
FORRESTER®

FORRESTER®

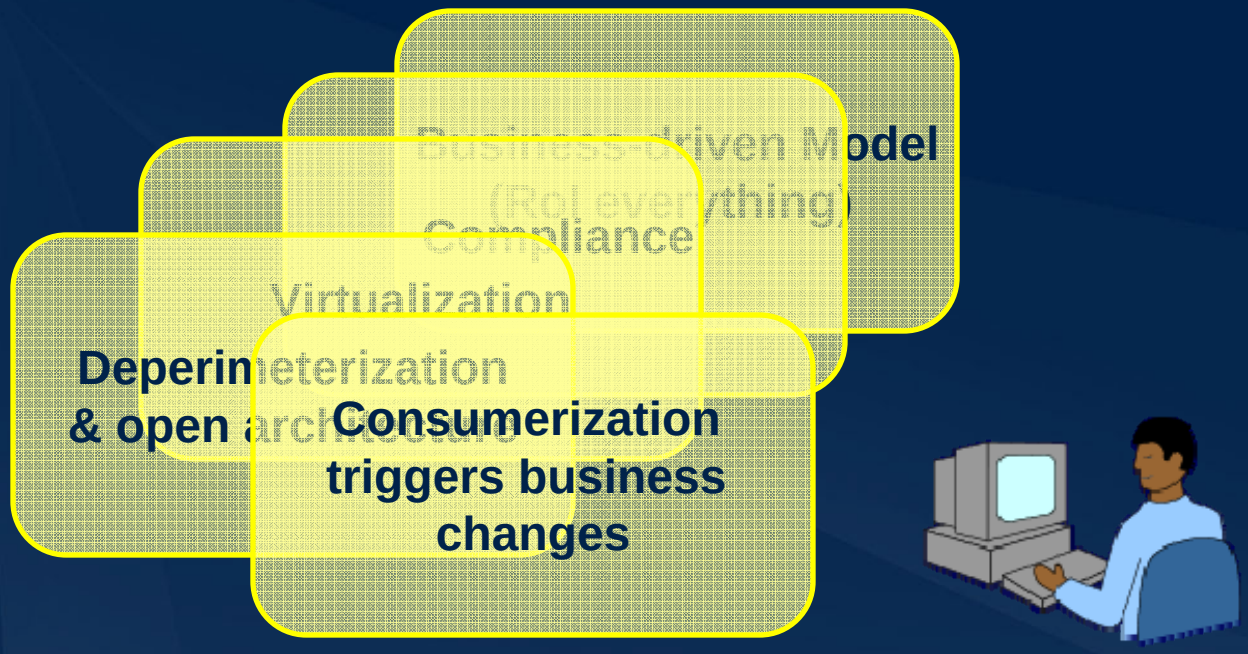
Web 2.0, Consumerization, and Application Security

Chenxi Wang, Ph.D.
Principal Analyst
Forrester Research

OWASP, New York City

September 25, 2008

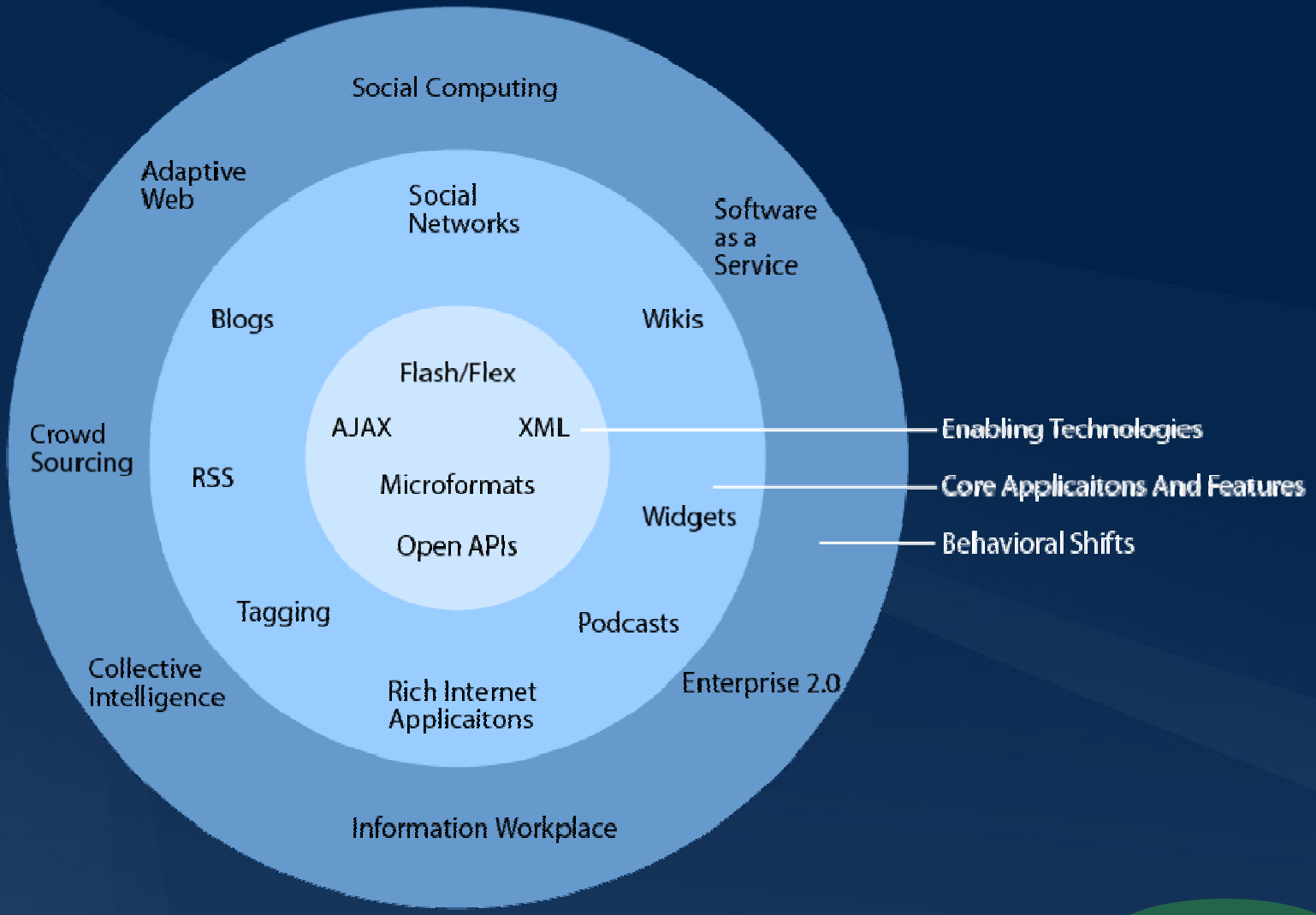
Today's enterprises face multitude of challenges



Consumerization

- Is at the forefront of innovation
- Fosters new business models
- Promotes new social structures
- have become part of enterprise fabric

Three Lenses to View Consumerization



Consumer technologies foster new business

Example: Random House book widget

- Provides top-selling books in your ZIP code
- New viral distribution channel
- Creates new business opportunities

