UNIVERSITY OF TWENTE.

Security in software development

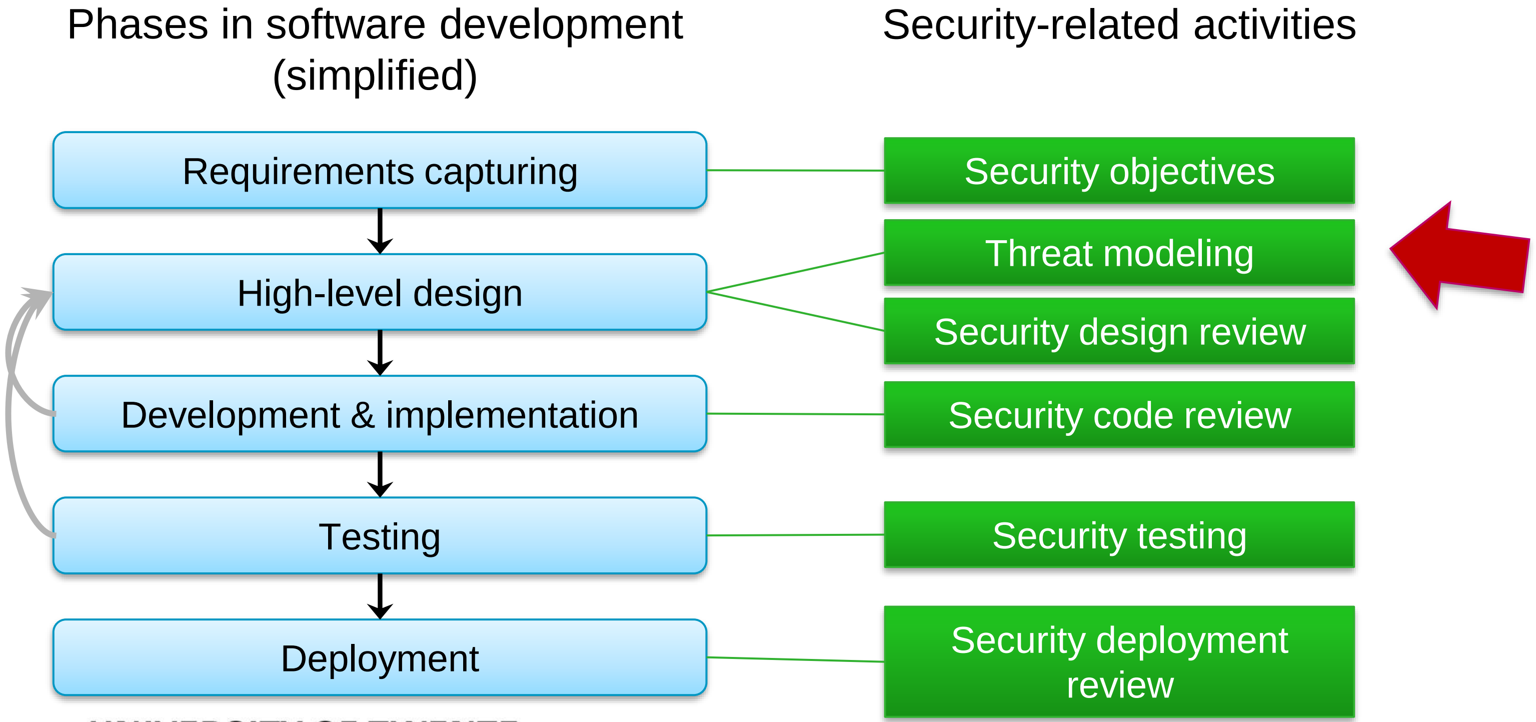
Topic of Software Systems (TCS module 2)

Lecturer: Maarten Everts

TERMINOLOGY IN (SOFTWARE) SECURITY

- Threat: potential violation of security
- Vulnerability: “Security-relevant software defect that can be exploited to effect an undesired behavior”
  - Flaw: defect in design
  - Bug: defect in the implementation
- Exploit

SECURITY IN THE DEVELOPMENT PROCESS



THREAT MODELING (SIMPLIFIED)

Semi-structured approach to identify, quantify and address security risks in an application.

