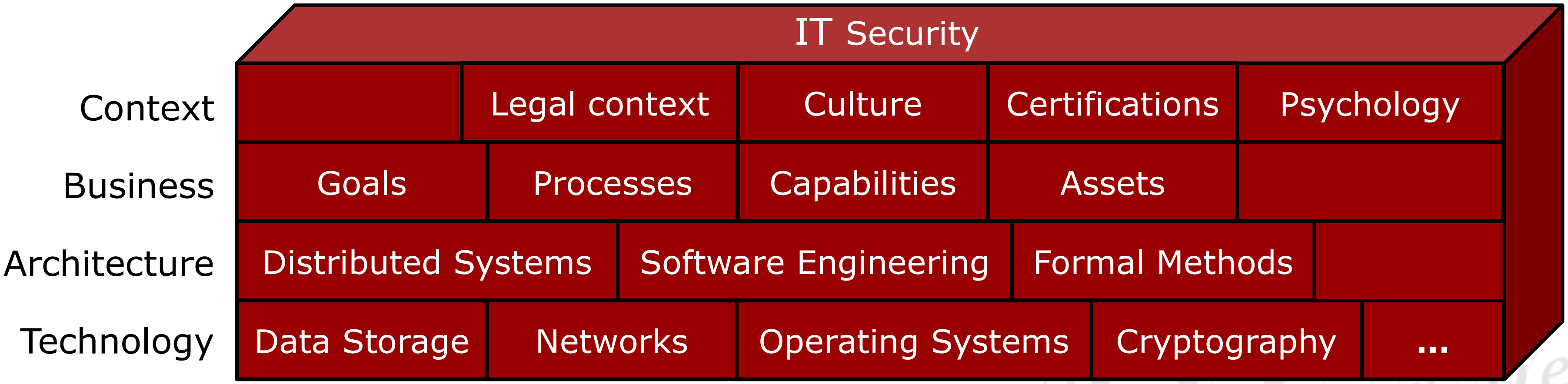
An Overview of Computer Security

DTU Compute
Department of Applied Mathematics and Computer Science

Elements of Cybersecurity

Cybersecurity must consider aspects from many domains of human activity, in addition to the theory and technologies normally attributed to the domain



… this makes it challenging, exciting and rewarding to work with!

The Basic Components

Primary Security Goals (CIA-properties)

- Confidentiality
- Integrity
- Availability

Other goals frequently listed

- Accountability
  - *Actions can be traced back to a single entity*
    - People can be made responsible for their actions
  - *Principle known as non-repudiation in cryptography*
- Privacy (e.g. privacy families defined by *Common Criteria*)
  - *Pseudonymity, unlinkability, anonymity, unobservability*
  - *There is an inherent conflict between accountability and privacy*
- Authenticity
  - *Requests or information are authentic and authenticated*
  - *Resources (both hardware and software) are genuine*

Confidentiality

Preventing unauthorised observation of information or resources (keeping secrets secret)

- War-plans, business strategies, client confidentiality (priest/lawyers), …

Particularly important in military information security

- Security models, policies and mechanisms developed to enforce the *need-to-know* principle