Browse

Order

Click to read

Title & author

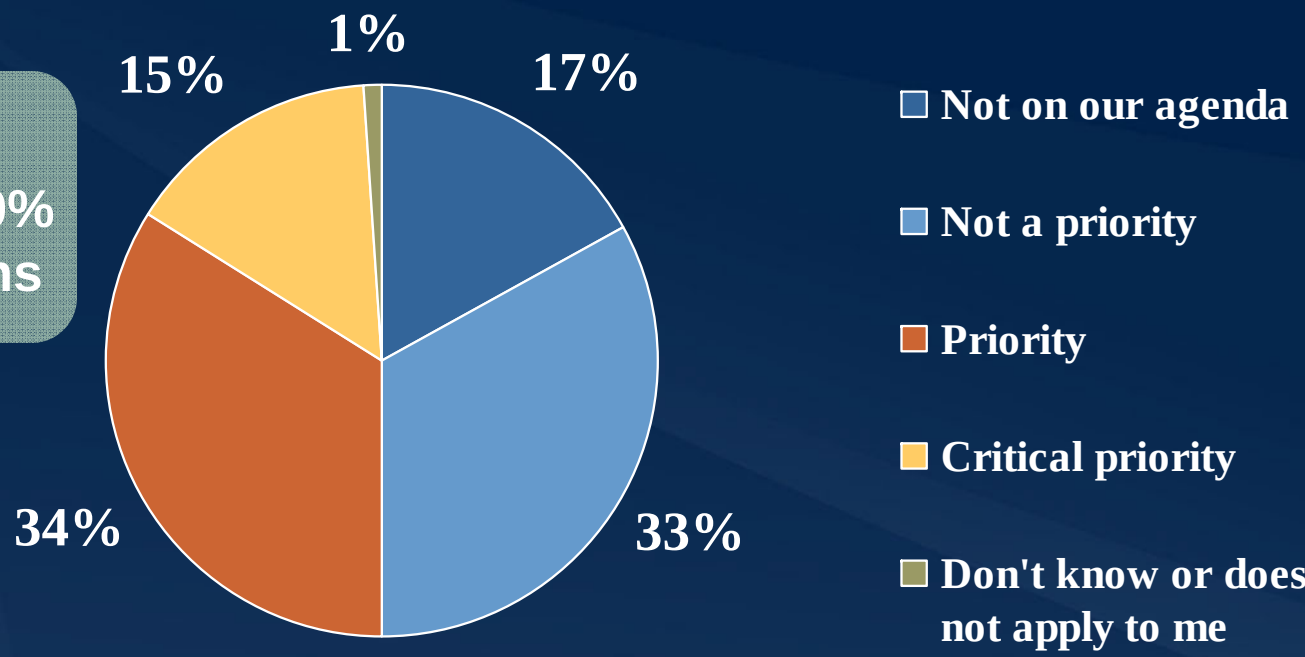
FORRESTER®

Collaboration is part of enterprise strategy

"Which of the following are likely to be one of your IT organization's major software technology initiatives for the next 12 months?"

Implement an enterprise collaboration strategy

Collaboration Is priority for 50% of organizations



Near 50% of businesses view it as a priority

Base: 2,252 Software IT decision-makers at North American and European companies
Source: Forrester Enterprise And SMB Software Survey, Q3, 2007

FORRESTER®

Many are building serious applications on top of consumer technologies

- Crew portal (mission-critical app)
- Compliance documentation management

AMD
www.amd.com

- AMD Central intranet
- Global partner sites (mission-critical)

EXOSTAR®

- “Trusted workplace” for multi-company collaboration
- Sharing of critical design documents (airplane design docs, CAD drawings)

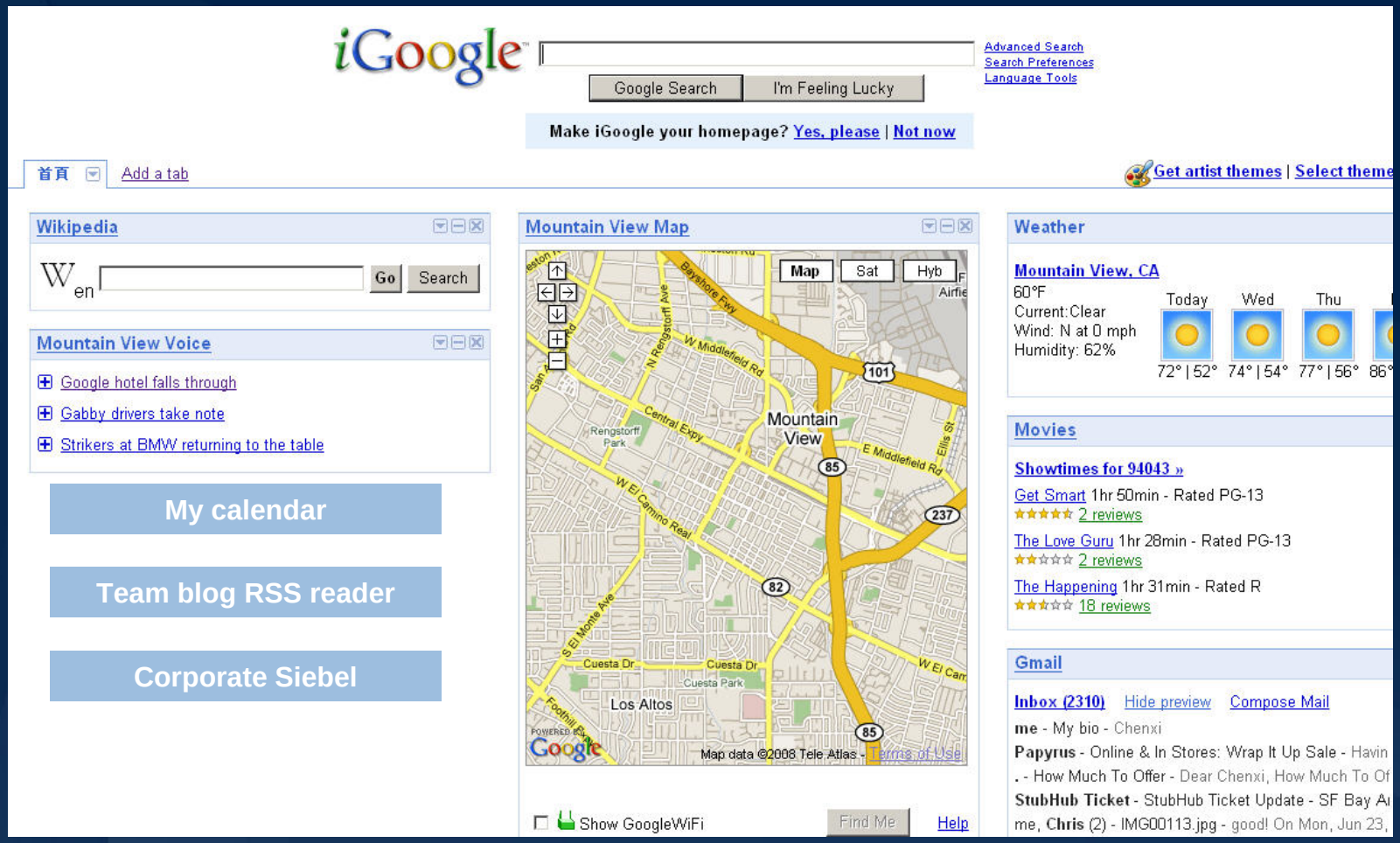
FORRESTER®

This is how a distributed team could work . . .

This is how a distributed team could work . . .

FORRESTER®

Perhaps this is what happens . . .



