



Chapter 1: Introduction

- Components of computer security
- Threats
- Policies and mechanisms
- The role of trust
- Assurance
- Operational Issues
- Human Issues

Basic Components

- Confidentiality
 - Keeping data and resources hidden
- Integrity
 - Data integrity (integrity)
 - Origin integrity (authentication)
 - Mechanisms: Prevention and Detection
- Availability
 - Enabling access to data and resources

Classes of Threats

- **Threat**
 - Potential violation of security through **attacks**
- **Disclosure**
 - Unauthorized access to information
 - Snooping (passive wiretapping)
- **Deception**
 - Acceptance of false data
 - Modification, spoofing, repudiation of origin, denial of receipt

Classes of Threats

- **Disruption**

- Interruption or prevention of correct operation
- Modification

- **Usurpation**

- Unauthorized control of some part of a system
- Modification, spoofing, delay, denial of service

Policies and Mechanisms

- Policy says what is, and is not, allowed
 - This defines “security” for the site/system/*etc.*
- Mechanisms enforce policies
- Composition of policies
 - If policies conflict, discrepancies may create security vulnerabilities

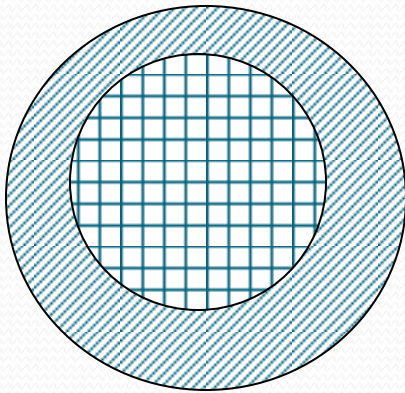
Goals of Security

- A policy defines “secure” and “non-secure” actions and mechanisms aim for the following:
- **Prevention**
 - Prevent attackers from violating security policy
 - Cumbersome, reduce flexibility
- **Detection**
 - Detect attackers’ violation of security policy
- **Recovery**
 - Stop attack, assess and repair damage
 - Continue to function correctly even if attack succeeds

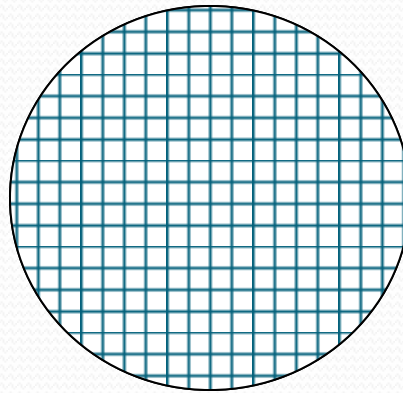
Trust and Assumptions

- Underlie *all* aspects of security
- Policies
 - Unambiguously partition system states (secure, not secure)
 - Correctly capture security requirements
- Mechanisms
 - Assumed to enforce policy
 - Support mechanisms work correctly

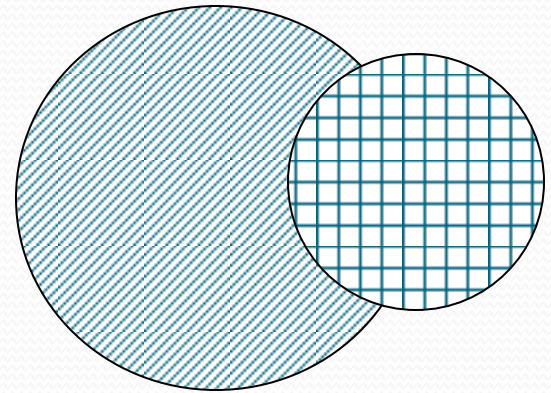
Types of Mechanisms



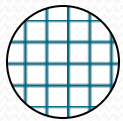
secure



precise



broad



set of reachable states



set of secure states

Assurance

- Measure of how well the system meets its requirements; i.e. how much you can trust the system to do what it is supposed to do.
- NIST Computer Security Handbook definition
 - “degree of confidence one has that the security measures, both technical and operational, work as intended to protect the system and the information it processes”
 - “Does the security system design meet its requirements?”
 - “Does the security system implementation meet its specifications”

Assurance

- Specification
 - Requirements analysis
 - Statement of desired functionality
- Design
 - How system will meet specification
- Implementation
 - Programs/systems that carry out design
 - Proof of correctness vs. testing

Operational Issues

- Cost-Benefit Analysis
 - Is it cheaper to prevent or recover?
 - Overlap of mechanism's effects
 - Will it be possible to enforce
 - Ease of use
- Risk Analysis
 - Should we protect something?
 - How much should we protect this thing?
 - What would happen if the data/resource is compromised?
 - What is the likelihood that the threats will materialize?
 - The level of protection is a function of the likelihood and the effect of the attack.