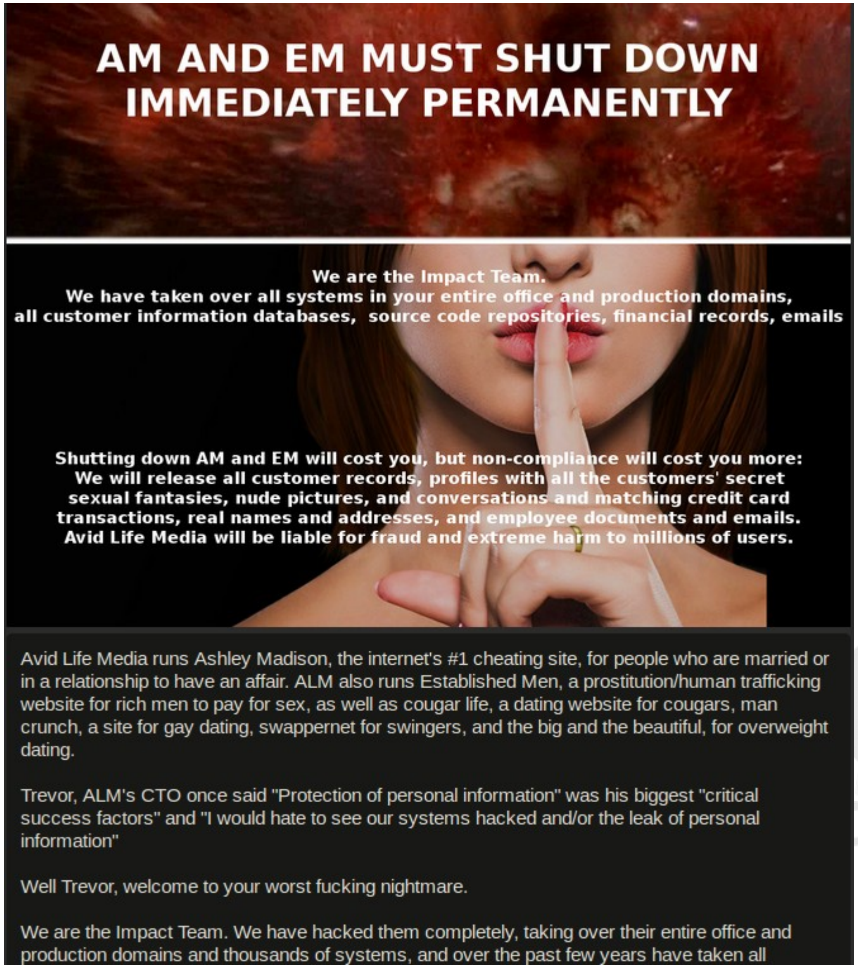
Confidentiality can be ensured with cryptography

- A cryptographic key is used to scramble (encrypt) data so that unauthorised entities cannot read it
- Authorised entities have access to a cryptographic key so that they can restore (decrypt) data to its original form

Access control mechanisms protect data from unauthorised access

Confidentiality may extend to protect knowledge about the existence of information or resources

Data Breach – Ashley Madison



Integrity

Preventing unauthorised modification of information or resources

- Data integrity pertains to the content of the information
- Origin integrity pertains to the source of the information

  Origin integrity implies authentication of the source of the information, which is part of authenticity

Two classes of integrity mechanisms:

- Prevention mechanisms

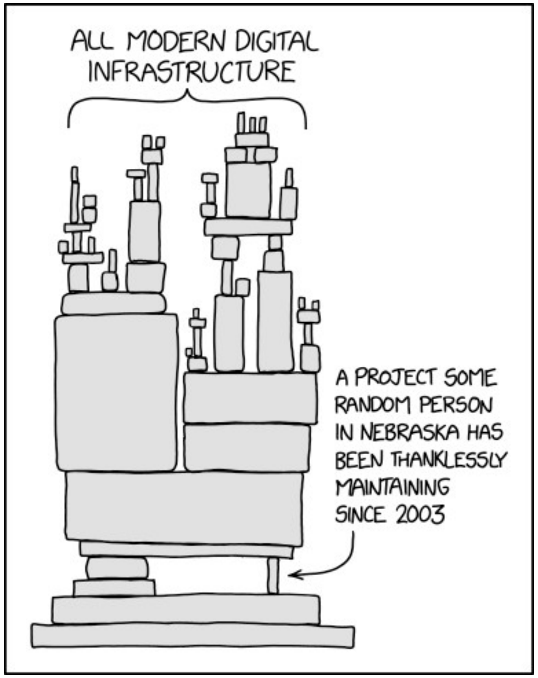
  Prevents data from being modified in unauthorised ways
  Prevents the bank's janitor from modifying my bank account, but does not prevent the bank manager from moving all my money to his own account