FORRESTER®

Consumerization accentuates every aspect of deperimeterization

FORRESTER®

Consumerization accentuates every aspect of deperimeterization

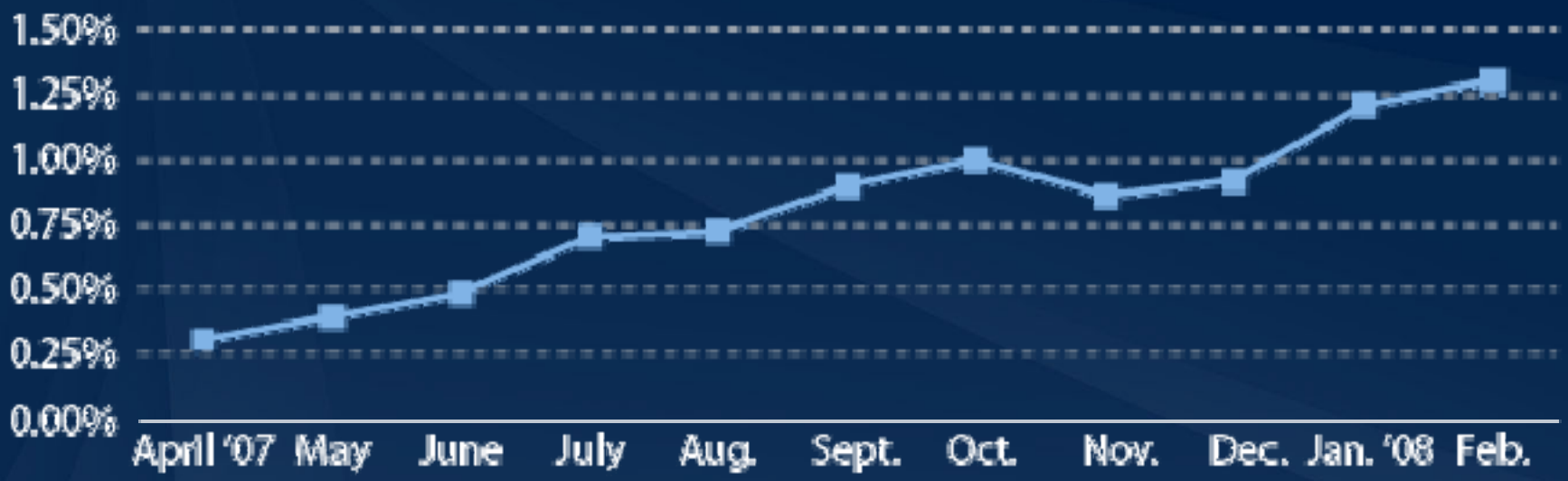
FORRESTER®

But, organizations are wary of consumer technologies

- Greater number of security risks
  - » Attackers are targeting collaboration and Web 2.0 applications.
  - » Staying ahead of all the issues is difficult.
- Increased complexity of IT operations and management
- Further dilution of the enterprise boundary

Web 2.0 Security risks

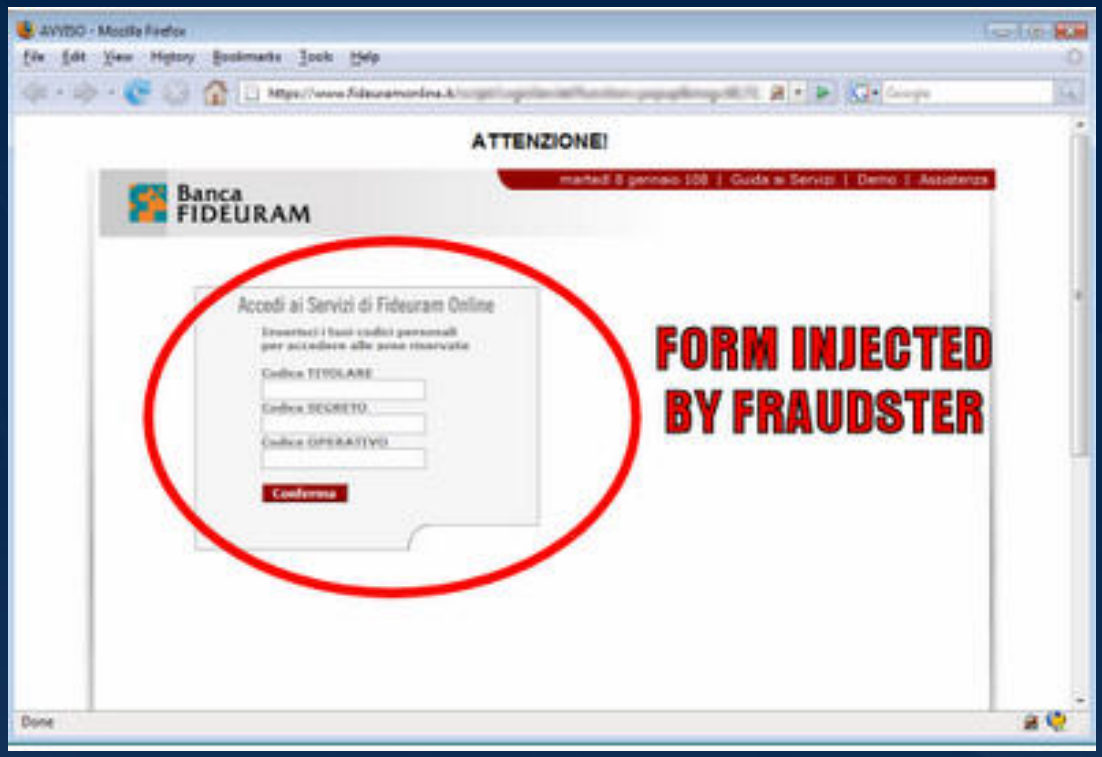
Internet is becoming more dangerous . . .



Source: "All Your IFRAMEs Point to Us," Google, February 4, 2008

FORRESTER®

Innocent participants in malware infection chain



- Fraudsters exploited a XSS vulnerability to inject a modified login form onto Banca Fideuram's Web page
- User's account info is sent to a server in Taiwan

FORRESTER®

Data theft is rampant …

Low	High	
$2	$5	Payment card
$25	$35	Magnetic stripe data
$200	$300	Full account information
$25,000	$40,000	Zero-day malware