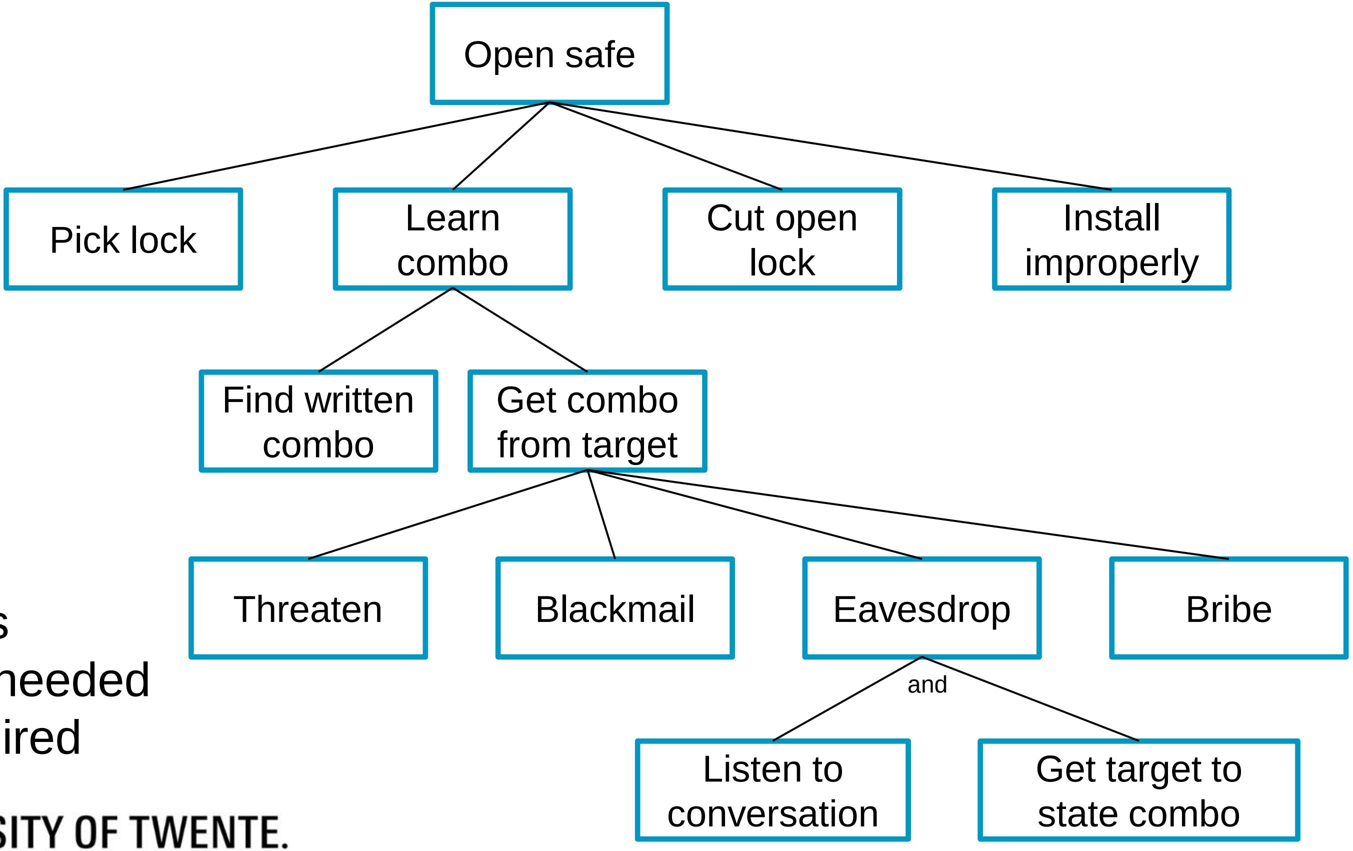
High-level steps:

