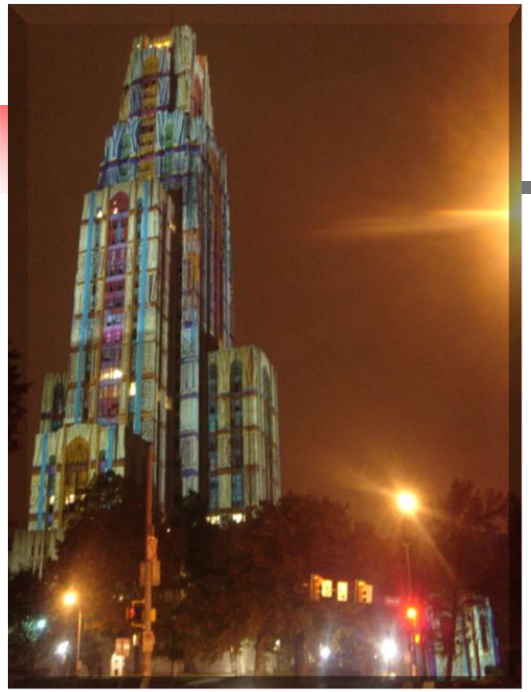
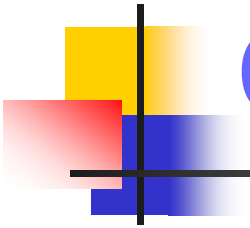
IS 2150 / TEL 2810
Information Security and Privacy



James Joshi
Associate Professor, SIS

Lecture 1
Jan 16, 2013



Contact

- Instructor: James B. D. Joshi
  - 706A, IS Building
  - Phone: 412-624-9982
  - E-mail: jjoshi@pitt.edu
  - Web: http://www.sis.pitt.edu/~jjoshi/
  - Office Hours:
    - By appointments
- GSA:
  - TBD

Course Goals

- to develop a broader understanding of the information security field,
  - *Recognize*, *analyze* and *evaluate* security problems and challenges in networks and systems.
  - *Apply* their knowledge to synthesize possible approaches to *solve* the problems in an integrated way.

Recognize the various security issues/terminologies related to software, networks and applications to *show* how they are interrelated and available techniques and approaches to solve/tackle security problems.

Analyze and *evaluate* the fundamentals of security policy models and mechanisms, and their need for different types of information systems and applications

Apply the basics of Cryptographic techniques and network security for ensuring the basic security goals of security of information systems.

Describe/identify the various basic social, legal and non-technical dimensions of security and its relation to technical counterparts.

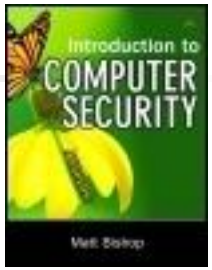
Certified for IA Standards

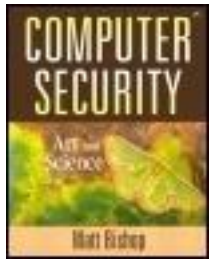
- SAIS Track is certified for 5 CNSS standards
  - 85% of content address the requirements of the first three CNSS standards
  - Hence CORE course for SAIS track
- Course webpage:
  http://www.sis.pitt.edu/~jjoshi/courses/IS2150/Fall10/

Course Outline

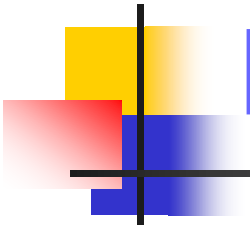
- Security 7 Privacy Basics
  - General overview and definitions
  - Security models and policy issues
  - Privacy
- Basic Cryptography and Network security
  - Crypto systems, digital signature, authentication, PKI
  - IPSec, VPN, Firewalls
- Systems Design Issues and Information assurance
  - Design principles; Security Mechanisms; Auditing Systems;
  - Risk analysis; System verification
- Intrusion Detection and Response
  - Attack Classification and Vulnerability Analysis
  - Detection, Containment and Response/Recovery
- Legal, Ethical, Social Issues
- Evaluation, Certification Standards
- Miscellaneous Issues
  - Malicious code
  - Security in cloud, social networks, BigData

Course Material



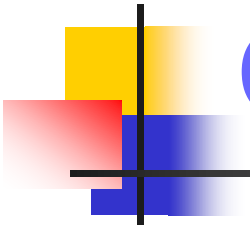


- Textbook
  - Introduction to Computer Security, Matt Bishop,
    - Errata URL: http://nob.cs.ucdavis.edu/~bishop/
  - Computer Security: Art and Science, Matt Bishop – is fine too
- Other Recommended
  - Security in Computing, Charles P. Pfleeger, Prentice Hall
  - Inside Java 2 Platform Security, 2nd Edition, L. Gong, G. Ellision, M. Dageforde
  - Security Engineering: A Guide to Building Dependable Distributed Systems, Ross Anderson, Wiley, John & Sons, Incorporated, 2001 (newer version)
  - Practical Unix and Internet Security, Simon Garfinkel and Gene Spafford
- Additional readings will be provided
  - Required or Optional