eBay, PayPal accounts	Other items for sale
Driver's licenses	
Birth certificates	

Source: The Aegenis group, USENIX 2008

FORRESTER®

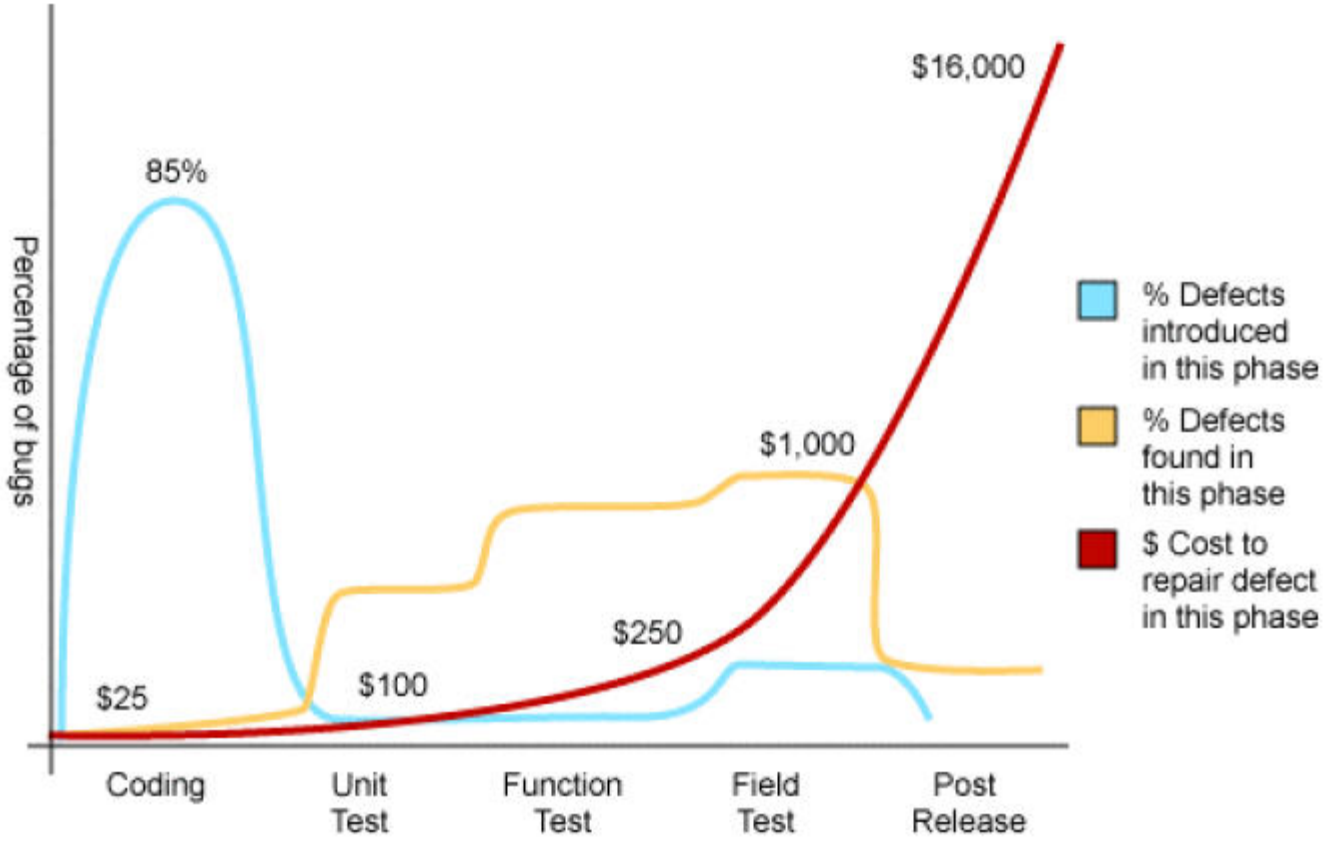
Web 2.0 highlights application security

- Increased attack surface
  - » Client side state matters
  - » Code injection is a top security risk
- Developing secure web 2.0 apps is challenging
  - » Security testing is more complex
  - » Authentication/Authorization logic is more complex
- Traditional trust model is broken


Security has to be baked in
rather than painted on ...

Why Security during development?

85% of code flaws are introduced during coding



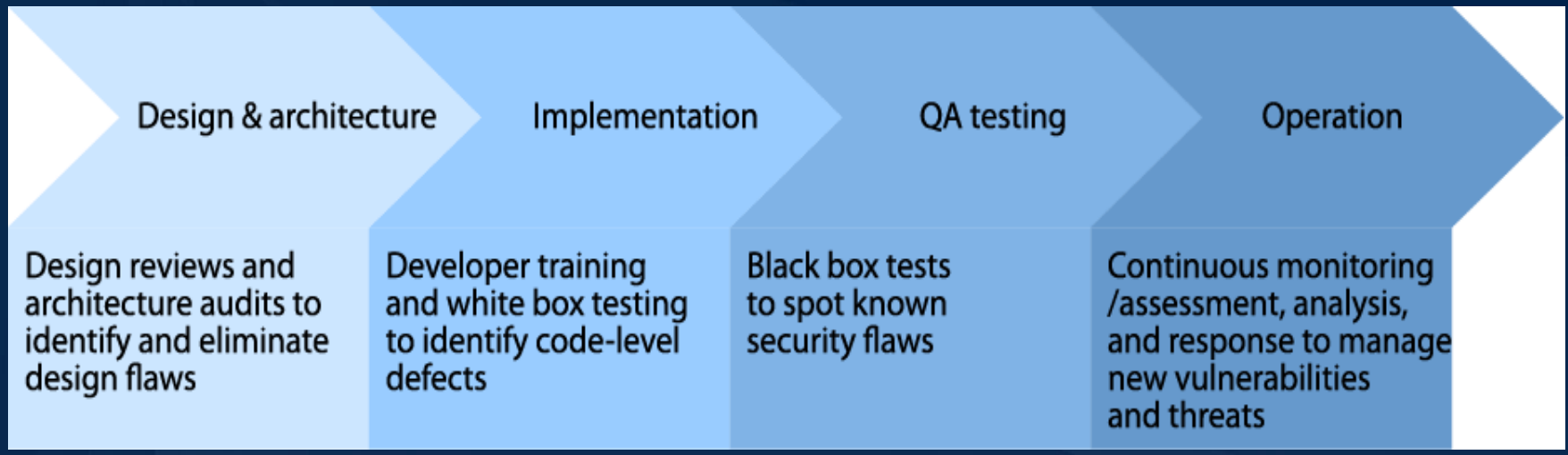
Source: Applied Software Measurement, Capers Jones, 1996

FORRESTER®

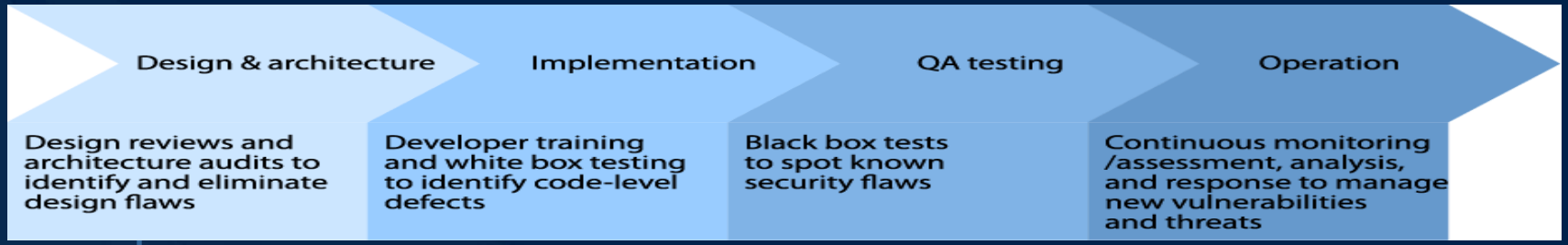
Why security during development - recap

- The #1 reason: to catch vulnerabilities before attackers do
- Reduce cost of fixing bugs and vulnerabilities
  - Example: $30,000 post release, $3,000 integrated testing, $500 development
- Help to achieve regulatory compliance
  - PCI, HIPAA, Sarbanes-Oxley, …

An application lifecycle view of security

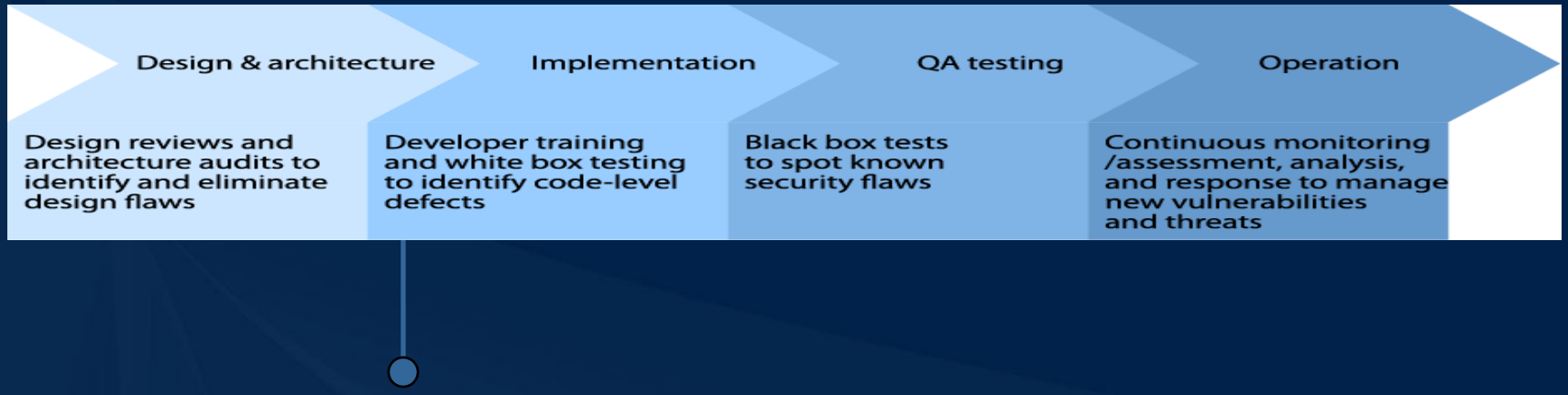


FORRESTER®



- Begin with requirements
  - » What business asset does the software represent?
  - » What assets of value are accessible from the software?
  - » What protections must be provided for those assets?
- Design and architecture review
  - » What kind of operational environment
  - » What are the threats to my assets in this environment?
  - » What threat-mitigation measures must be provided?

