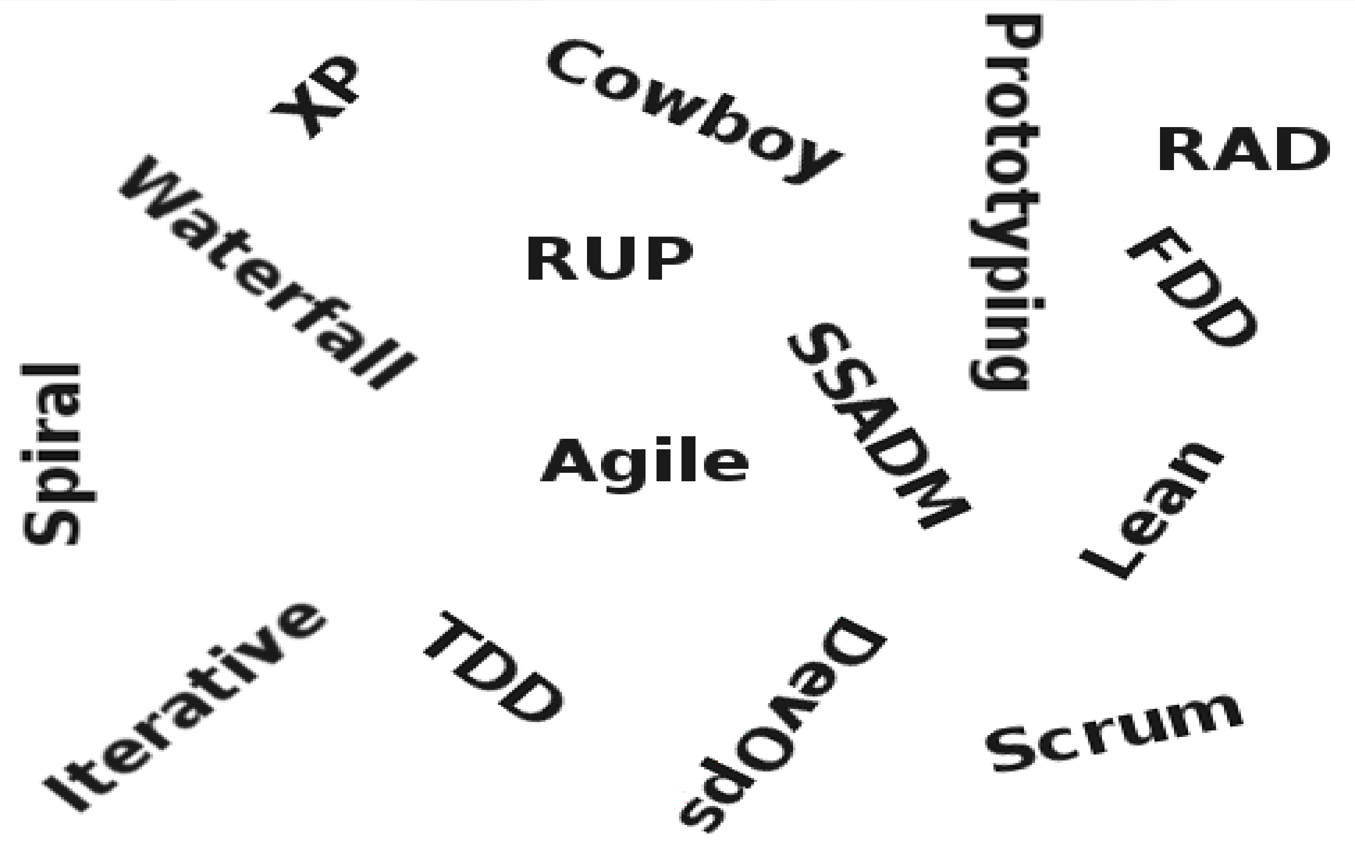
Securing It All



Greg Sternberg, MSc, CISSP, TOGAF

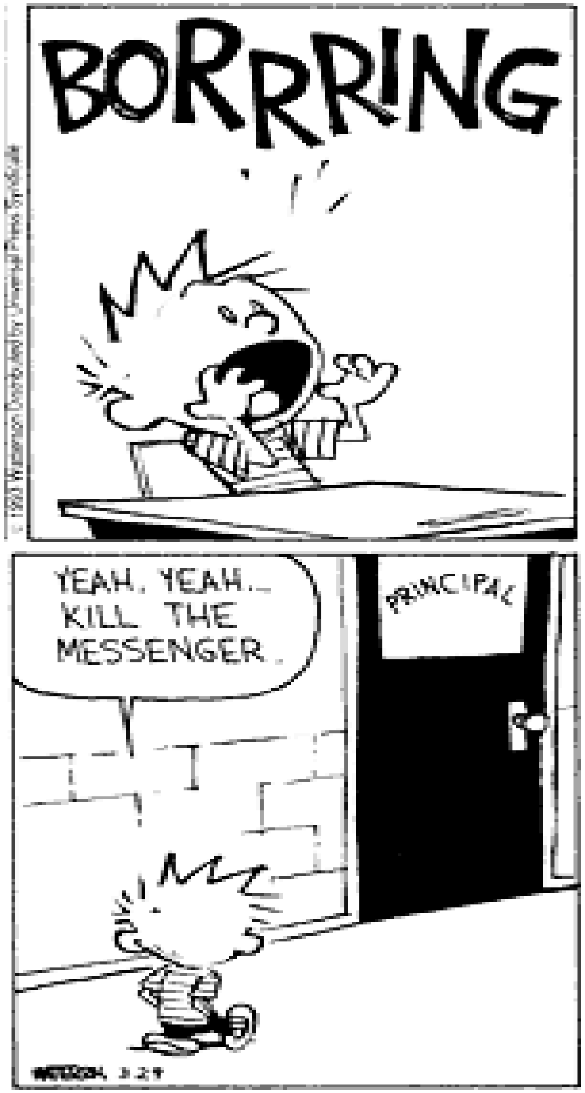
Must Secure Regardless of the Methodology

- Application security fundamentals are overlooked

Back To The Basics