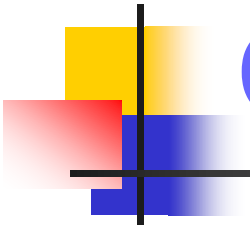
Prerequisites

- Assumes the following background
  - Programming skill
    - Some assignments in Java
  - Working knowledge of
    - Operating systems, algorithms and data structures, database systems, and networks
  - Basic Mathematics
    - Set, logic, induction techniques, data structure/algorithms
- Not sure? SEE ME



Grading

- Assignments (55%)
  - Homework/paper review: 35%
  - Labs and quizzes: 20%
- Programming project 15%
- Exams (30%) includes
  - Midterm: 15%
  - Final: 15%
- Other
  - Seminar (LERSAIS) and/or participation



Course Policies

- Your work MUST be your own
  - Zero tolerance for cheating/plagiarism
  - You get an F for the course if you cheat in anything however small – NO DISCUSSION
  - Discussing the problem is encouraged
- Homework
  - Penalty for late assignments (15% each day)
    - Seek extension under pressing circumstances
  - Ensure clarity in your answers – no credit will be given for vague answers
  - Sample solutions will be provided
- Check webpage for everything!
  - You are responsible for checking the webpage for updates

LERSAIS

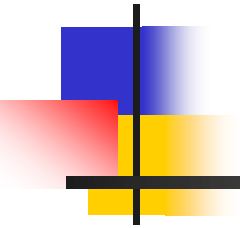

Laboratory of Education and Research on
Security Assured Information Systems
LERSAIS
University of Pittsburgh
School of Information Science
NSF
National Security Agency
United States of America
CNSS

LERSAIS

- Laboratory of Education and Research in Security Assured Information Systems
  - Established in 2003
  - National Center of Academic Excellence in Information Assurance Education - Research Program
    - A US National Security Agency program initiated in 1998 through a presidential directive to SECURE the Cyberspace
    - Partnered by Department of Homeland Security since 2003
  - LERSAIS is Pitt's representative center
- Website: http://www.sis.pitt.edu/~lersais/
  - Check out for Friday Seminars:

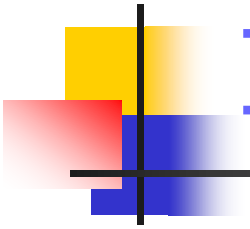
A Word on SAIS Track

- Pitt's IA curriculum has been certified for
  - Committee on National Security Systems IA Standards
    - CNSS 4011: Information Security Professionals
    - CNSS 4012: Designated Approving Authority
    - CNSS 4013: System Administrator in Information Systems Security
    - CNSS 4014: Information Systems Security Officer
    - CNSS 4015: System Certifiers
- Pitt is one among few Institutions in the US and one of two in the State of Pennsylvania to have five certifications
- One of the first group of schools to be designated as CAE-Research



What is Information Security?

Overview of Computer Security



Information Systems Security

- Deals with
  - Security of (end) systems
    - Examples: Operating system, files in a host, records, databases, accounting information, logs, etc.
  - Security of information in transit over a network
    - Examples: e-commerce transactions, online banking, confidential e-mails, file transfers, record transfers, authorization messages, etc.

"Using encryption on the internet is the equivalent of arranging an armored car to deliver credit card information from someone living in a cardboard box to someone living on a park bench" –