FORRESTER®

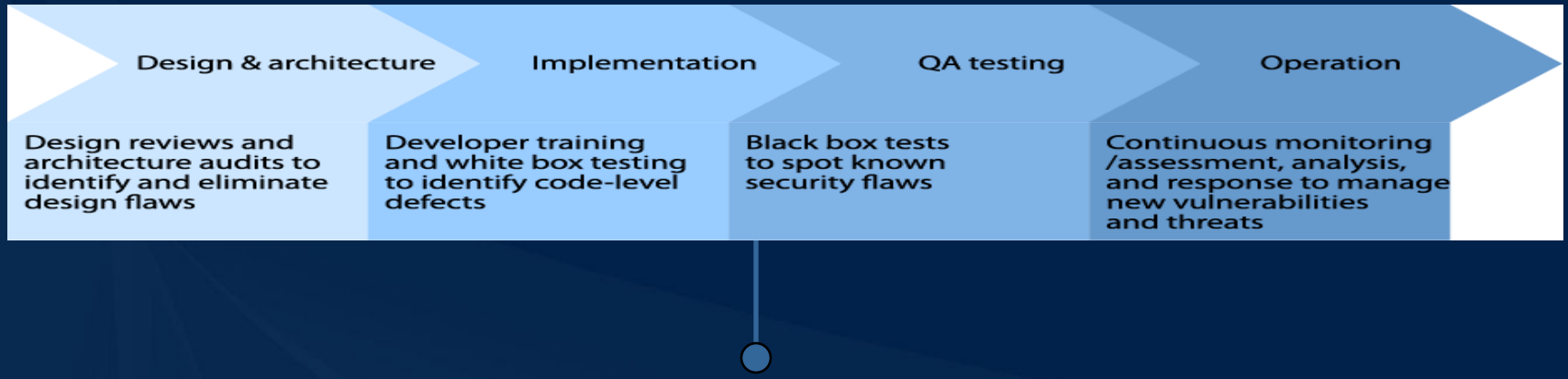


- Develop a company secure coding guidelines and standard
  - » Libraries of approved security functions
  - » Vulnerability remediation process
- Develop language-specific checklists
- Utilize code-level analysis tools
  - » Static analysis tools
  - » Security testing tools
- Code reviews and walkthroughs

FORRESTER®

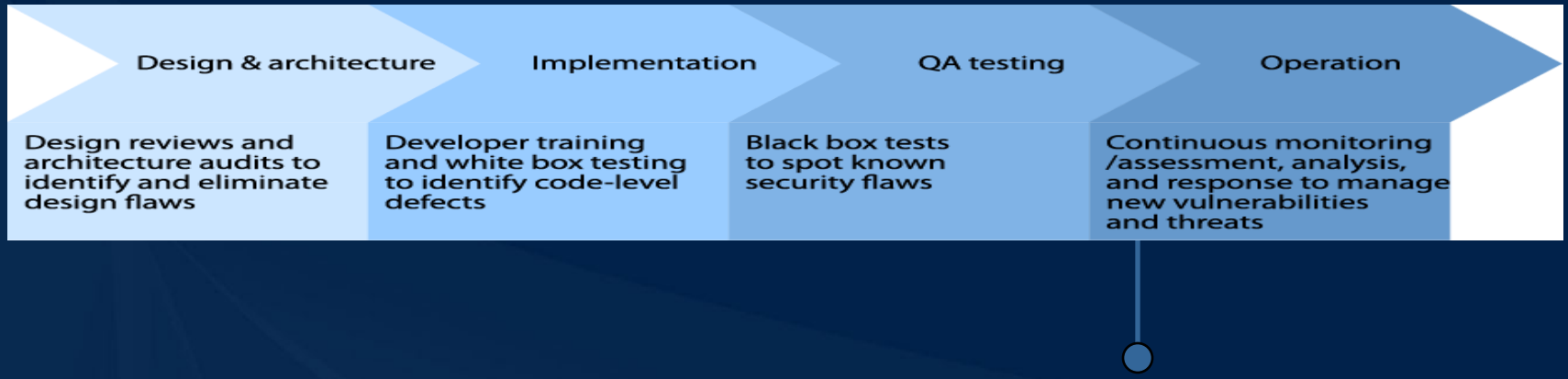
Specific code-level guidelines

- Validate input from all untrusted data sources, including web inputs, environmental variables, network interfaces, and user controlled content
- Do not use any unsafe functions
- Sanitize all data sent downstream
- Use tried-and-true crypto packages: e.g, Bouncy castle,
- Verify security policies are enforced
- Do not hard-code default and test accounts
- Default deny: base access decisions on permission rather than exclusion



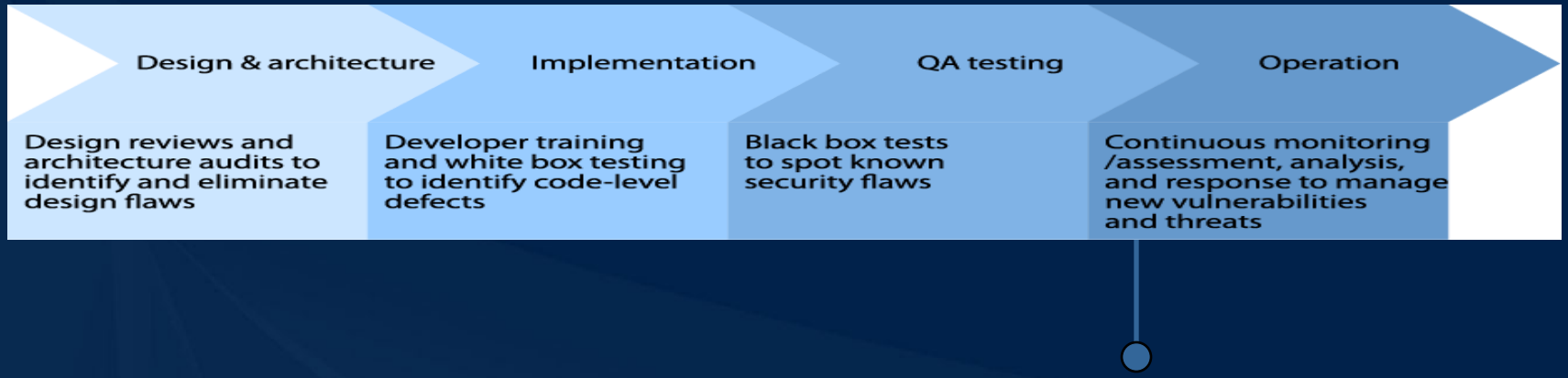
- Fault injection
- Fuzzing
- Black-box and automated penetration testing
- Augment with manual testing
- Whittaker and Thompson, *How To Break Software Security*, Addison-Wesley, 2004

FORRESTER®



- Secure configuration
  - » Disable all default accounts at the end of installation
  - » Force the user to set strong administrative passwords
  - » Configure appropriate logging and auditing procedures
- Secure operation
  - » Consider web application firewall
  - » Periodically conduct penetration and black box testing