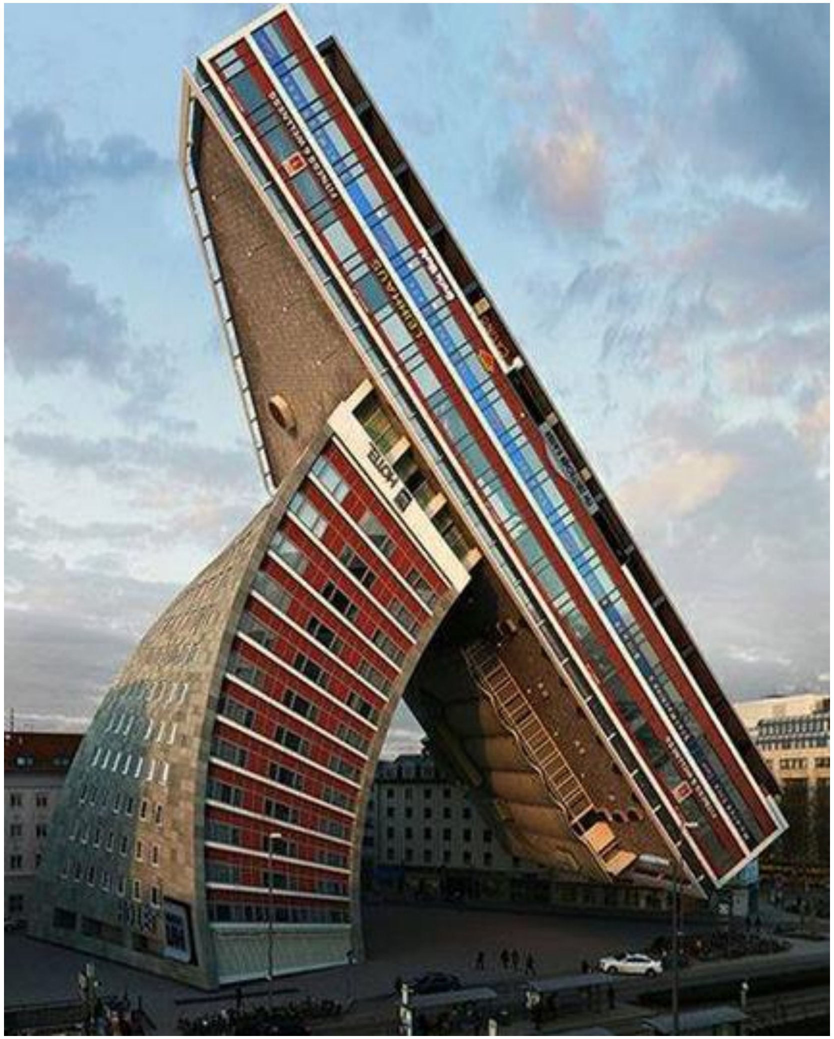
- What hardware do you have?
  - Versions? Owners? Licenses?
- What software do you have?
  - Versions? Owners? Licenses?
- What IPs do you have?
  - Internal? External?
- What URLs do you have?
- It’s not about the methodology; it’s about the data