Gene Spafford

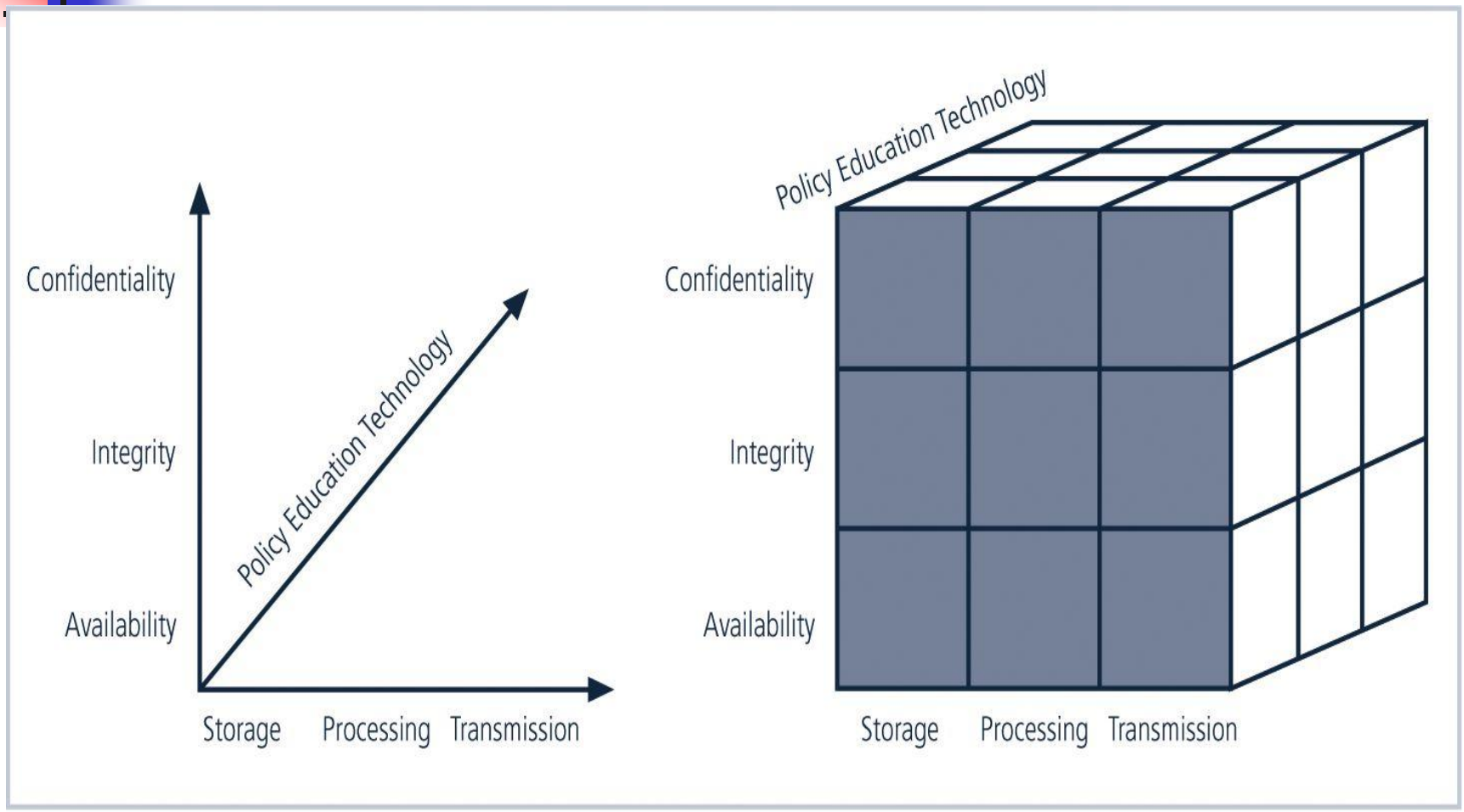
Basic Components of Security

- Confidentiality
  - Keeping data and resources secret or hidden
  - Conceal existence of data
- Integrity
  - Refers to correctness and trustworthiness
  - Ensuring authorized modifications;
  - May refer to
    - Data integrity
    - Origin integrity (Authentication)
- Availability
  - Ensuring authorized access to data and resources when *desired*
    - *Often assume a statistical model for pattern of use – which can be distorted*

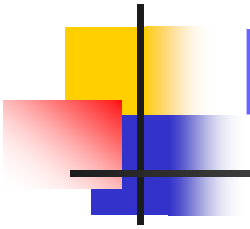
- Prevention
- Detection

Trust Management (Emerging Challenge)

CIA-based Model



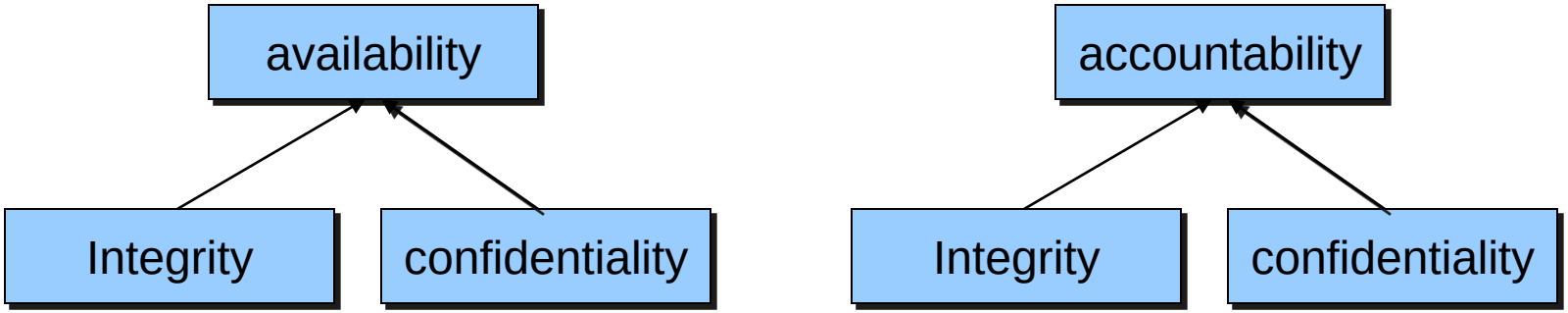
NSTISSC 4011 Security Model (CNSS 4011)

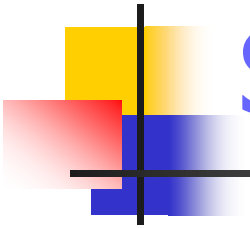


Basic Components of Security

- Additional from NIST (National Institute of Standards and Technology
  - Accountability
    - Ensuring that an entity's action is traceable uniquely to that entity
  - [Security] assurance
    - Assurance that all four objectives are met
- Other
  - Non-repudiation:
    - false denial of an act