FORRESTER®



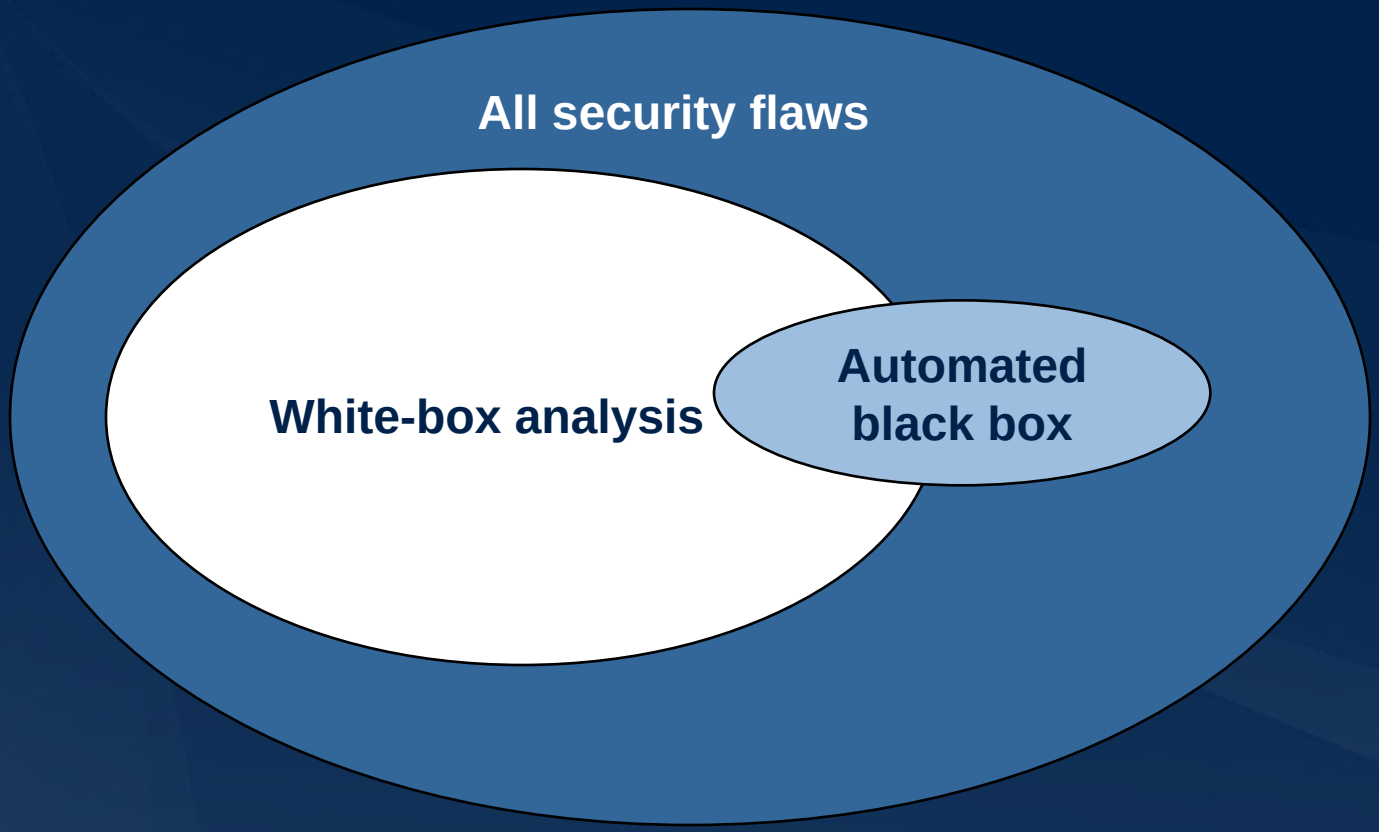
- Maintenance
  - » Enforce all of your secure software development processes for maintenance releases of code
  - » Integrate with change management procedures
  - » Make sure that your maintenance engineers fully understand the intended use and architecture of the product and adhere to operational principles
  - » Have an incident response plan

FORRESTER®

How are companies adopt SDLC?

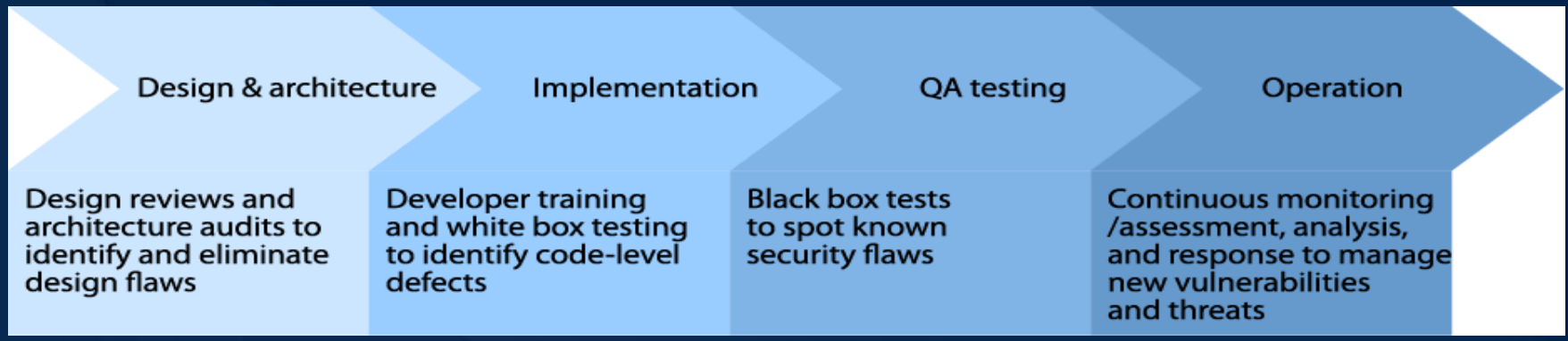
- >65% companies have penetration testing and black box scanning
- Source code security technology is by far the smallest
  - » Developers are under time-to-market pressure
  - » Security tools render false positives are not acceptable

Comparing white-box and black-box



FORRESTER®

1 + 1 > 2?



