endava

SECURITY WITH REAL TEAMS

secure development with
non-specialist software developers

Eoin Woods
Endava
@eoinwoodz

endava

SECURITY IN REAL TEAMS

endava

SOME COMMON CONCERNS

Will this cost a lot?

Where do we start?

Who is involved?

What tools do we use?

Can we do this with agile?

Won't this slow everything down?

endava

SOME OBSERVATIONS

- Some **individuals** will find it **fascinating**, some will **hate** it
- Teams will need **guidance and inspiration**
- Teams need to **own their security process**
  - But a clearly defined **starting point** and **standards** very valuable
- A clear **roadmap** helps to avoid overload

endava

SOME USEFUL TACTICS

- Form a group of **security champions** - invest in them
  - involve many roles (BA, developer, tester, architect, …)
- **Communicate importance** of security from the top
  - and from the customer
- Make the **right thing the easy thing**
  - checklists and templates, clear guidance, packaged tools
- Be prepared for the process to **take time**

endava

USUALLY A GRADUAL PROCESS

EXPERT APPLICATION SECURITY TEAM

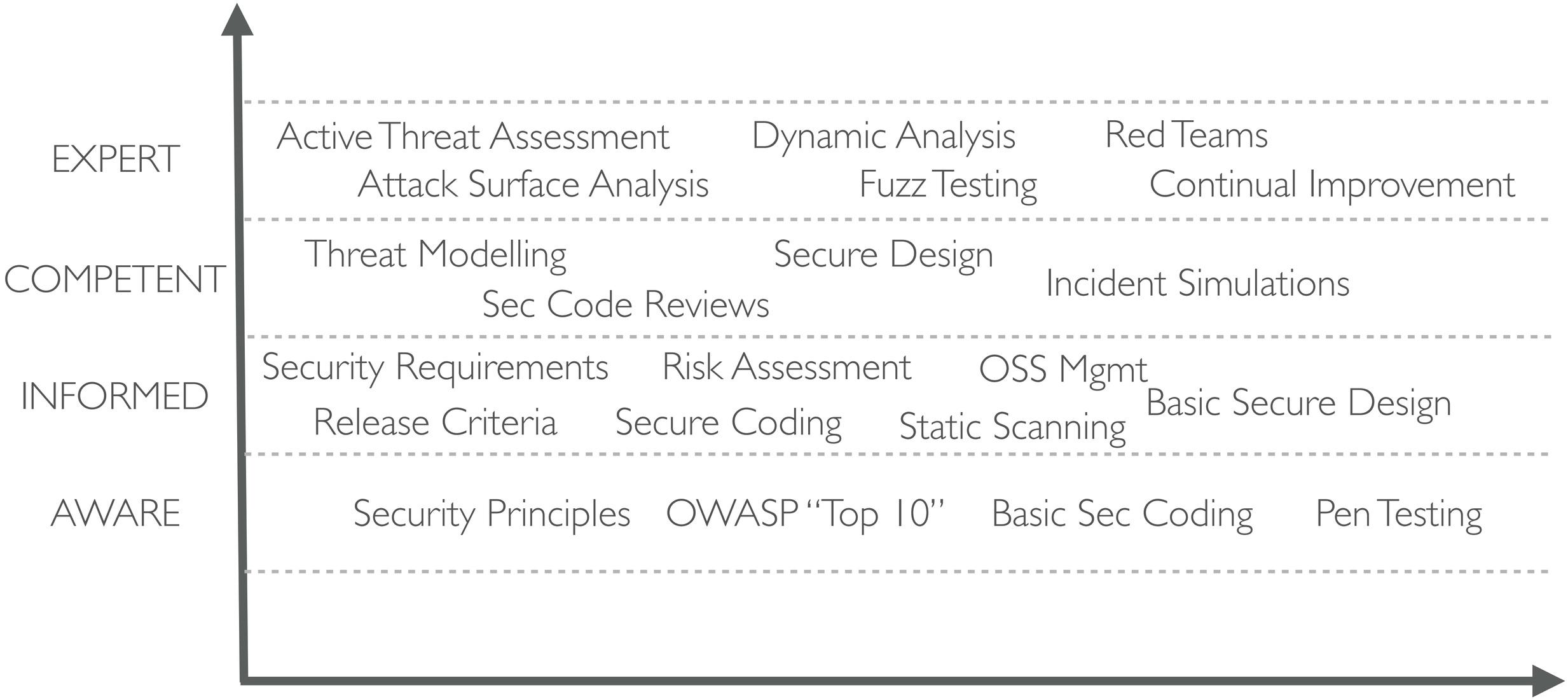
COMPETENT APPLICATION SECURITY TEAM

INFORMED APPLICATION SECURITY TEAM

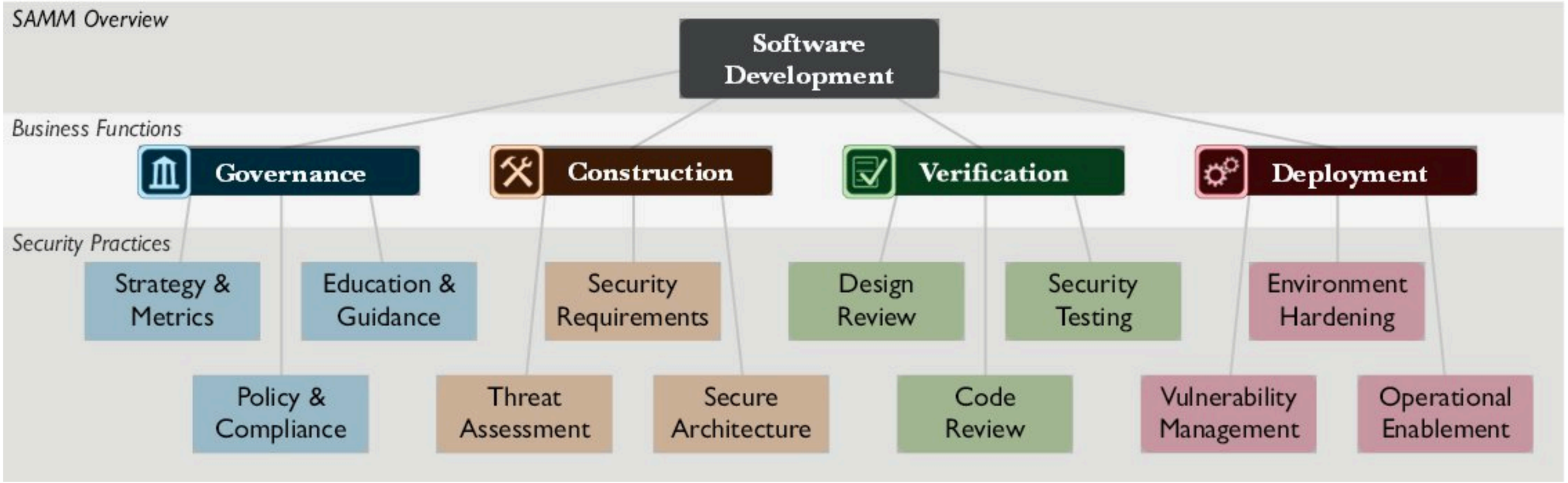
SECURITY AWARE TEAM

NO SECURITY PRACTICE

EXAMPLE CAPABILITY PLAN

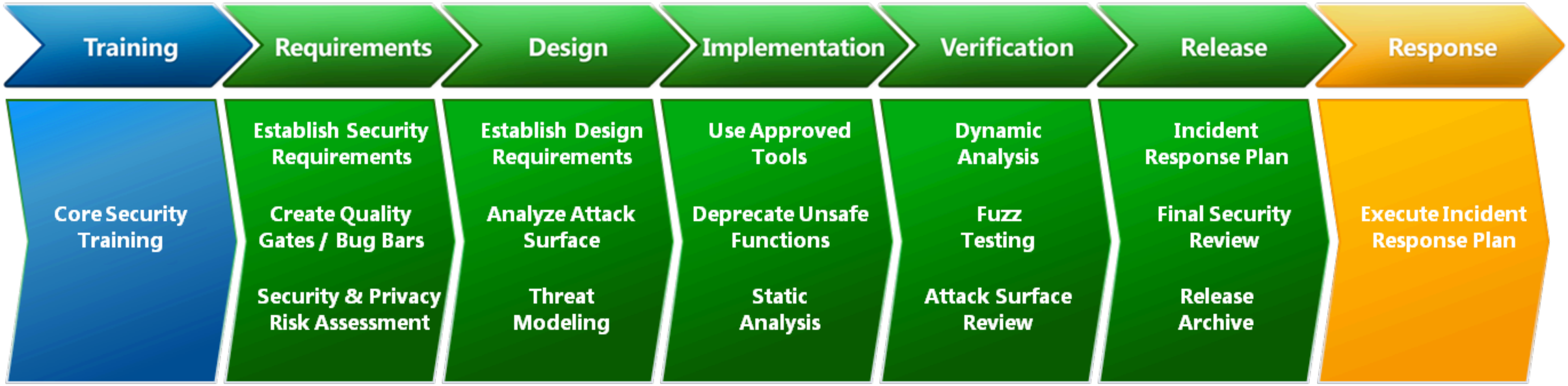


OWASP SAMM



http://www.opensamm.org