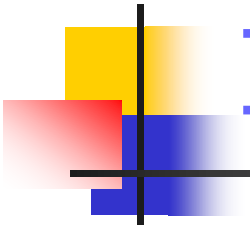
Interdependencies





Security - Years back

- Physical security
  - Information was primarily on paper
  - Lock and key
  - Safe transmission
- Administrative security
  - Control access to materials
  - Personnel screening
  - Auditing

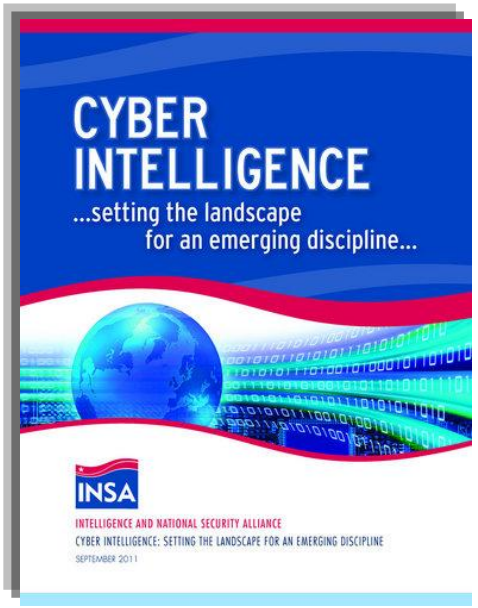


Information security today

- Emergence of the Internet and distributed systems
  - Increasing system complexity
  - Open environment with previously unknown entities interacting
- Digital information needs to be kept secure
  - Competitive advantage
  - Protection of assets
  - Liability and responsibility

Information security today

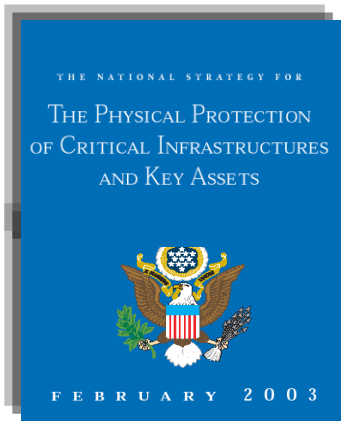
- Financial losses (FEW YEARS back)
  - The FBI estimates that an insider attack results in an average loss of $2.8 million
  - Reports indicate annual financial loss due to information security breaches of $5 - 45 billion
- More recent
  - Sony's estimiate: $170M from hacks on Playstation network (77 M accounts compromized)
  - Citibank – 360K bank card users

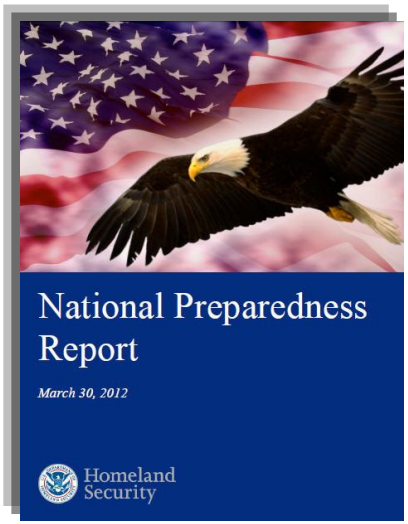


The cost of global cybercrime, at $114 billion annually, is significantly more than the annual global market for marijuana, cocaine and heroin combined