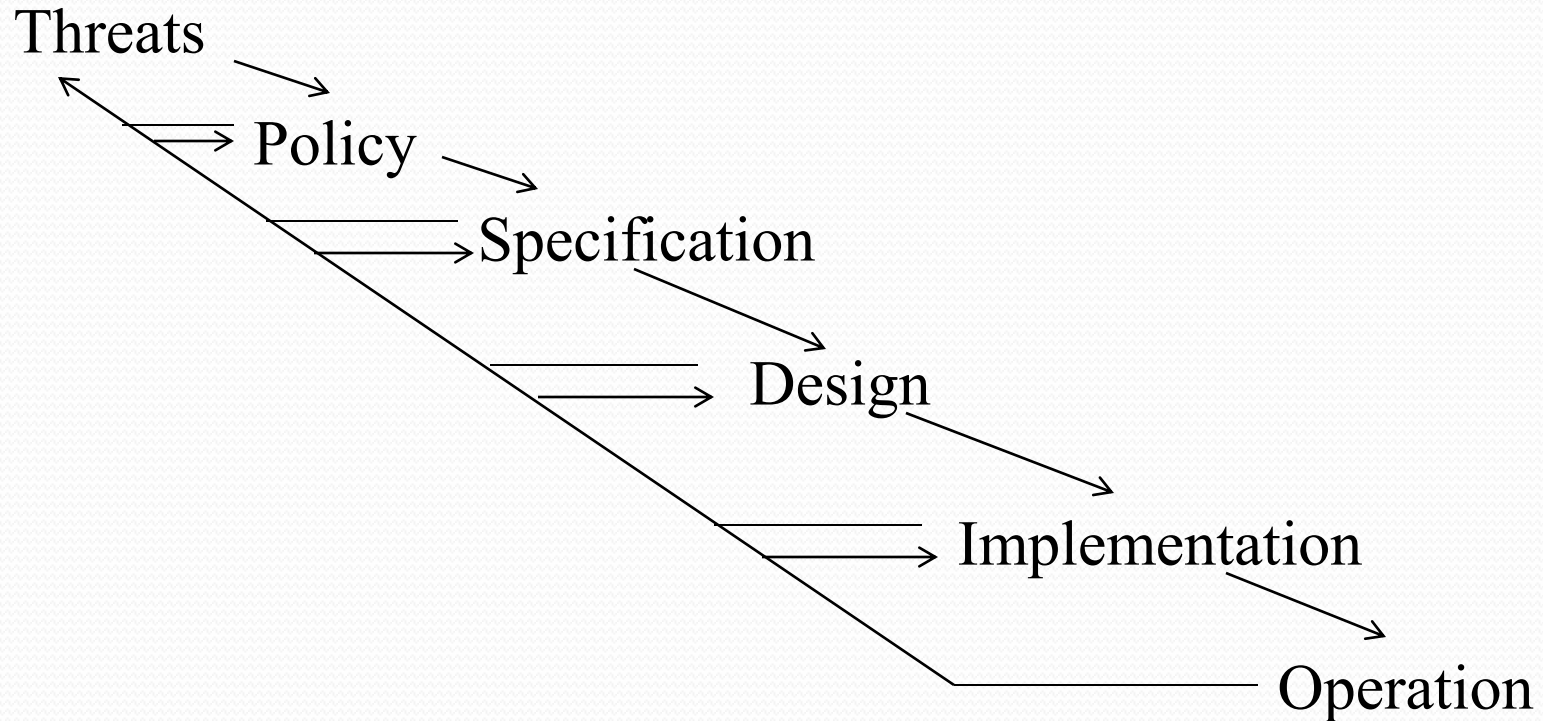
Operational Issues

- Risk Analysis
 - The amount of risk is a function of the environment
 - Risks change with time
 - Many risks are remote but exist
 - Problem of “analysis paralysis”
- Laws and Customs
 - Are desired security measures illegal?
 - Will people do them?