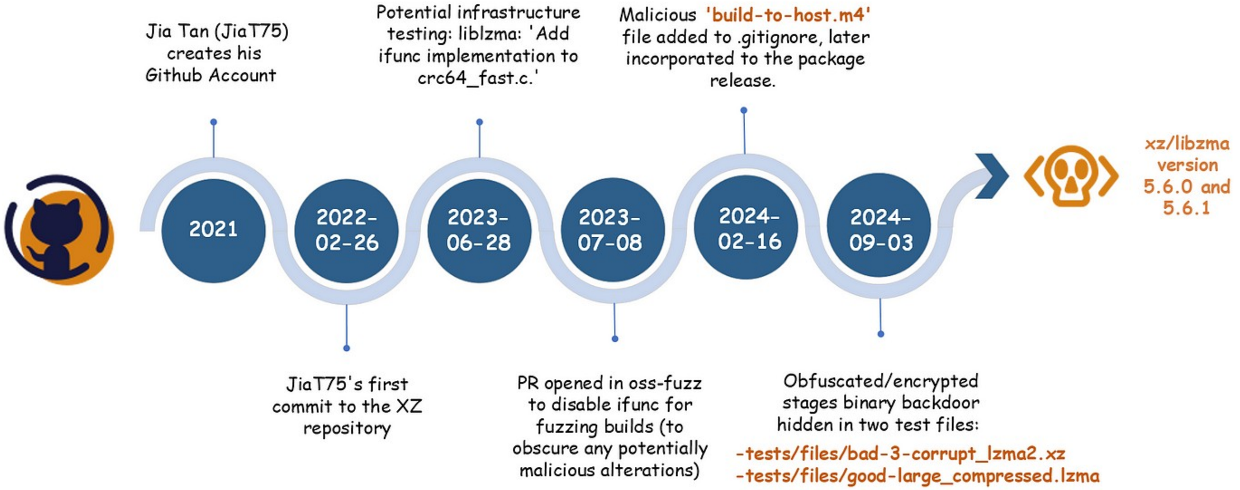
- Detection mechanisms

  Detects unauthorised modification of data after the fact
  Prevents neither of the scenarios above, but allows both to be detected and corrected

Integrity is often more important than confidentiality in commercial information systems

Supply Chain Attack – XZ Backdoor





XZ Backdoor: Navigating The Complexities Of Supply Chain Attacks Detected By Accident - Yoad Fekete: https://www.youtube.com/watch?v=CrhVXicHZJk

Availability

Availability means that the systems information and resources are available to authorised users when they need them

Attacks against availability is known as *Denial-of-Service* (DoS)

- Many spectacular DoS attacks reported in the press

Availability is devilishly difficult and most security research has focused on confidentiality and integrity

- It is easy to ensure confidentiality and integrity, simply unplug the computer and store it in a bank vault

Difficulties in ensuring availability include:

- Difficult to distinguish between high load and DoS
- Influenced by factors outside the security model

A backhoe may be used to cut power or communication supplies

Risks

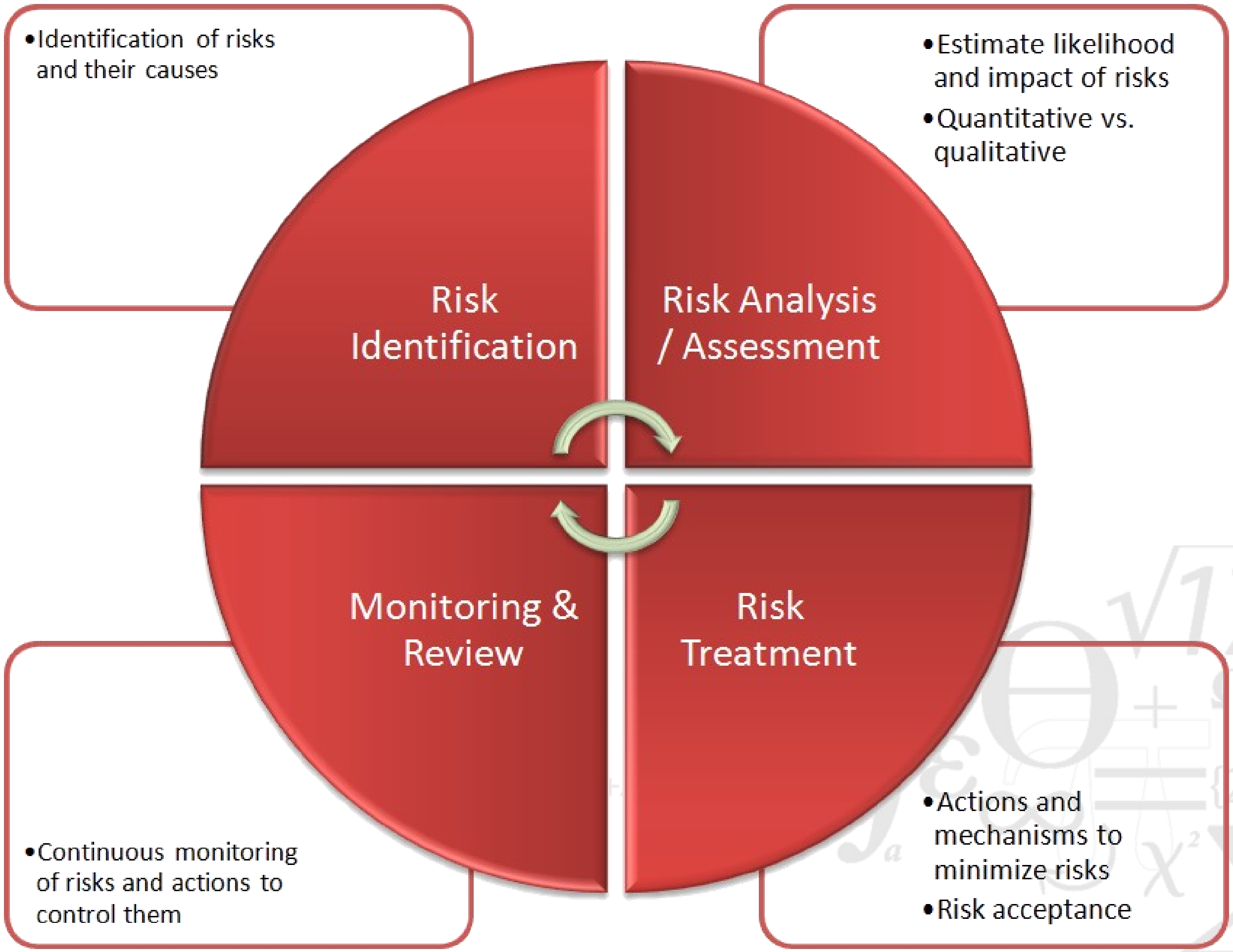
Security is concerned with management of risk
- Eliminating or reducing harm to assets