Training/Education/Awareness



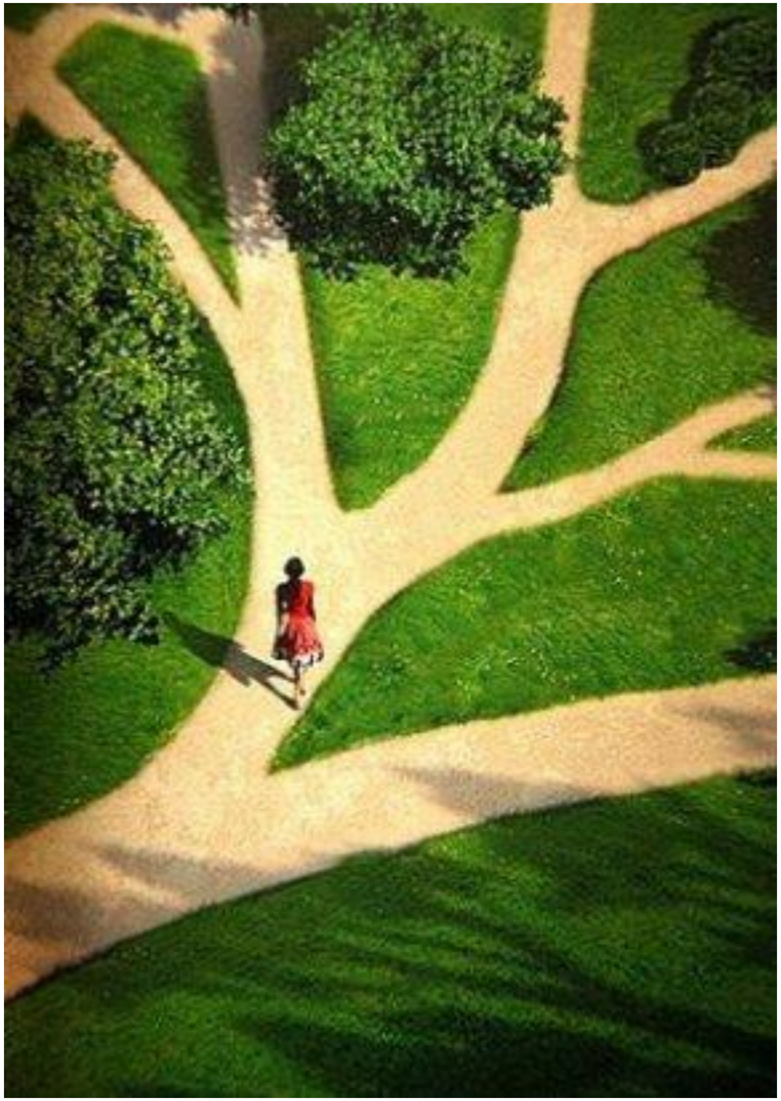
- Continuous (a.k.a. sneaky)
- Make it personal and pertinent
- Different strokes for different folks
- Application security training
- Gamification/Edutainment
  - Are You Smarter Than a 5th Grader?
  - What’s wrong with this code?
- Who’s been trained?

got plan?



- If you don't have pictures you don't know what you have
  - Whatcha talking about?
    - Logical diagram
  - Where’s my data?
    - Data flow diagram
  - Make it so
    - Network diagram
- What are your threats?
- What’s our risk?
- Who are your vendors?