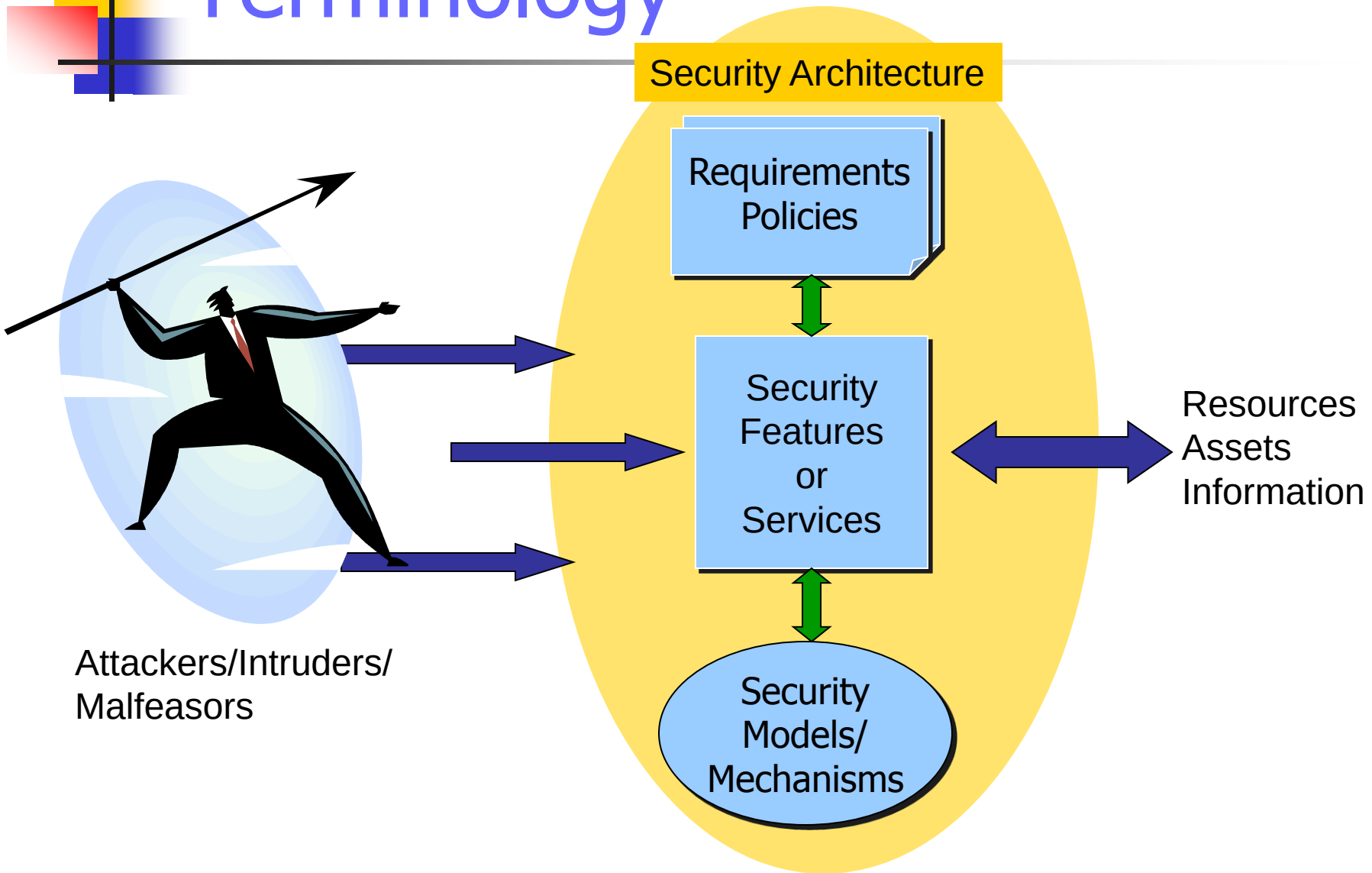
Information security today

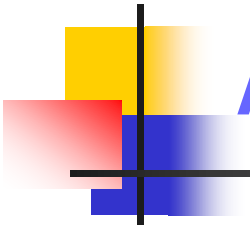
- National defense
  - Cybersecurity "was the single core capability where states had made the least amount of overall progress,"
  - U.S. Computer Emergency Readiness Team (US-CERT) reported an over 650-percent increase in the number of cyber incidents reported by federal agencies over a 5 year period





Terminology

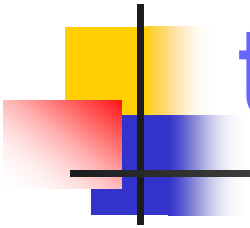




Attack Vs Threat

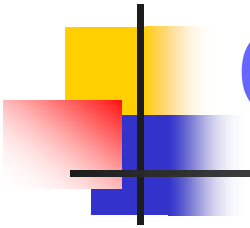
- A threat is a "potential" violation of security
  - The violation need not actually occur
  - The fact that the violation *might* occur makes it a threat
  - It is important to guard against threats and be prepared for the actual violation
- The actual violation of security is called an attack

Common security threats/attacks

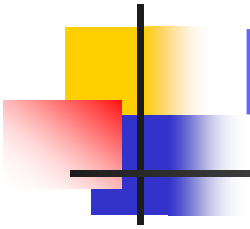


- Interruption, delay, denial of service
  - System assets or information become unavailable or are rendered unavailable
- Interception or snooping
  - Unauthorized party gains access to information by browsing through files or reading communications
- Modification or alteration
  - Unauthorized party changes information in transit or information stored for subsequent access
- Fabrication, masquerade, or spoofing
  - Spurious information is inserted into the system or network by making it appear as if it is from a legitimate entity
- Repudiation of origin
  - False denial that an entity did (send/create) something

Classes of Threats (Shirley)

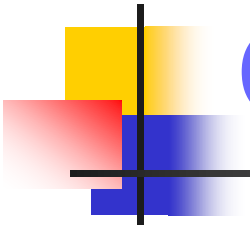


- Disclosure: *unauthorized access to information*
  - Snooping
- Deception: *acceptance of false data*
  - Modification, masquerading/spoofing, repudiation of origin, denial of receipt
- Disruption: *interruption/prevention of correct operation*
  - Modification
- Usurpation: *unauthorized control of a system component*
  - Modification, masquerading/spoofing, delay, denial of service



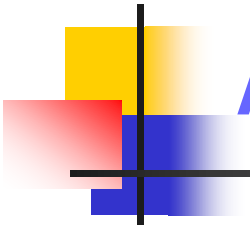
Policies and Mechanisms

- A security policy states what is, and is not, allowed
  - This defines "security" for the site/system/*etc*.
  - Policy definition: Informal? Formal?
- Mechanisms enforce policies
- Composition of policies
  - If policies conflict, discrepancies may create security vulnerabilities