Material Harm
- Theft of property (e.g. computers, peripherals, …) or money
- Harm to people or property (e.g. health, vandalism, …)

Immaterial Harm
- Theft of Intellectual Property (incl. copyright violations)
- Harm to Intellectual Property (e.g. disclosure of trade secrets)
- Harm to reputation (e.g., website defacement, bad mouthing, …)

Risk Management
- Risk Identification
- Risk Analysis/Assessment
- Risk Treatment
- Monitoring/Review

Risk Management Cycle



Threats

A threat is a potential violation of security

- Four elements need to be present for an attack

  Threat –> vulnerability –> opportunity –> attacker (exploit)

Four major classes of threats:

- Disclosure (unauthorised access to information)
- Deception (acceptance of false data)
- Disruption (interruption or prevention of correct operation)
- Usurpation (unauthorised control of (part of) the system)

Five ways to deal with the effects of exploits:

- Prevention (remove all vulnerabilities)
- Deterrence (making exploits difficult – *but not impossible*)
- Deflection (make other targets relatively more attractive)
- Detection (as they happen or after the fact – *forensics*)
- Recovery (restore the system to a usable state)

Vulnerabilities

Weaknesses in the Security Architecture

- Weak assumptions
  - *Security requirements not specified or poorly understood*
- Weak architecture
  - *Security requirements not properly identified*
  - *Security architecture does not cover all security requirements*
  - *Security Architecture not up to date (outdated requirements)*
- Weak components
  - *Poor specification of components of the security architecture*
  - *Poor implementation of components of the security architecture*
  - *Components do not compose securely*

Weak operation

- Poor recruitment processes
- Poor security awareness

Possible Attackers

- Insiders (>*50%*)
  - Disgruntled employees
  - Guests, consultants, contract workers …
- Crackers (*hackers*)
  - Technically knowledgeable programmers
- Script-Kiddies (*cracker wannabes*)
  - Tools provided by others
- Spies (*industrial and military*)
  - Technical knowledge, technical means, many resources
- Criminals (*thieves, organized crime*)
  - Technical knowledge, technical means, many resources
- Hacktivists and Terrorists
  - Technical knowledge and means, disproportionate allocation of resources
- We need to consider: *means (method), motives* and *opportunity*

For more information: “Hacker types, motivations and strategies: A comprehensive framework” : https://www.sciencedirect.com/science/article/pii/S245195882200001X

Means of attackers

Insiders
- Knowledge of system configuration, network topologies, processes,…
- Only computing resources provided by organisation

Crackers (*hackers*)
- Able to adapt tools to configuration of target
- Able to write new tools/exploits
- Few computing resources (apart from bot-nets)

Script-Kiddies (*cracker wannabes*)
- Can only use tools provided by others (already known attacks)

Spies (*industrial and military*)
- Technical knowledge, rich computing resources, other resources

Criminals (*thieves, organized crime*)
- Technical knowledge, technical means, many resources

Terrorists
- Probably between spies and script-kiddies, but nothing is really known