Where oh Where Does My Data Flow?

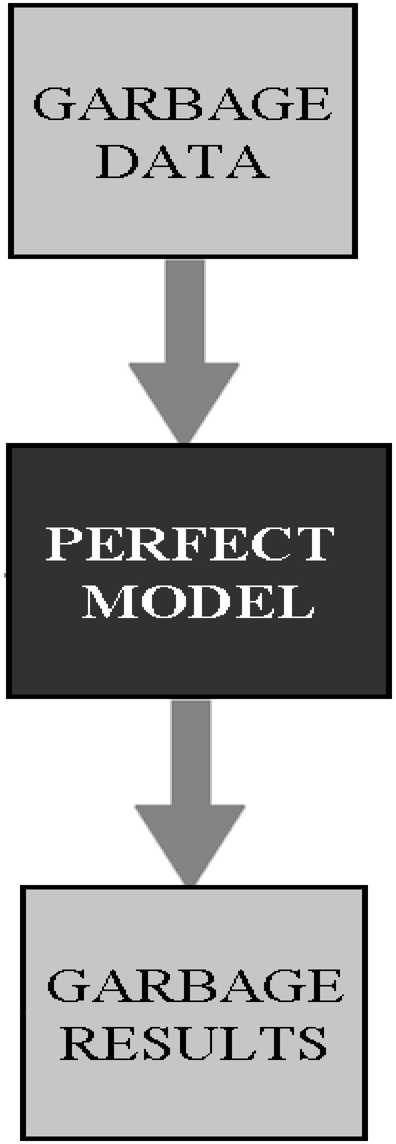


- What type of data is it?
- Where is the data?
- How is it stored?
- Who/What/How can see it?
- Who/What/How is making copies of it?
- Who/What/How has CRUD to it?

Do You Scan?

- If your security department isn’t as staffed as your development group then:
  - (Continuous) Dynamic scanning
  - (Continuous) Static scanning
  - Penetration testing
  - (Appropriate) Manual *security* review
  - Where are the results?
  - How frequently do you scan?

G.I.G.O.



- Four/Six of OWASP top 10 are input related
- “All input is evil unless proven otherwise”
- It’s just good programming practice
- Garbage comes from more than entry fields
- Not validating is ass.u.me(ing) things

Patching

- What versions do you have?
- How do you know there are updates?
- What’s your vendor policy?
- How, When, Why, Who, What, How
  - OSes
  - App / Web servers
  - Applications / servers
  - Library components
  - BIOS
  - Browsers
  - Databases
  - Network (firewalls, routers, …)
- Verify the patch

When It Comes To Data Not Everyone Is Equal

- Least privilege
- APIs (SOAP, RESTFull, …)
- Give me a reason
- Things should *never* run as root or administrator
- Admin accounts are bad
- Time boxing is your friend

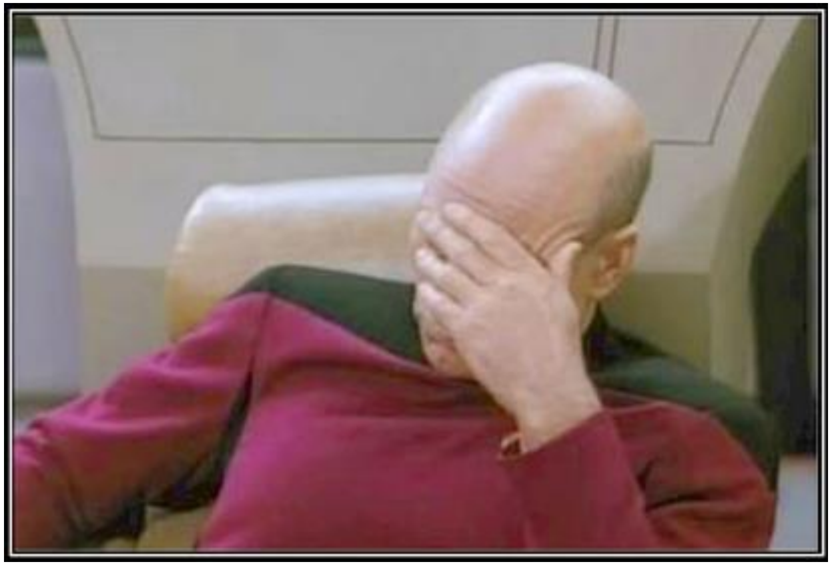
Deploy It Safely

- *Never* allow default anything
- Log management
- Backup and recovery
- Monitor for the unexpected and unexplained and unforeseen
- Avoid the head slap