MICROSOFT SDL



https://www.microsoft.com/en-us/sdl/

Thank you for your attention

Questions?

Eoin Woods
Endava
eoin.woods@endava.com
@eoinwoodz

SUMMARY OF OUR DAY

endava

SUMMARY - INTRODUCTION

- We've looked how to **improve** system **security**
  - we need to be **risk** and **principle** driven
- Security requires: **People**, **Process** and **Technology**
  - the weakest of the three is your security level
- Security needs to be **designed** in
  - its very difficult and expensive to add later

SUMMARY - INTRODUCTION

- Be guided by **risks not** security **technologies**
  - threat risk models (STRIDE and DREAD); attack trees
- Get the **experts** involved for **significant risks**
  - and never invent your own security technology!

endava

SUMMARY - REMEMBER …

Never stop asking "**why?**" and "**what if?**"

critically important security questions!

endava

SUMMARY - WEBAPP SECURITY

- **Much** of the **technology** we use is inherently **insecure**
  - Mitigation needs to be part of application development
- **Attacking** systems is becoming **industrialised**
  - Digital transformation is providing more valuable, insecure targets
- Fundamental **attack vectors** appear **again and again**
  - Injection, interception, page manipulation, validation, configuration, …

endava

SUMMARY - OWASP

- **OWASP** - The Open Web Application Security Project