Human Issues

- Organizational Problems
 - No direct financial benefit
 - Requires financial support, resources, manpower
 - Power and responsibility
 - Trained dedicated personnel
- People problems
 - Outsiders and insiders
 - Social engineering

Tying Together



Key Points

- Policy defines security, and mechanisms enforce security
 - Confidentiality
 - Integrity
 - Availability
- Trust and knowing assumptions
- Importance of assurance
- The human factor

Security Policy

- Policy partitions system states into:
 - **Authorized (secure)**
 - These are states the system can enter
 - **Unauthorized (nonsecure)**
 - If the system enters any of these states, it's a security violation
- **Secure system**
 - Starts in authorized state
 - Never enters unauthorized state
- **Breach of security**
 - Occurs when a system enters an unauthorized state

Confidentiality

- X set of entities, I information
- I has *confidentiality* property with respect to X if no $x \in X$ can obtain information from I
- I can be disclosed to others
- Example:
 - X set of students
 - I final exam answer key
 - I is confidential with respect to X if students cannot obtain final exam answer key

Integrity

- X set of entities, I information
- I has *integrity* property with respect to X if all $x \in X$ trust information in I
- Types of integrity:
 - trust I , its conveyance and protection (**data integrity**)
 - I information about origin of something or an identity (**origin integrity, authentication**)
 - I resource: means resource functions as it should (**assurance**)

Availability

- X set of entities, I resource
- I has *availability* property with respect to X if all $x \in X$ can access I
- Types of availability:
 - traditional: x gets access or not
 - quality of service: promised a level of access (for example, a specific level of bandwidth) and not meet it, even though some access is achieved

Policy versus mechanism

- A security mechanism is an entity or procedure that enforces some part of the security policy.
- e.g. a corporation's policy indicates that all transactions related to a specific product must be completely confidential.
 - Only Alice and Jenny are allowed to perform the transactions on behalf of the clients
 - The data is tape backed-up everyday and the tapes are kept secure in an off-site location.

Policy Models

- Abstract description of a policy or class of policies
- Represents a particular policy or set of policies

Types of Security Policies

- Military (governmental) security policy
 - Policy primarily protecting confidentiality
- Commercial security policy
 - Policy primarily protecting integrity
- Confidentiality policy
 - Policy protecting only confidentiality
- Integrity policy
 - Policy protecting only integrity

Integrity and Transactions

- Begin in consistent state
 - “Consistent” defined by specification
- Perform series of actions (*transaction*)
 - Actions cannot be interrupted
 - If actions complete, system in consistent state
 - If actions do not complete, system reverts to beginning (consistent) state

Role of “trust”

- Confidentiality policies
 - No trust in the object itself (can the object be believed?)
 - Policy dictates whether or not the object can be disclosed
- Integrity policies
 - Indicate how much an object can be trusted.
 - How is the level of trust assigned?
 - e.g.: new version of software is obtained
 - High integrity (trust the vendor)
 - Low integrity (not tested on local system)
 - Medium integrity (trust the vendor, but also test on local system)
 - Integrity policies more dependent on the “trust” factor.



Trust

- When one understands the assumptions the security policies, mechanisms, and procedures rest on, then one gets a good understanding how effective those policies, mechanisms, and procedures are.

Trust in Formal Verification

- Gives formal mathematical proof that given input i , program P produces output o as specified
- Suppose a security-related program S formally verified to work with operating system O
- What are the assumptions?

Types of Access Control

- Discretionary Access Control (DAC)
 - individual user sets access control mechanism to allow or deny access to an object
 - Identity-based access control (IBAC)
- Mandatory Access Control (MAC)
 - system mechanism controls access to object, and individual cannot alter that access
 - Rule-based access control
- Originator Controlled Access Control (ORCON)
 - originator (creator) of information controls who can access information