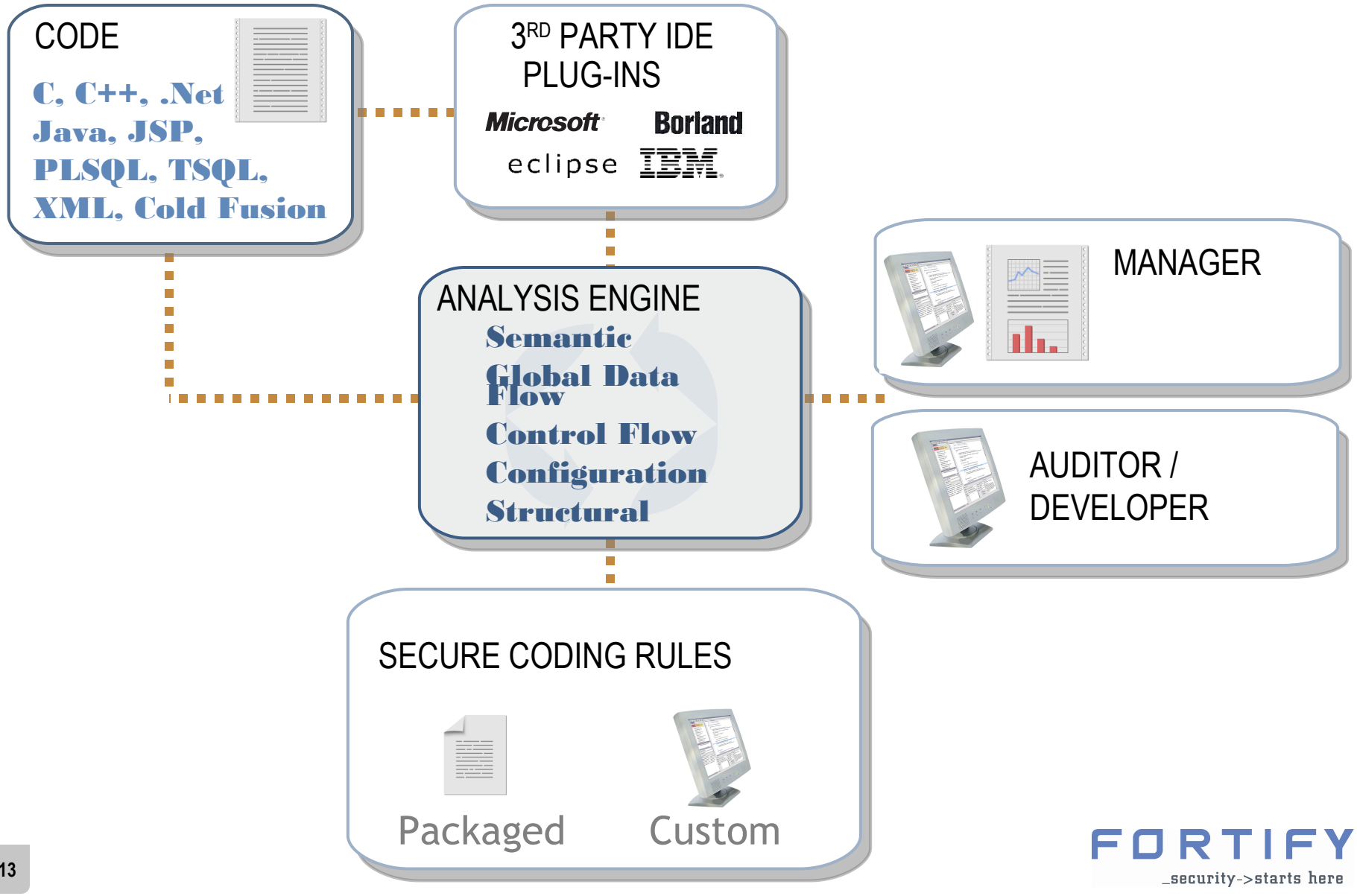
- Eliminate vulnerabilities when they are least costly
- Continual artifacts for compliance and secure quality assurance
- Built in protection



Automation support

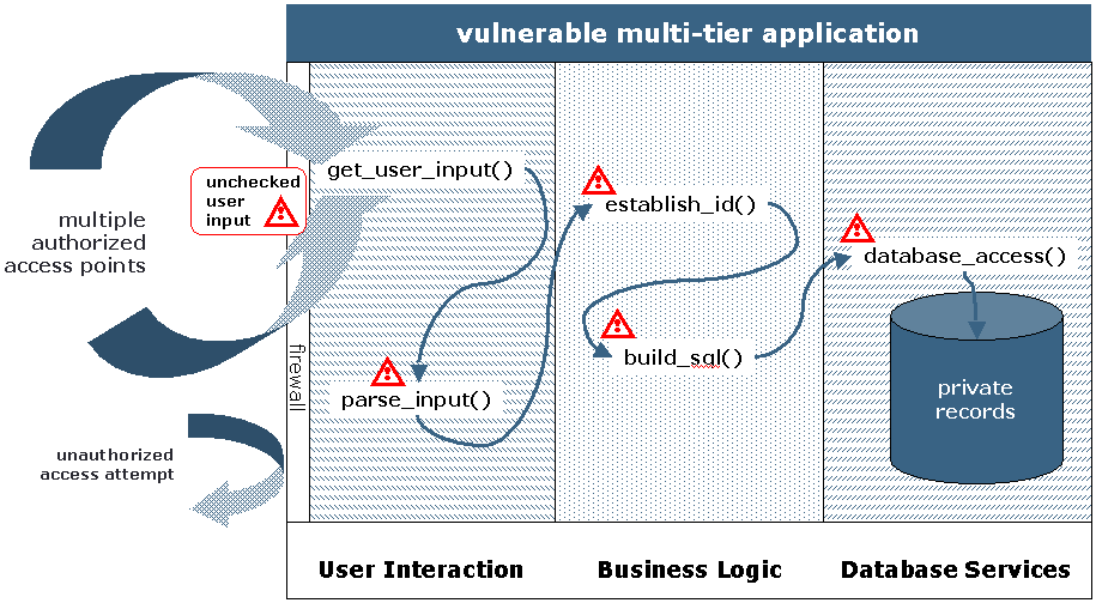


Source Code Analysis



Data Flow Analysis

- Ability to follow data flow across tiers
  - Doing this kind a analysis is really time consuming (at least without automation)
- Ability to follow data flow across languages
  - Reduces audit, development and testing skill set



Source code analysis demonstration

- Static analysis of OWASP's WEBGOAT version 3.7
- Metrics

FORTIFY

_security->starts here

Thank You

General contact: Jurgen Teulings, 06-30072736

jteulings@fortifysoftware.com