1. Understanding the application (the design)
2. Identifying & categorizing threats
   - Identify entry/exit points
   - Identify assets
   - Identify trust levels
3. Countermeasures & mitigation

CATEGORIZING THREATS (STRIDE)

- Spoofing: posing as something or somebody else (e.g., replay attacks, phishing attacks)
- Tampering: malicious modification of data or code
- Repudiation: participating in a transaction or communication, and later claiming that the transaction or communication never took place.
- Information Disclosure: exposure or leakage of information
- Denial of Service: render a service or resource useless
- Elevation of Privilege: gaining increased capability

THREAT/ATTACK TREES

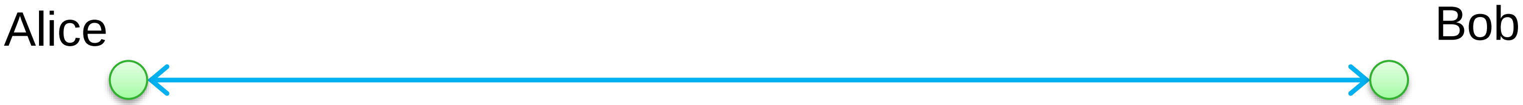


Extend with:

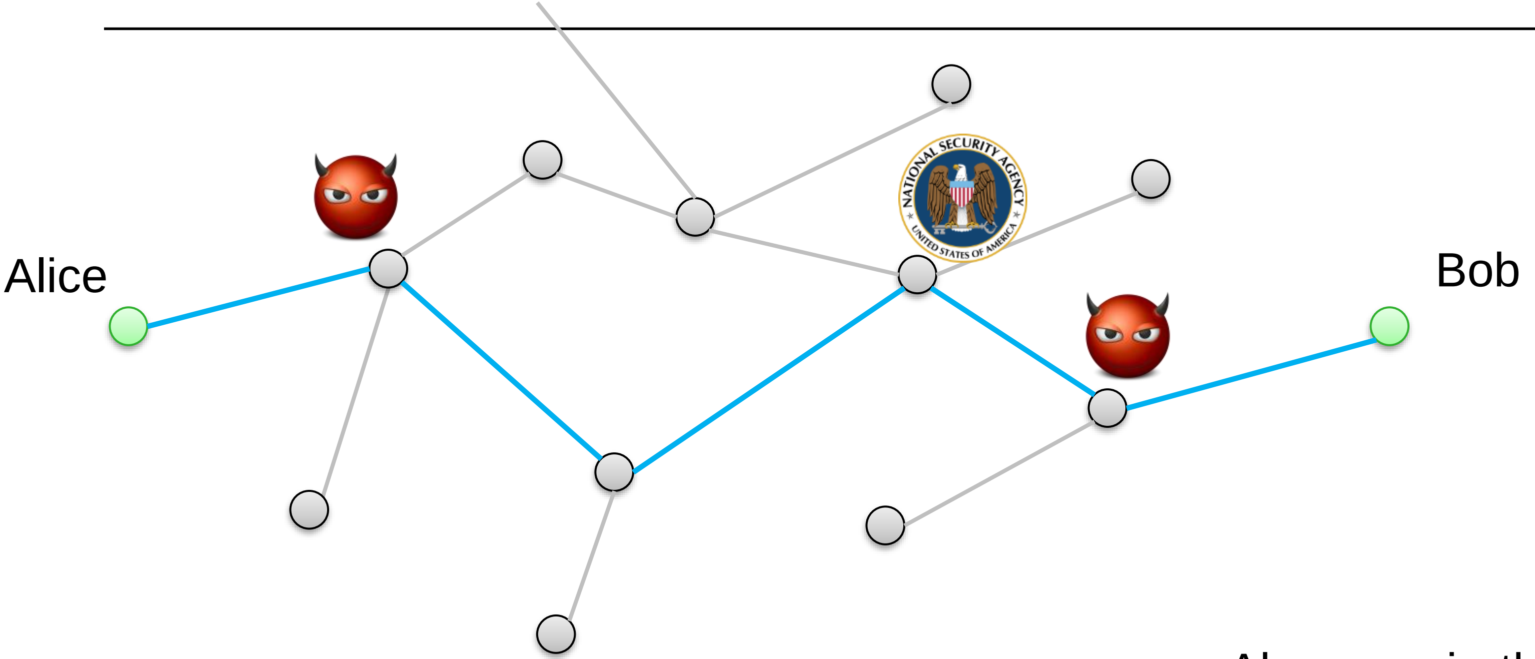
- Probabilities
- Equipment needed
- Money required