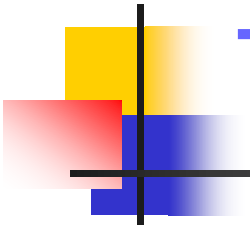
Goals of Security

- Prevention
  - To prevent someone from violating a security policy
- Detection
  - To detect activities in violation of a security policy
  - Verify the efficacy of the prevention mechanism
- (Response &) Recovery
  - Stop policy violations (attacks)
  - Assess and repair damage
  - Ensure availability in presence of an ongoing attack
  - Fix vulnerabilities for preventing future attack
  - Retaliation against the attacker



Assumptions and Trust

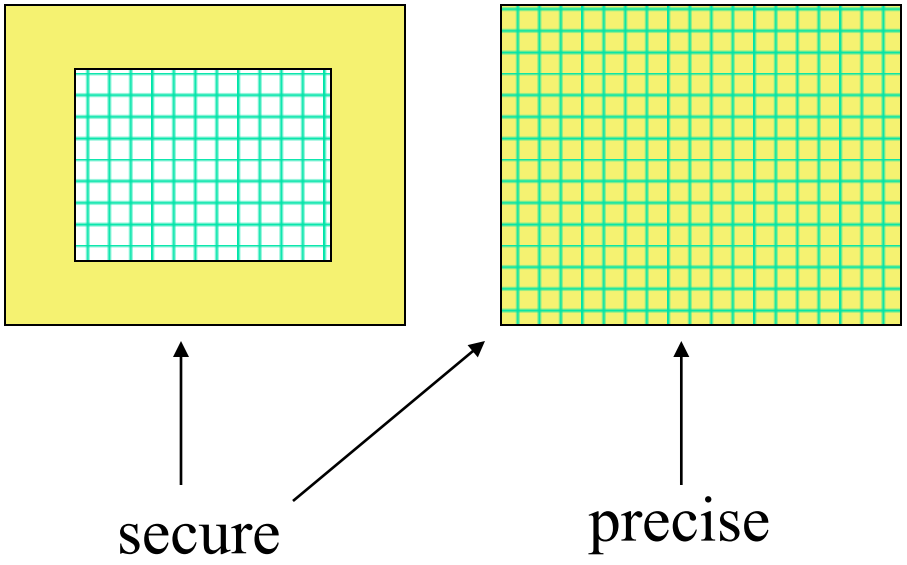
- Policies and mechanisms have implicit assumptions
- Assumptions regarding policies
  - Unambiguously partition system states into "secure" and "nonsecure" states
  - Correctly capture security requirements
- Mechanisms
  - Assumed to enforce policy; i.e., ensure that the system does not enter "nonsecure" state
  - Support mechanisms work correctly

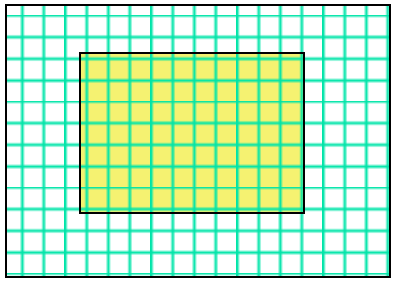


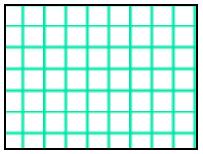
Types of Mechanisms

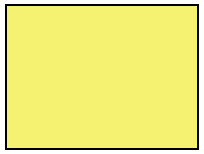
- Let P be the set of all the reachable states
- Let Q be a set of secure states identified by a policy: $Q \subseteq P$
- Let the set of states that an enforcement mechanism restricts a system be R
- The enforcement mechanism is
  - Secure if $R \subseteq Q$
  - Precise if $R = Q$
  - Broad if there are some states in R that are not in Q

Types of Mechanisms






set R


set Q (secure states)

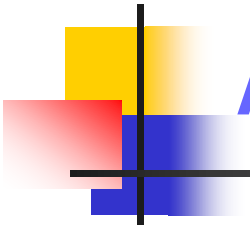
Information Assurance

- *Information Assurance Advisory Council (IAAC)*:

  "Operations undertaken to protect and defend information and information systems by ensuring their availability, integrity, authentication, confidentiality and non-repudiation"

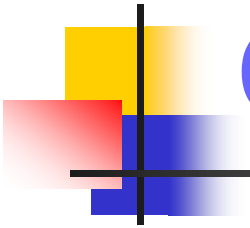
- National Institute of Standards Technology

  "Assurance is the basis for confidence that the security measures, both technical and operational, work as intended to protect the system and the information it processes"



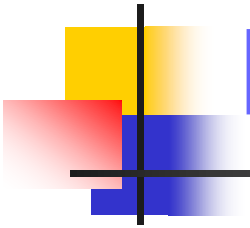
Assurance

- Assurance is to indicate "how much" to trust a system and is achieved by ensuring that
  - The required functionality is present and correctly implemented
  - There is sufficient protection against unintentional errors
  - There is sufficient resistance to intentional penetration or by-pass
- Basis for determining this aspect of trust
  - Specification
    - Requirements analysis
    - Statement of desired functionality
  - Design
    - Translate specification into components that satisfy the specification
  - Implementation
    - Programs/systems that satisfy a design