  - Largely volunteers, largely online, improving state of software security
  - Research, tools, guidance, standards
  - Runs local chapters for face to face meetings

- "**OWASP Top 10**" project lists top application security risks
  - Data-driven list of most significant threats to webapps
  - Referenced widely by MITRE, PCI DSS and similar
  - Updated as threats change (2003, 2004, 2007, 2010, 2013, 2017)

endava

OWASP TOP 10 WEB SECURITY THREATS

1. **Injection** Attacks
2. Broken **Authentication**
3. Sensitive **Data Exposure**
4. XML External Entities (**XXE**)
5. Broken **Access Control**
6. Security **Misconfiguration**
7. Cross Site Scripting (**XSS**)
8. Insecure **Deserialisation**
9. Component **Vulnerabilities**
10. Insufficient **Logging** and **Monitoring**

endava

SUMMARY - WEBAPP MITIGATIONS

- Don't trust **clients** (browsers)
  - Validation, authorisation, …
- Identify "**interpreters**"
  - Escape inputs, use bind variables, …
  - Command lines, web pages, database queries, …
- **Protect** valuable **information**
  - At rest and in transit
  - Use encryption judiciously
- **Simplicity**
  - Verify configuration and correctness
- **Standardise** and **Automate**
  - Force consistency
  - Avoid configuration errors

endava

SUMMARY - WEBAPP SECURITY

- Most **real attacks** exploit a **series** of **vulnerabilities**
  - Each vulnerability may not look serious, the combination is