Motivation for Attackers

Curiosity about how the system works

- The challenge of hacking the system
- "Ethical hacking" (expose vulnerabilities and warn owners)

  SIEM cannot tell difference between white-hat and black-hat hackers, so ***defenders must always react***

Fame

- Recognition for their achievements

Financial Gains

- Fraud, theft
- Industrial Espionage

Ideology

- Hactivism: disrupt but do not cause serious damage
- Cyberterorism: disrupt/destroy important services

Policy and Mechanisms

It is always important to distinguish between the policy and the mechanism that enforces the policy

- **Policy**: statement of what is, and what is not, allowed
- **Mechanism**: method, tool or process for enforcing a security policy

Security Policies may be defined in different ways

- Different levels of detail from very general (users must not copy other users' files) to very specific (user A must not copy user B's files)
- Different levels of accuracy from general statements in English to precise mathematical formalisms

Formal statements are more precise when they are formulated correctly

When multiple organizations collaborate the composed entity often has a security policy based on the individual security policies

- This raises the problem of policy composition which is inherently difficult

Goals of Security Mechanisms

Security mechanisms are put in place to *prevent* attacks, *detect* attacks and *recover* from attacks

Prevention

- Cryptography and access control are often used to prevent attacks

Detection

- Intrusion detection systems are often used to detect attacks during or after the attack

Recovery

- React to the attack
  - *Common reactions are to stop the attack in progress or allow it to continue with extensive logging (to allow the attacker to be traced)*
- Repair the damage caused by the attack
  - *Identify the vulnerability, identify appropriate prevention mechanism (e.g., patch the system), determine damage caused by the attack, repair damage caused by the attack (e.g., roll-back of data bases to before the attack)*

Assumptions and Trust

Security policies are always based on some assumptions about the behaviour of components and entities in the system

Two assumptions are generally made:

- Security policy unambiguously partitions the system state into *secure* and *nonsecure* states
- Security mechanism will guarantee that a system in the *secure* states will never become *nonsecure*
- If either assumption fails the system is insecure

A security mechanism can be characterised as:

- Secure: it does not allow the system to enter nonsecure states
- Precise: it allows the system to enter all secure states
- Broad: it allow the system to enter states that are nonsecure
- In practise, most security mechanisms are broad

Trust Assumptions

Trusting that mechanisms work requires several assumptions

- Each mechanism is designed to implement one or more elements of the security policy
- The union of security mechanisms implements all aspects of the security policy
- The mechanisms are implemented correctly
- The mechanisms are installed and administered correctly

So what is required to trust encrypted data from the network

- Encryption algorithm must be strong (design)
- Encryption algorithm must be implemented correctly (implementation)
- Encryption software must be installed correctly (operation)
- Cryptographic key must be secret (administration)

Assurance

Assurance attempts to quantify some of the assumptions about trust in the security mechanism

Assurance requires a specification of the behaviour of the system

- A system is said to *satisfy* a specification if the specification correctly states how the system will function

Assurance considers all aspects of software development

- *Specification* of the system must be correct and unambiguous
- *Design* of the system translates the specification into components that will implement them
- *Implementation* creates a system that satisfies the design

  A program is correct if its implementation performs as specified
- *Testing* verifies *a posteriori* if the implementation is correct

  NB! testing cannot prove correctness, only incorrectness

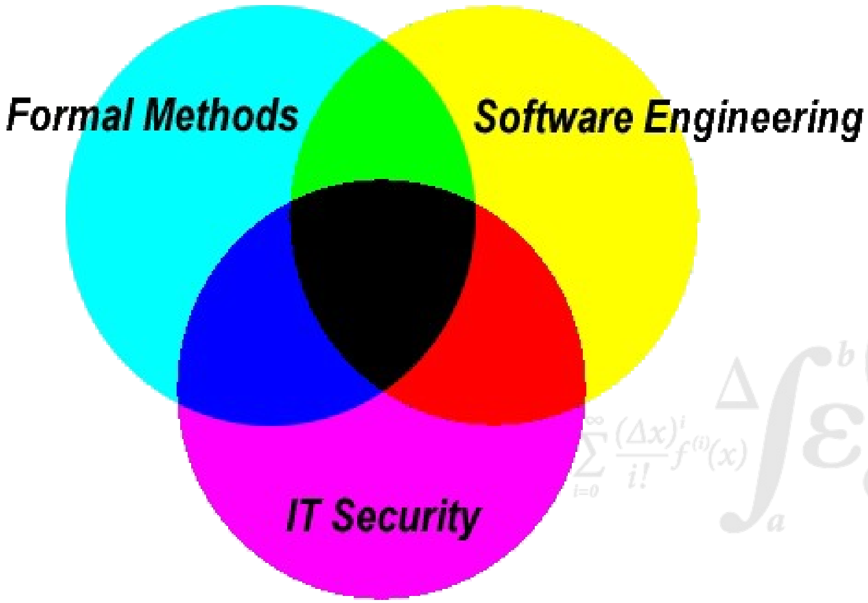
Stronger assurance requires formal proofs of correctness