Operational Issues

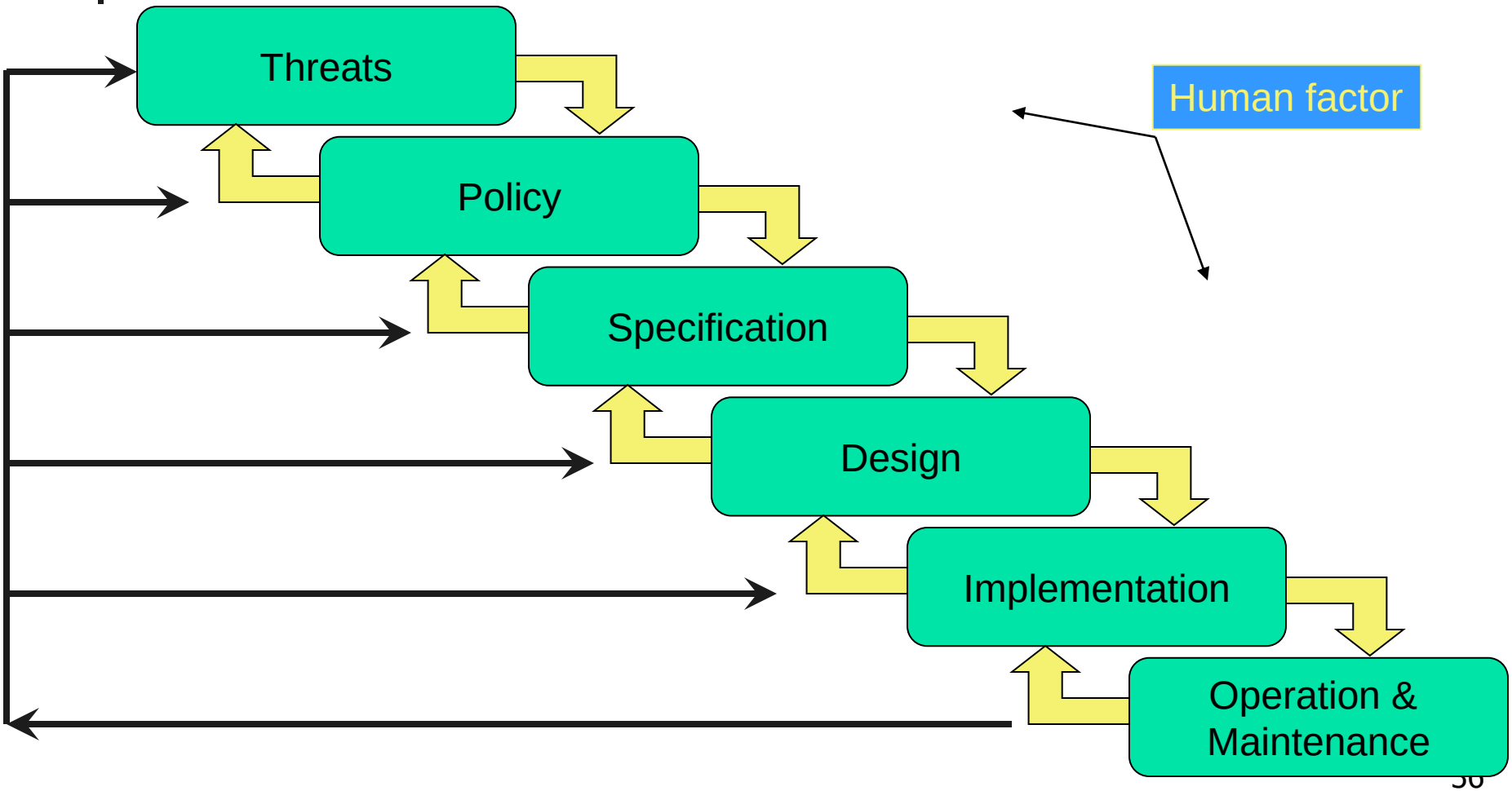
- Designing secure systems has operational issues
- Cost-Benefit Analysis
  - Benefits vs. total cost
  - Is it cheaper to prevent or recover?
- Risk Analysis
  - Should we protect something?
  - How much should we protect this thing?
  - Risk depends on environment and change with time
- Laws and Customs
  - Are desired security measures illegal?
  - Will people do them?
  - Affects availability and use of technology



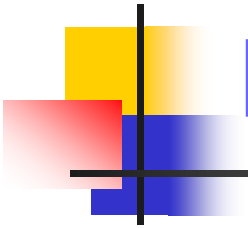
Human Issues

- Organizational Problems
  - Power and responsibility
  - Financial benefits
- People problems
  - Outsiders and insiders
    - *Which do you think is the real threat?*
  - Social engineering