- Most **mitigations not difficult** but need to be **applied consistently**

… and may conflict with other desirable qualities

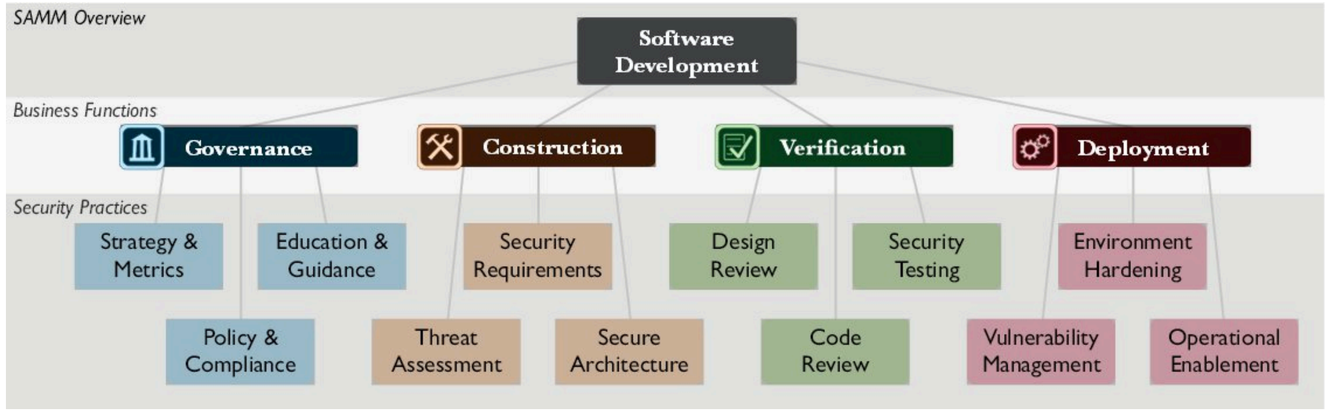
endava

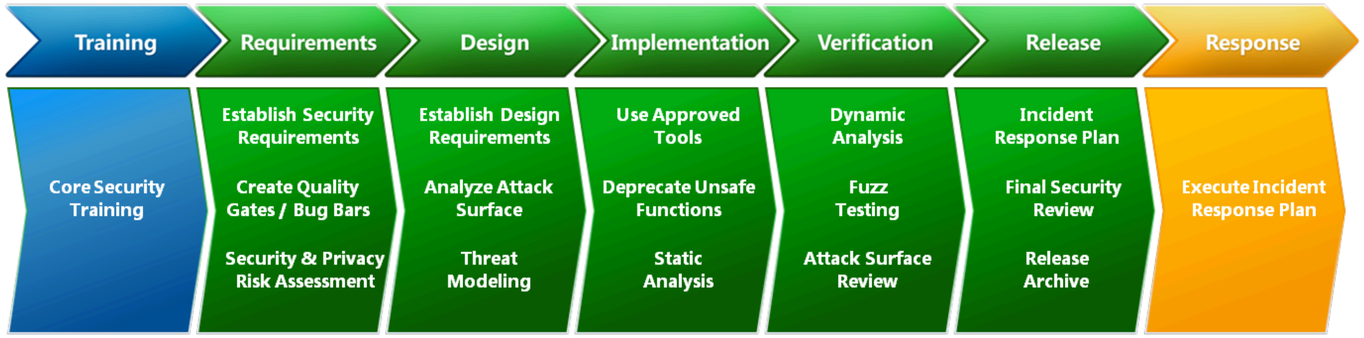
SUMMARY - TEN KEY PRINCIPLES

- Assign the **least privilege** possible
- Separate **responsibilities**
- **Trust cautiously**
- **Simplest** solution possible
- **Audit** sensitive events
- **Fail securely** & use secure defaults
- Never rely upon **obscurity**
- Implement **defence in depth**
- **Never invent** security technology
- Find the **weakest link**