Formal Methods for Security

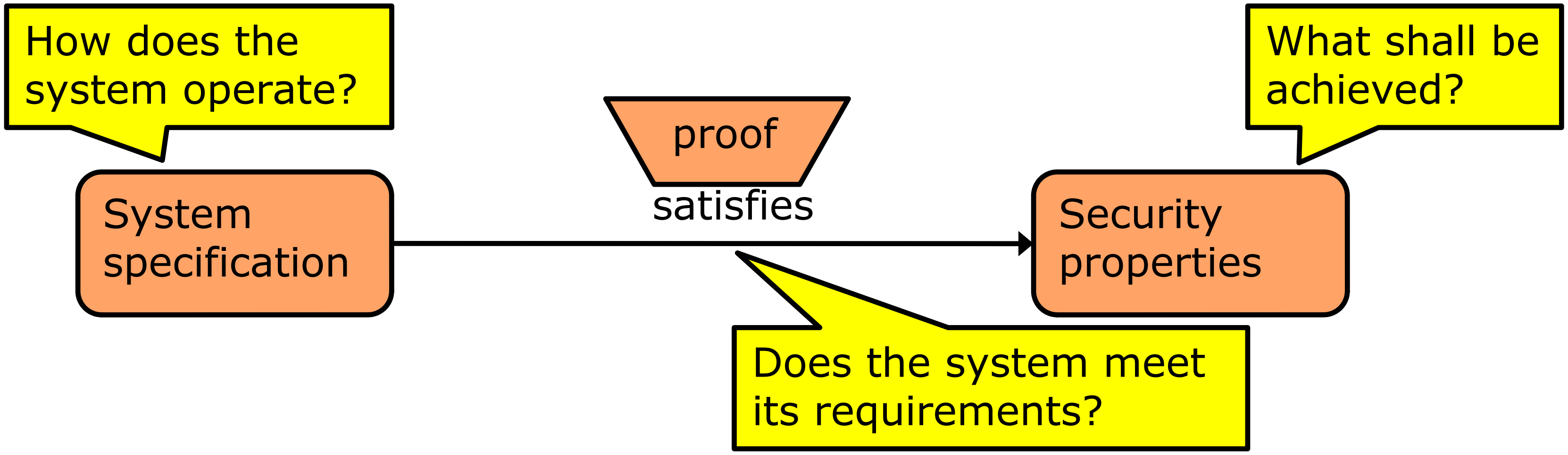
Systems can be understood as mathematical objects.

Formal methods based on mathematics and logic should be used to model, analyze, and construct them.

Doing so can substantially improve the security



Formal Methods for Security



- Even the mere attempt to formalize the security properties/goals of a system in a mathematically precise way can be revealing!
- SSE group focuses on automatic methods to find either a proof or a counter example – given a specification of the system and the security properties
- Tools based on model-checking, static analysis, abstract interpretation…
- Many attacks have been detected – and fixed -- using such tools:
- H.530, Google-Apps SSO, Kerberos PKInit

Organisational Issues

Implementing security in a large organisation is difficult and technology is often the easy bit (although we often get it wrong)

- Benefits of security are not directly visible *on the bottom line*
- Some security mechanisms may actually be costly
  - *Protocol overhead slows down communications*
  - *Requiring passwords to be entered frequently tends to lower productivity*

Responsibility for computer security is not always obvious

- IT department has responsibility for servers, hardware and software
- Individual users are responsible for choosing good passwords (and keeping them secret)
- If an attack exploits a weak password then who is to blame?

Social Engineering (e.g. phishing) is surprisingly effective

Security incidents are not always attacks

- An employee who reads and sends private emails during work hours is generally breaching policy, but the breach is normally accepted

Human Issues

Social Engineering is surprisingly effective

- Main approach used by Kevin Mitnick (FBI Most Wanted for many years)

Security Usability

Security mechanisms must be comprehensible and acceptable


Time to take a break