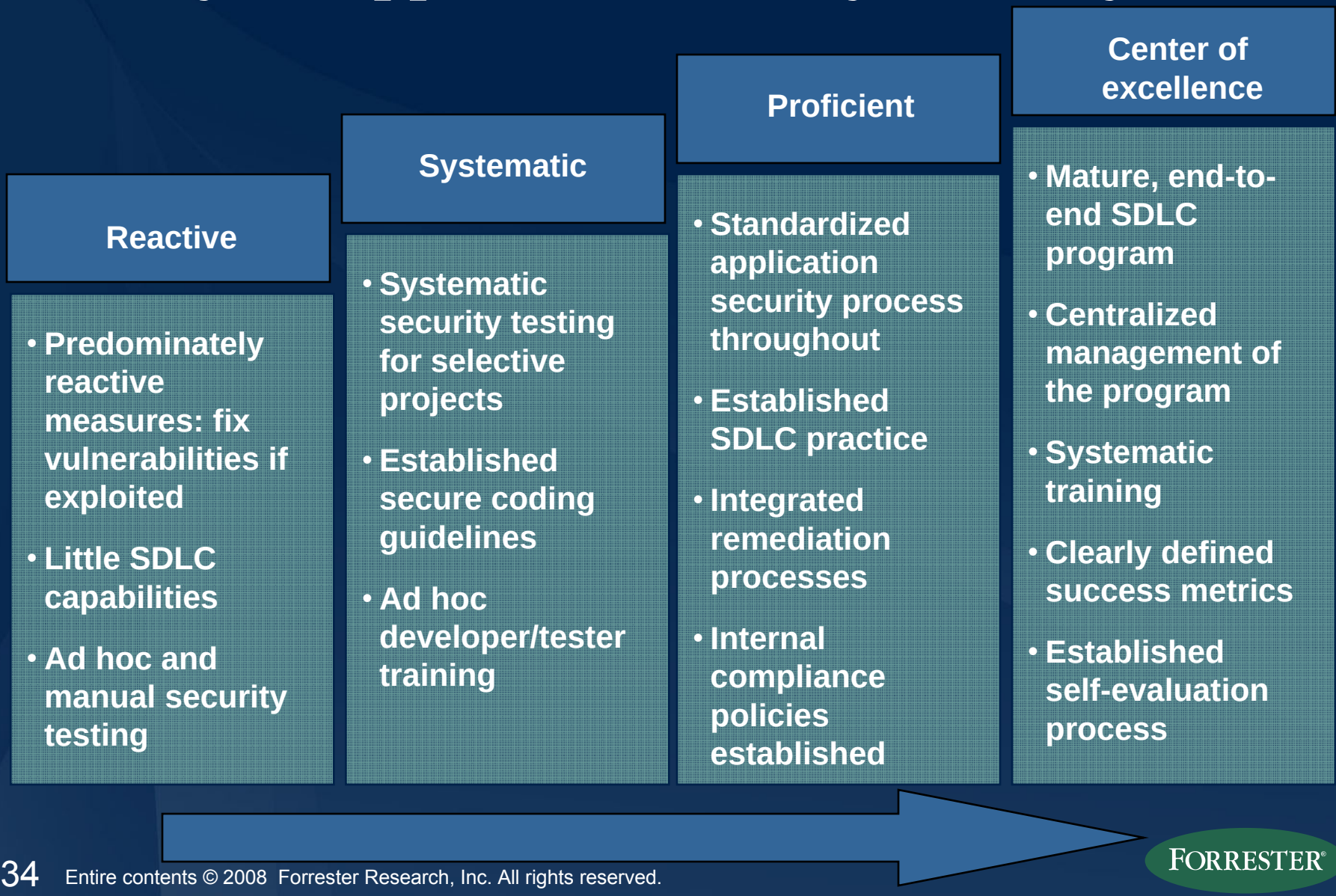
- **Integrated whitebox and blackbox, e.g.,**
  - **Using black box tests to verify white-box findings**
  - **Use white-box analysis to guide black box tests**
- **Users should demand integrated vendor functionality**
  - **Only combine reports is not enough**

FORRESTER®

Developer training is essential

- Defense in depth
- The principle of least privilege
- The principle of segregation
- Fail closed or fail secure
- Enforce known good traffic (white-list approach)
- Verify trust model
- Keep it simple. Complex designs increase the likelihood of security flaws.
- …

