Engineering Secure Software

SECURITY RISK ASSESSMENT

Why do we study risk?

- Many outcomes are possible, not all are probable
- Enumeration
- Prioritization
- Discussion

Naïve Security Risk Assessment

- The naïve approach
  - Write down your worst fears for the system
  - Try to avoid those things
- Cons
  - Requires a big “bag of tricks”
  - Easily overwhelming for security

What is risk?

- p(occurrence)*impact
- The risk associated with an event is the probability that the event will happen times the impact magnitude of the event
- For the math-oriented… *expected value*
- Matches how people generally think
  - Low p(occ), high impact
    … terrorist attacks, struck by lightning
  - High p(occ), low impact
    … credit card theft, keeping my old truck unlocked

What is security risk?

- p(exploit)*value of an *asset*
- p(exploit)

  The probability that an exploit will occur on your system
- Asset

  A [tangible or intangible] resource of the system that has value in confidentiality, integrity, availability

p(exploit)

- Increased by *more* vulnerabilities
- Increased by a *far-reaching* vulnerability
- Increased by *discoverable vulnerabilities*
  …although you cannot rely on security through obscurity alone …
- Increased by *scope of the project*
  …although sometimes that is unavoidable…
- Other factors that we cannot control
  - Market share → exposure
  - New malicious actors (e.g. activism spike)
  - Many, many other factors that we must ignore for the sake of simplicity
- Thus, we generally assume p(vulnerability) is proportional to p(exploit)

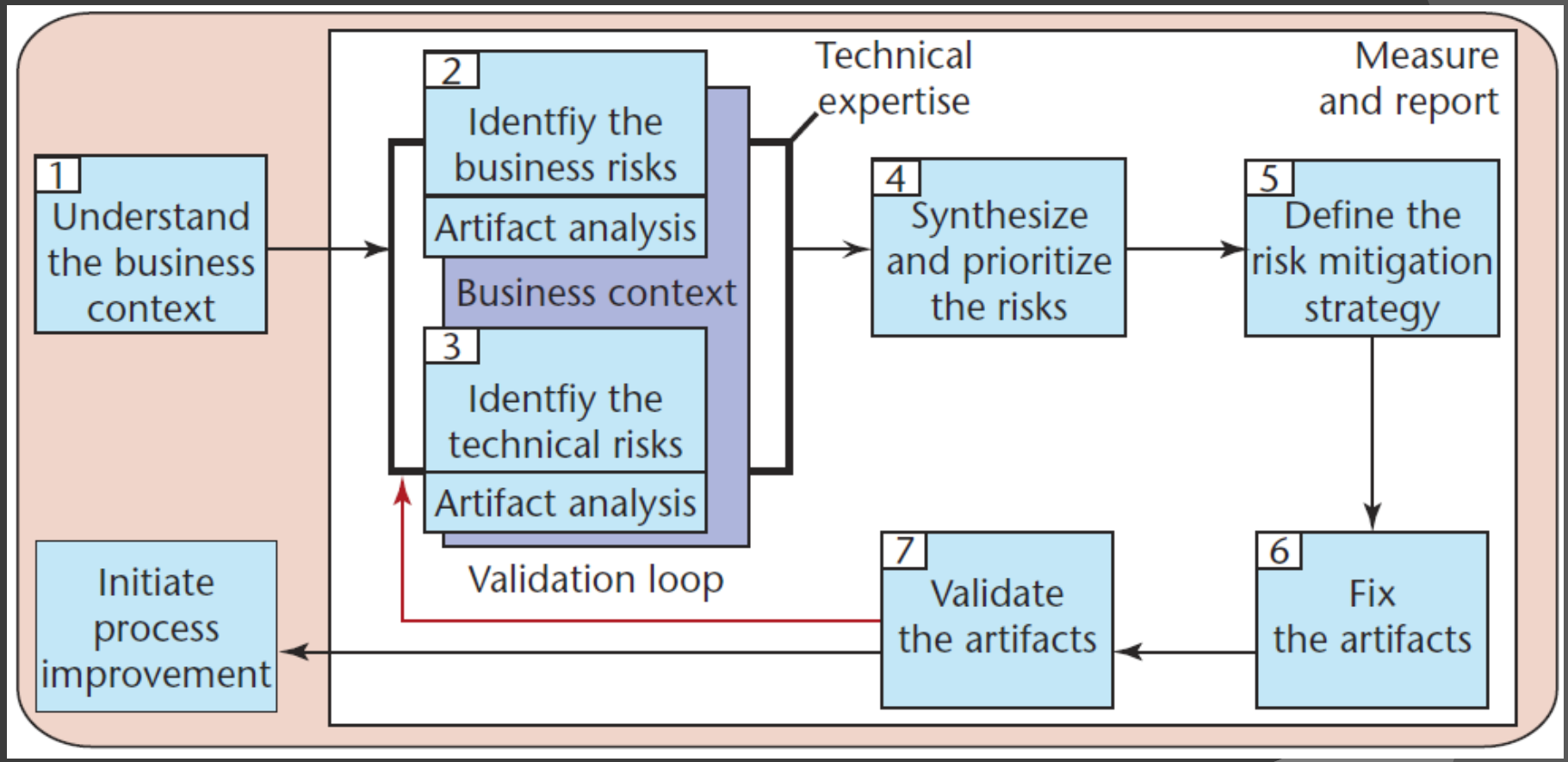
Assets

- An asset is data store or a component that the deployed system must protect
- Every software system has assets
  - Domain-specific — *e.g. patient records*
  - Domain-independent — *e.g. passwords*
  - Intangible properties — *e.g. availability*
- These can be identified at the *requirements* and *design* stages
- Assets exist in the deployed system, so source code is not (necessarily) an asset

Places where assets live

- Database tables
- User-supplied data
- Configuration files
- Configuration consoles
- File systems
- Security feature inputs
- Logs
- Sandboxing features
- Built-in examples
- Network traffic
- Cookies
- User interfaces

Risk Assessment in Process



- From: http://www.cigital.com/papers/download/bsi3-risk.pdf

The Planning > The Plan

- One of the most important elements of risk analysis is the process itself
  - Discussions that are brought up
  - Fighting over the mitigation strategies
- Communication is very important at this stage
- Assessing the *change* in risk is more sound than the final numbers
  - New assets?
  - Increased p(exploit)?

Abuse Cases vs. Risk Assessment

- Abuse & Misuse Cases
  - Involves planning
  - Potentially infinite
  - Emphasize domain
  - Scenario-driven
  - Originates from abusing functionality
  - What if?
- Risk Assessment
  - Involves planning
  - Potentially infinite
  - Emphasize all risks
  - Quantitative
  - Originates from CIA, assets, p(exploit)
  - What might?

Protection Poker

- A combination of product & process risk
  - Trace stories to assets
  - Quantify the risk for prioritization
    - Ease of attack
    - Value of the asset
  - Discuss the elements of the risk
- Originally designed for agile processes
  - Assumes we are in a sprint
  - Not comprehensive, but just-in-time

Story Points Estimation

- In PP, we use story points
  - Dimensionless (unit-less)
  - Should *not* translate to hours, effort, etc.
- Limited to a few choices
  - Why argue over 51 vs. 50?
  - Exponential in scale (~Fibonacci)
- Ease of attack ~ p(vulnerability)

Protection Poker in Action

- Identify some assets
- Calibrate your asset values
- Calibrate your ease of attack
- For each item
  - Trace the item to the assets affected
  - Vote on affected asset values, as needed
  - Vote on ease of attack
- Examine two rankings
  - Ease*Max(value)
  - Ease*Sum(value)