From: https://www.schneier.com/attacktrees.pdf

ATTACKER MODELS – FEATURING: ALICE & BOB



ATTACKER MODELS – FEATURING: ALICE & BOB



Aka, man-in-the-middle

Passive attacker: only listens

Active attacker: listens & modifies!

MITIGATION

- Implementation of security features:
  - Cryptography
  - Authorization (access control)
  - Authentication

No magic bullet!
Easy to make mistakes!

- Prevention (of bugs)
  - Testing!
  - Formal specifications (e.g., JML, langsec)
  - Defensive programming
- Detection, Audits
- Recovery & response

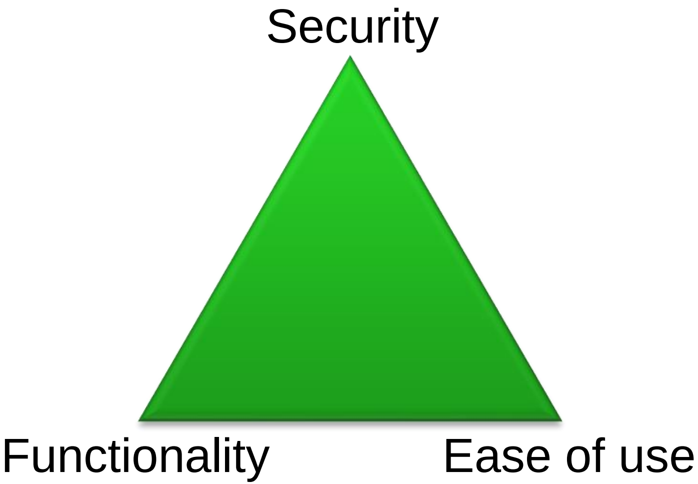
SOME SECURITY DESIGN PRINCIPLES

- Favor simplicity
  - Use fail-safe defaults
  - Do not expect expert users
- Trust with reluctance
  - Employ a small trusted computing base
  - Grant the least privilege possible
    - Promote privacy
    - Compartmentalize
- Defend in Depth
- Monitor and trace

BALANCING SECURITY

Security vs.