GETTING TEAMS DOING IT

Continuous Process





Towards Secure SDLC

endava

RESOURCES

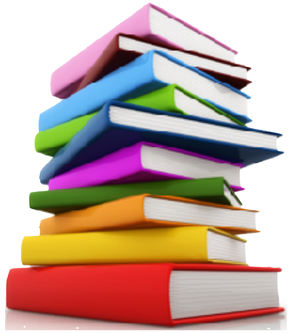
- **OWASP** - http://www.owasp.org
  - Top 10, cookbooks, guides, sample apps, tutorials, …
- **Microsoft SDL** - http://www.microsoft.com/security/sdl
  - complete security development lifecycle with resources
- **Elevation of Privilege** game- http://tinyurl.com/eopgame
  - card game which helps to explain and drive threat modelling
- **Trike** - http://www.octotrike.org
  - alternative threat modelling approach
- CAPEC, **CWE** - http://{capec,cwe}.mitre.org
  - threat and vulnerability lists

endava

RESOURCES

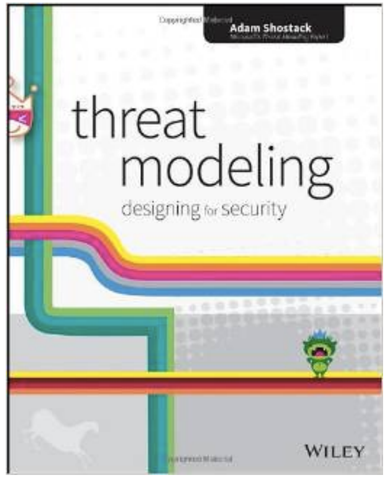
- **CPNI** - http://www.cpni.gov.uk
  - UK government support for cyber security
- **US Government CERT** - https://www.us-cert.gov
- **CMU's CERT** - http://cert.org
  - vulnerability monitoring and alerting
- **WASC** - http://www.webappsec.org
  - similar organisation to OWASP
- **SANS Institute** - http://www.sans.org
  - security research and education

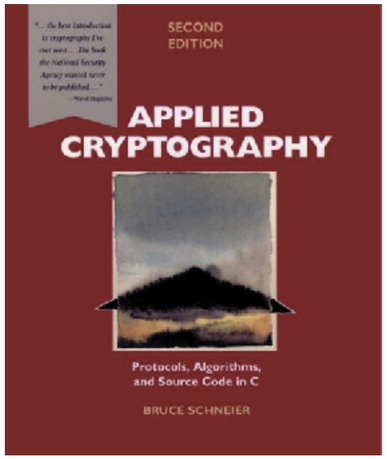
REFERENCES

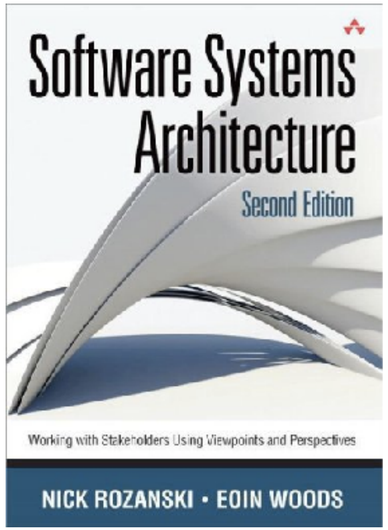


- UK Government NCSC Security Principles:
  https://www.ncsc.gov.uk/guidance/security-design-principles-digital-services-main
- NIST Engineering Principles for IT Security:
  http://csrc.nist.gov/publications/nistpubs/800-27A/SP800-27-RevA.pdf
- Short intro to McGraw's set:
  http://www.zdnet.com/article/gary-mcgraw-10-steps-to-secure-software/
- OWASP Principles set:
  https://www.owasp.org/index.php/Category:Principle

