

Ch.27 Security Engineering







Why Security Engineering?

- Security is a perquisite to system integrity, availability, reliability, and safety
- Security provides the mechanism that enable a systems to protect its assets from attack
- Assets are system resources (information, files, programs, storage, processor capacity) that have value to its stakeholders
- Attacks take advantage of vulnerabilities that allow unauthorized system access
- It is difficult to make a system more secure by responding to bug reports, security must be designed in from the beginning testing appropriate?



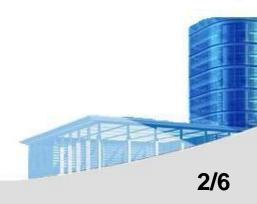
1/6



27.1 Analyzing Security Requirements

- Exposure is the value in terms of the time or cost to recreate a lost system asset
- Threat analysis is the process of determining the conditions or threats that may damage system resources or make them inaccessible to unauthorized access
- Controls are created to avoid the attacks and to mitigate their damage







27.2 Online Security Threats

- Social Media networks often allow their users to develop applications that have access to personal details of their users
- Mobile Applications native apps running on mobile devices may have the same access resources as the device owner
- Cloud Computing brings additional confidentiality and trust issues into the security picture because it blurs the line between "trusted inside" and "untrusted outside"
- Internet of Things ability of everyday objects to communicate and report contextual information about its user and its environment







27.3 Security Engineering Analysis

Security requirements elicitation

- Determine how users need to interact with system resources
- Create abuser stories that describe system threats
- User treat modeling and risk analysis to determine the system security policies as part of the nonfunctional requirements
- Locate attack patterns that identify solutions to system security shortcomings

Security modeling

- Captures policy objectives, external interface requirements, software security requirements, rules
 of operation, description of security architecture
- Provides guidance during design, coding and review
- State models can help software engineers ensure that the series of state transitions allowed by the system start and end in a secure state
- Using formal security models may improve the trustworthiness of a system since correctness proofs may be used as part of the system security case

Measures design

- Security metrics should focus on system dependability, trustworthiness, and survivability
- Measures or asset value, threat likelihood

Correctness checks

- Software verification activities and security test cases must be traceable to system security cases
- Data collected during audits, inspections, and test cases are analyzed and summarized as a security case



27.4 Security Assurance

- Used to show that you have created a secure product that inspires confidence among end users and stakeholders
- Security Case Elements
 - Security claims
 - Arguments linking claims to each other
 - Evidence (reviews, proofs, etc.) supporting arguments

27.5 Security Risk Analysis

- Steps to create a threat model by Microsoft.
 - Identify assets
 - Create architecture overview
 - Application decomposition
 - Identify threats
 - Document threats
 - Rate threats







27.7 System Trustworthiness

- Trust is the level of confidence that software components and stakeholders can rely on one another
- Verification ensures that the security requirements are assessed using objective and quantifiable techniques traceable to the security cases
- Evidence used to prove the security case must be acceptable and convincing to all system stakeholders
- Most trust metrics are based on historical data derived from past behavior in situations involving trust