If The Basics Aren't Covered, The Rest Doesn't Really Matter

- 5 +- 2
  - What do you have?
  - Edutainment
  - Scanning
  - Input/Output sanitization
  - Patching
  - Least access
  - Make it safe; not available
- Tighten at the next go around

Reasons Not To Secure



- Security doesn't fit with our methodology
- Security can be done on demand, piecemeal, iterative, as needed, …
- We'll do security when…
- Security is hard/expensive/time consuming/…
- I'm not a target; why do I need security?
- Security keeps finding bugs in the code
- I've done what I need to for security
- I'm compliant therefore I'm secure

Engage

- Avoid saying ‘no’
- Inception
  - How will security fit in?
- Requirements
  - What security do we need?
- Architecture
  - How will we do security?
- Development
  - Make it so, securely
- Test
  - Did we miss anything? Due diligence
- Operations
  - Putting the pieces into place, securely

Questions?

(And maybe even answers!)

Supporting Slides

Inception

- Setting and understanding security and project expectations at the very beginning avoids the ‘last minute panic’ (or worse a headline)
- How will the project be used?
    - i.e. web facing, internal, customer facing, …
- What are the risks?
- What compliance and regulations need to be considered?
    - i.e. PCI-DSS, HIPPA, EU regulations, SOX, COBIT, state regulations, …
- Security milestone/quality gates
- Security assessment document

Requirements

- Security must be a requirement, just like everything else. Some project require extensive security while others don’t.
- Does access to the system need to be controlled?
- What are the confidentiality needs?
- What are the integrity needs?
- What are the availability needs?
- What are the compliance and regulatory needs?

Architecture

- The question is more than “How will the system be put together?” but also “How will the system be secured?”
- Diagrams are vital
  - Logical – “High level”
  - Data – “Where o’ where does my data flow?”
  - Network / Deployment – “Make it so”
- How are the threats addressed?
- Appropriate security
- Connections (software, hardware and people) and Assets

Development

- “Where the rubber hits the road”
- Principles
  - “All input is evil unless proven otherwise.”
  - G.O.G.I.
  - Compiler warnings are your friends
  - Minutes securing a design will save hours of tedious (re)coding
  - “I can’t overstate how completely evil complexity is”
  - Just say no
  - Good for the project, good for the soul, but only if we all know
  - It will fail – so do it well
  - Never assume
- Threat modeling (STRIDE and AADSVL/Countermeasures)
- Continuous scanning

Testing / QA

- There are two components to security testing - one is testing to see if the security requirements were implemented, the other is testing for the ‘head slap’
- Ensure comprehensive vulnerability, static and penetration scans are run and resolved
- Don’t ask what the system can do for you; ask what you can do to the system
  - i.e. change a URL, try common attacks - ‘ or ‘1’=‘1’
- Users do what they should; hackers do what they can
- It’s not about thinking like an attacker; rather avoiding being low hanging fruit
- Let’s be more secure than our neighbor (i.e. let’s not be caught by the simple/obvious stuff)
- Look at the system – do configuration files contain passwords? Can sensitive data be accessed by any account? Are admin functions restricted to admins? Are default accounts disabled?

Operations / Deployment / Installation

- The last thing done on a project is the first thing tested by hackers
- Are default passwords and accounts disabled?
- How will the hardware, software, application, operating system, etc… be patched?
- How are logs managed? Are there audit logs?
- How is the system backed up? How is it restored? Has that process been validated?
- Is the system monitored for the unexpected?

Security (yes, security has a role)

- We’re more than Dr. No.
  - We want you to succeed (we like paychecks too)
  - Partnership
- Security gates
- Subject matter experts
- Help with threat modeling, risk evaluation, compliance, regulations, privacy, mitigation, …
- Understand and communicate about the changing security landscape
- Assessments, validate and review solutions
- Metrics