Engineering Secure Software

MISUSE AND ABUSE CASES

What is a requirement?

- How do you define it?
- If done well, what are they useful for?

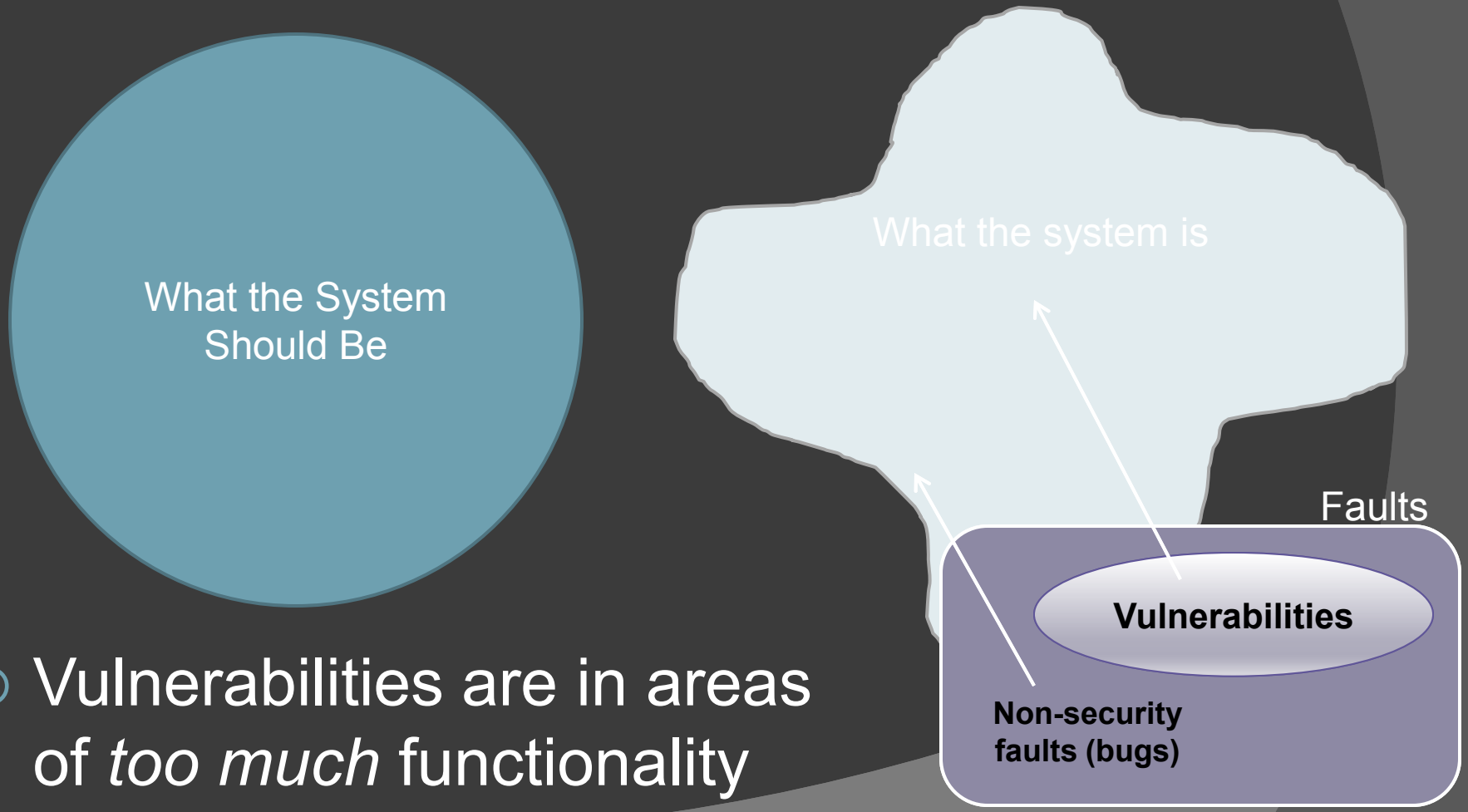
Key Requirement Properties

- What the system should…
  - Do
  - Not do
- Who interacts with the system (actors)
- Highly domain-specific
- Describe how the surrounding environment has changed as a result of the system

Security is Not a Set of Features

- *Secure* is an emergent property of software
  - “Being dry” in a tent in the rain
  - Being secure is the result of many, many factors, not one feature (e.g. SSL)
- …so requirements documents should not just be a list of features

Unintended Functionality



- Vulnerabilities are in areas of *too much* functionality

Use Case Review

- Use cases include:
  - Actor
  - Preconditions
  - Main flow describes the primary scenario
  - Alternative scenarios describe how the system reacts to alternative cases

Misuse & Abuse Cases

- A scenario within a use case in which an actor compromises the system
- Flow of events, but with malicious usage
- Define the harm done to the system
- Keys:
  - Domain, domain, domain.
    Don't focus on coding and design vulnerabilities here
  - Malicious actors are creative
  - Question the assumptions of the system
  - Focus on what the actor *can* do over *will* do
    (prioritize later)

Misuse vs. Abuse

- Misuse is unintentional
- Abuse is intentional

- Misuse cases are still security-related (crime of opportunity)

- Abuse cases imply the actor is actively seeking vulnerabilities

e.g. Maintain Drug Interactions

- Actor: Hospital Administrator
- Precondition: Admin is authenticated.
- Main Flow:
  1. Admin selects two different drug codes from the National Drug Codes selection menu
  2. Admin enters a minimum dosage for both drugs
  3. Admin enters text notes about the potential consequences of the interaction
  4. Admin is shown a table of patient records where the interaction rule would apply
  5. Admin saves the interaction rule

e.g. Misusing Maintain Drug Interactions

- Misuse case
  1. Main flow steps 1-3
  2. Admin is shown a set of patient records that have not been authorized for hospital administrator viewing

  Harm done: Patient privacy has been violated

e.g. Abusing Maintain Drug Interactions

- Abuse case

  Actor: attacker who has spoofed an administrator's identity

  1. Repeat Main Flow steps 10,000 times
     - Providing many rules for all different drug interactions
     - Auto-generate vague, technical text notes for each interaction rule
  2. Stop when the preview step takes over a minute to complete

  Harm done:

  - Patients are overwhelmed by alerts, so alerts become ignored
  - Availability of the system is compromised

Isn’t this infinite?

- Yes
- But even one good abuse case goes far
  - Easier to think beyond one scenario
  - Starts a discussion
  - Gets stakeholders and developers into a balanced mindset early on
  - Motivates good design decisions

Security Requirements

- Generalized forms of misuse and abuse cases
- Use-cases trace to security requirements
  - Document these in the main flow
- Also called “anti-requirements”
- E.g. from Maintain Drug Interactions
  - Hospital administrators are only allowed to view a patient’s record with explicit, opt-in indication from the patient
  - All actors are limited to 10 server requests per minute

Actor Inspiration

- Think of the best engineer on your team
- Fire them and humiliate them publicly
- Now challenge them to break your system
  - What would they go after?
  - What knowledge could they leverage the most?

Assumptions Inspiration

- What are the other non-malicious users expecting in this domain?
- What are the ramifications of violating access restrictions?
- Where could an attacker “sit in the middle”
  - Sniff the network?
  - Load a plug-in?

Today's Activity

- See course website for "Abuse & Misuse Cases"

- Systems are:
  - An auction website
  - A charity micro-lending website (e.g. Kiva.org)
  - A social networking website for model rocket hobbyists
  - A smartphone app for trading recipes with people in your neighborhood
  - A reservation system for virtual images to be run on server farms