Build your application security maturity



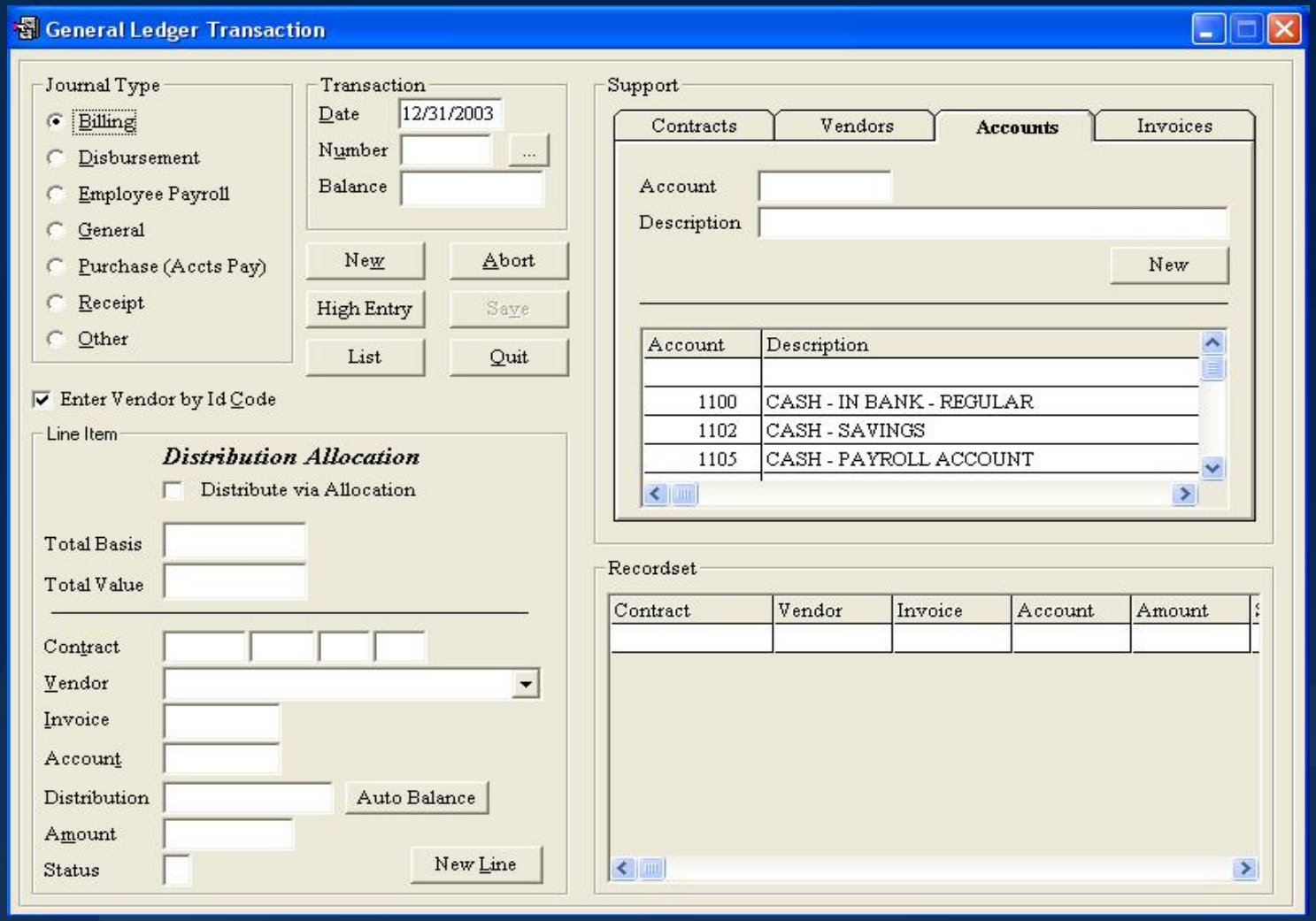
FORRESTER®

A look at trends going forward

The consumer experience

FORRESTER®

The enterprise experience

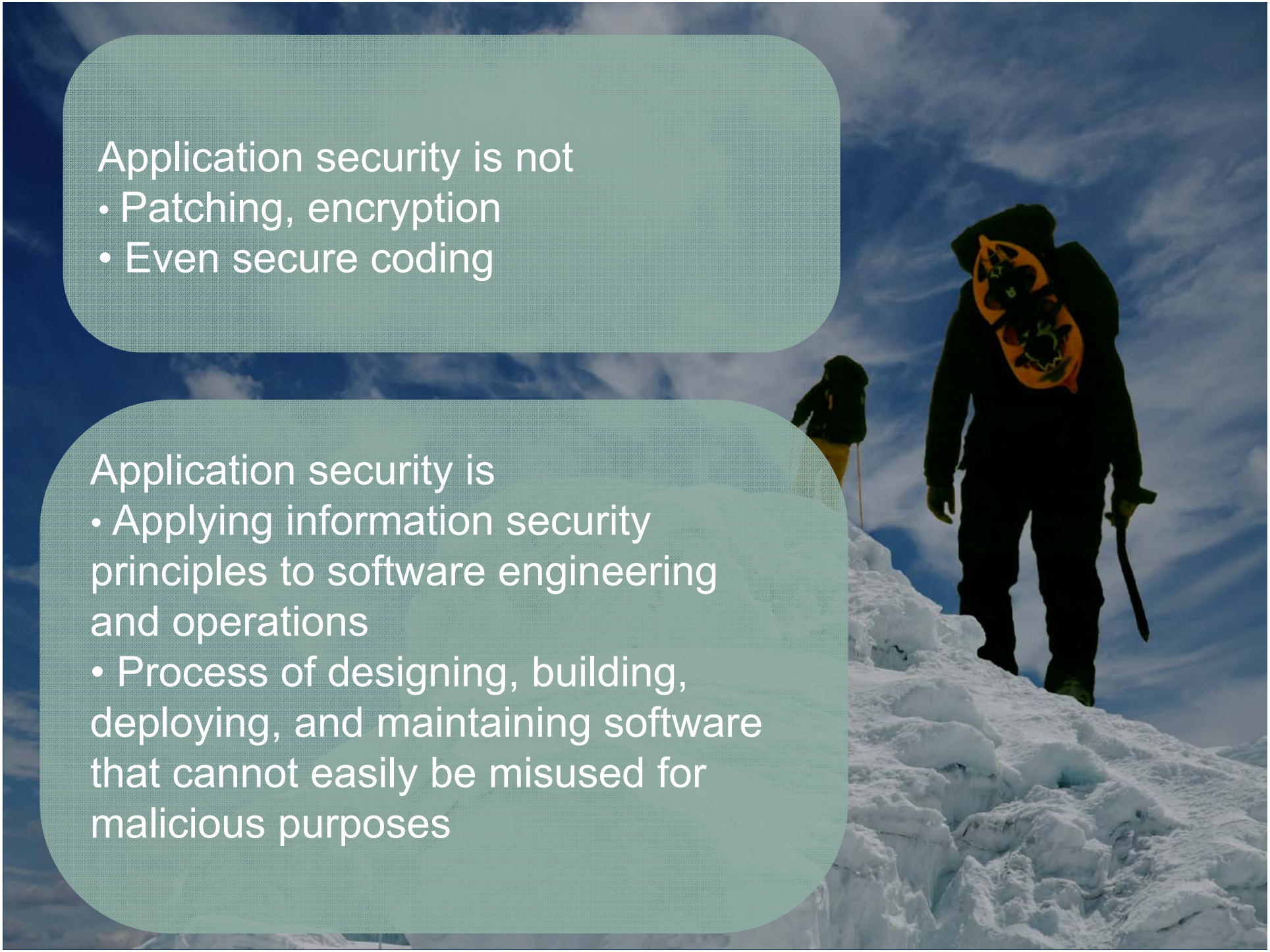



The consumer and
enterprise experience
gap will disappear …

Prediction …

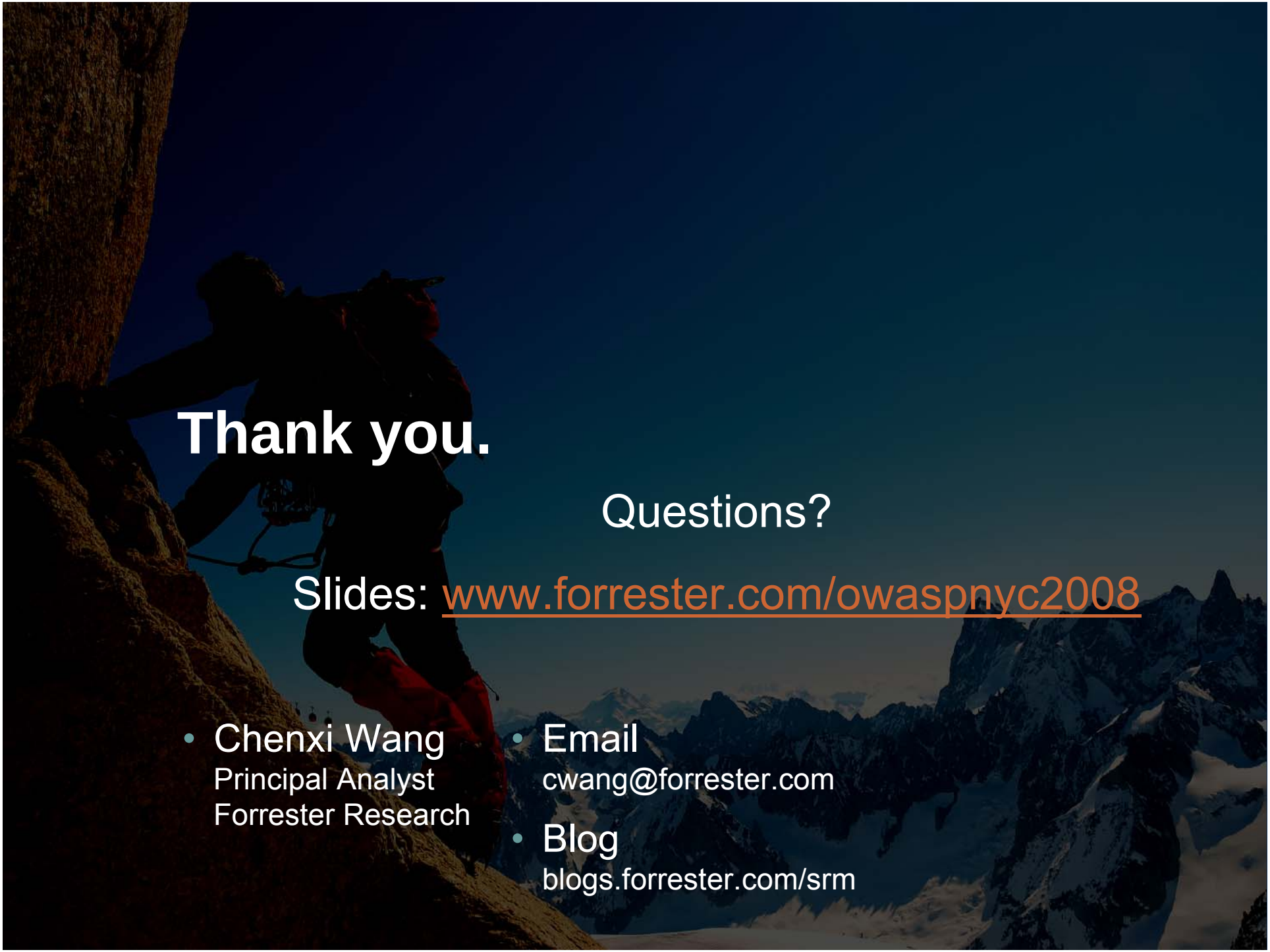
- Reputation will be pervasive
  - » Reputation for program/code
- Ways to recognize invariants in the code
  - » Innovative fingerprinting technology
- Ways to analyze software
  - » Faster than current emulation technology
- Reputation will live in the cloud
- Optimization on the gateway

Summary


Application security is not
• Patching, encryption
• Even secure coding
Application security is
• Applying information security principles to software engineering and operations
• Process of designing, building, deploying, and maintaining software that cannot easily be misused for malicious purposes

Information resources for application security

- Open Web Application Security Project (OWASP) Top Ten at http://www.owasp.org
- Web Application Security Consortium (WASC) Threat Classification at http://www.webappsec.org
- Chenxi Wang, Ph.D. “Managing Application Security from beginning to end”, Forrester research report, October 2007.
- Michael Howard. “Writing Secure code for Windows Vista,” Microsoft Press, 2007
- Gary McGraw. *Software Security: Building Security In.* Addison-Wesley, 2006
- “Improving Web Application Security: Threats and Countermeasures,” Microsoft Press, 2003



Thank you.

Questions?

Slides: www.forrester.com/owaspnyc2008

- Chenxi Wang
  Principal Analyst
  Forrester Research
- Email
  cwang@forrester.com
- Blog
  blogs.forrester.com/srm