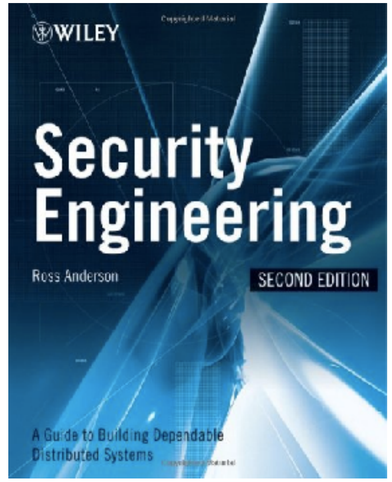
BOOKS


Adam Shostack
threat modeling
designing for security
WILEY

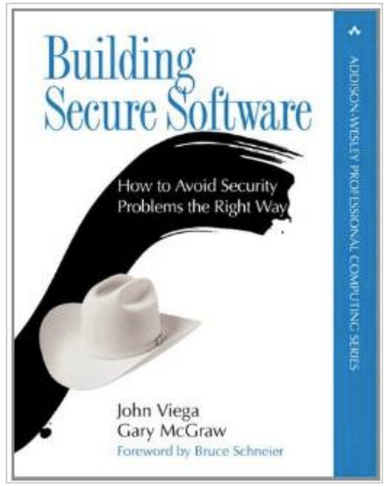

SECOND EDITION
APPLIED CRYPTOGRAPHY
Protocols, Algorithms, and Source Code in C
BRUCE SCHNEIER

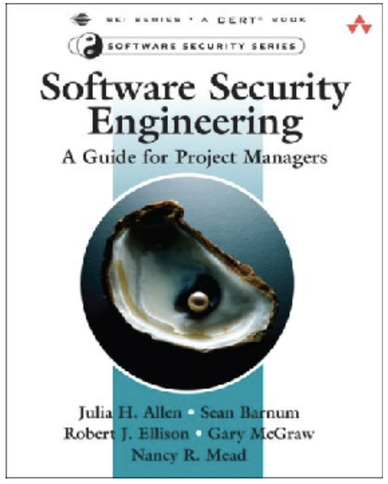

Software Systems Architecture
Second Edition
Working with Stakeholders Using Viewpoints and Perspectives
NICK ROZANSKI • EOIN WOODS

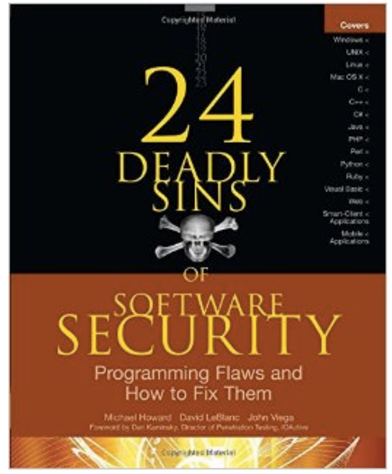

INTERNATIONAL BESTSELLER
Over 150,000 copies sold in hardcover
Secrets & Lies
Digital Security in a Networked World
With new information about post-9/11 security
Bruce Schneier


WILEY
Security Engineering
Ross Anderson
SECOND EDITION
A Guide to Building Dependable Distributed Systems


Addison-Wesley Software Security Series
SOFTWARE SECURITY
BUILDING SECURITY IN
GARY McGRAW
Foreword by Dan Geer


Building Secure Software
How to Avoid Security Problems the Right Way
John Viega
Gary McGraw
Foreword by Bruce Schneier
ADDISON-WESLEY PROFESSIONAL COMPUTING SERIES


SOFTWARE SECURITY SERIES
Software Security Engineering
A Guide for Project Managers
Julia H. Allen • Sean Barnum
Robert J. Ellison • Gary McGraw
Nancy R. Mead


24
DEADLY SINS
OF
SOFTWARE SECURITY
Programming Flaws and How to Fix Them

Thank you for your attention

Questions?

Eoin Woods
Endava
eoin.woods@endava.com
@eoinwoodz