- cost
- performance
- usability
- acceptance



Security

Functionality

Ease of use

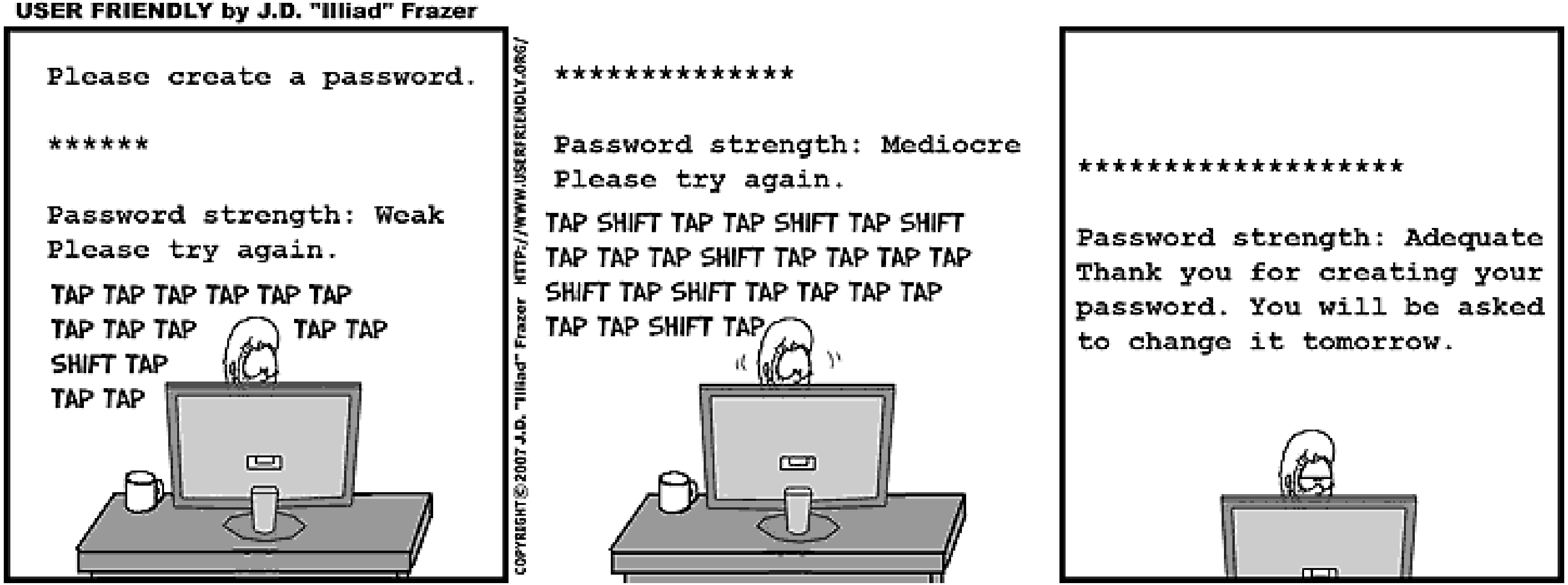
From: http://blog.infosanity.co.uk/2010/06/12/infosec-triads-securityfunctionalityease-of-use/

BALANCING SECURITY

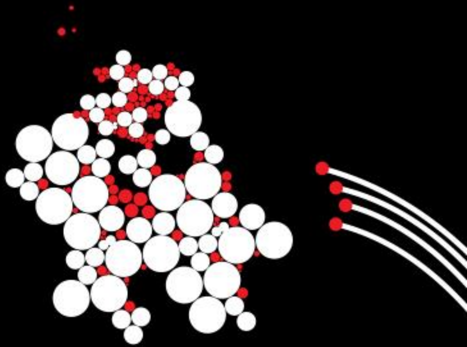
MORDAC, THE PREVENTER OF INFORMATION SERVICES.
SECURITY IS MORE IMPORTANT THAN USABILITY.
scottadams@aol.com
www.dilbert.com
IN A PERFECT WORLD, NO ONE WOULD BE ABLE TO USE ANYTHING.
© 2007 Scott Adams, Inc./Dist. by UFS, Inc.
To complete the log-in procedure, stare directly at the sun.

UNIVERSITY OF TWENTE.

BALANCING SECURITY


USER FRIENDLY by J.D. "Illiad" Frazer
Please create a password.
******
Password strength: Weak
Please try again.
TAP TAP TAP TAP TAP TAP
TAP TAP TAP
TAP TAP
SHIFT TAP
TAP TAP
COPYRIGHT © 2007 J.D. "Illiad" Frazer HTTP://WWW.USERFRIENDLY.ORG/
****
Password strength: Mediocre
Please try again.
TAP SHIFT TAP TAP SHIFT TAP SHIFT
TAP TAP TAP SHIFT TAP TAP TAP TAP
SHIFT TAP SHIFT TAP TAP TAP TAP
TAP TAP SHIFT TAP
****
Password strength: Adequate
Thank you for creating your
password. You will be asked
to change it tomorrow.

UNIVERSITY OF TWENTE.



Security in software development

Topic of Software Systems (TCS module 2)

Lecturer: Maarten Everts