Tying all together: The Life Cycle

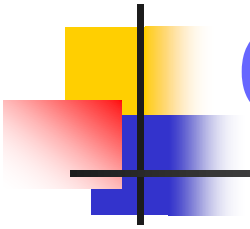


Design Principles



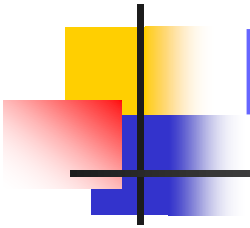
Design Principles for Security

- Principles
  - Least Privilege
  - Fail-Safe Defaults
  - Economy of Mechanism
  - Complete Mediation
  - Open Design
  - Separation of Privilege
  - Least Common Mechanism
  - Psychological Acceptability



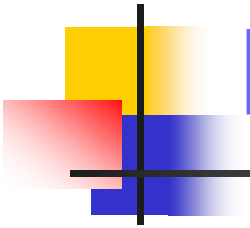
Overview

- Based on the idea of *simplicity* and *restriction*
  - *Why* Simplicity?
  - *Why* Restriction?



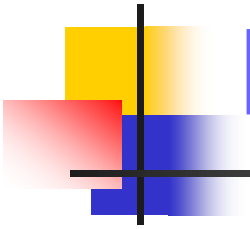
Least Privilege

- A subject should be given only those privileges necessary to complete its task
  - Assignment of privileges based on
    - Function OR Identity-based, … ?
  - Based on "Need to know"; "Relevance to situation" …
    - Examples?
  - Confine processes to "minimal protection domain"
  - How can it be enforced?
    - In Unix? Windows?
    - Challenge? [Complexity?]



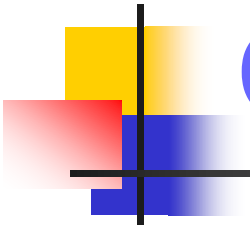
Fail-Safe Defaults

- What should be the default action?
- If action fails, how can we keep the system safe/secure?
  - Transactions based systems?
  - When a file is created, what privileges are assigned to it?
    - In Unix? In Windows?



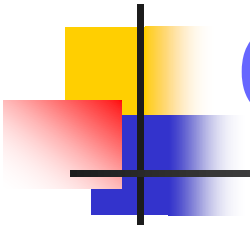
Economy of Mechanism

- Design and implementation of security mechanism
  - KISS Principle (Keep It Simple, Silly!)
- Simpler means?
- Careful design of Interfaces and Interactions



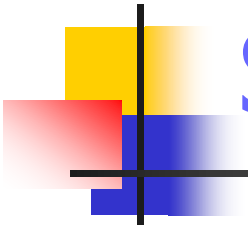
Complete Mediation

- No caching of information
- Mediate all accesses
  - Why?
  - How does Unix read operation work?
  - Any disadvantage of this principle?



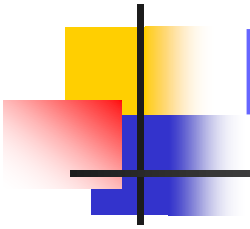
Open Design

- Security should not depend on secrecy of design or implementation
  - Source code should be public?
  - "Security through obscurity" ?
  - Does not apply to certain "information"
    - Secrecy of : keys vs encryption algorithm"?
  - What about the "Proprietary software"?



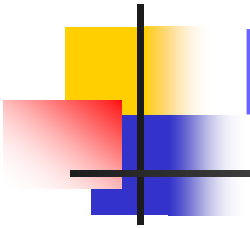
Separation of Privilege

- Restrictive access
  - Use multiple conditions to grant privilege
  - Equivalent to Separation of duty
    - Example?
  - Changing to root account in Berkley-based Unix … need two conditions!



Least Common Mechanism

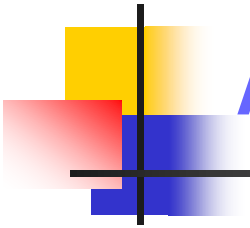
- Mechanisms should not be shared
  - What is the problem with shared resource?
    - Covert channels?
- Isolation techniques
  - Virtual machine
  - Sandbox



Psychological Acceptability

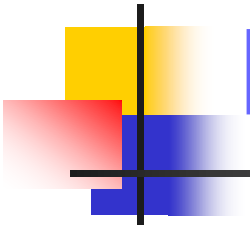
- Security mechanisms should not add to difficulty of accessing resource
  - Hide complexity introduced by security mechanisms
  - Ease of installation, configuration, use
  - Human factors critical here
    - Proper messages

Access Control - Introduction



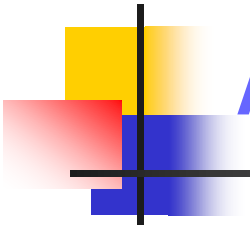
ACM Background

- Access Control Matrix
  - Captures the current protection state of a system
- Butler Lampson proposed the first Access Control Matrix model
- Refinements
  - By Graham and Denning
  - By Harrison, Russo and Ulman – with some theoretical results



Protection System

- Subject (S: set of all subjects)
  - Active entities that carry out an action/operation on other entities;
  - Examples?
- Object (O: set of all objects)
  - Examples?
- Right (R: set of all rights)
  - An action/operation that a subject is allowed/disallowed on objects
  - Access Matrix A: $a[s, o] \subseteq R$
- Set of Protection States: (S, O, A)



Access Control Matrix Model

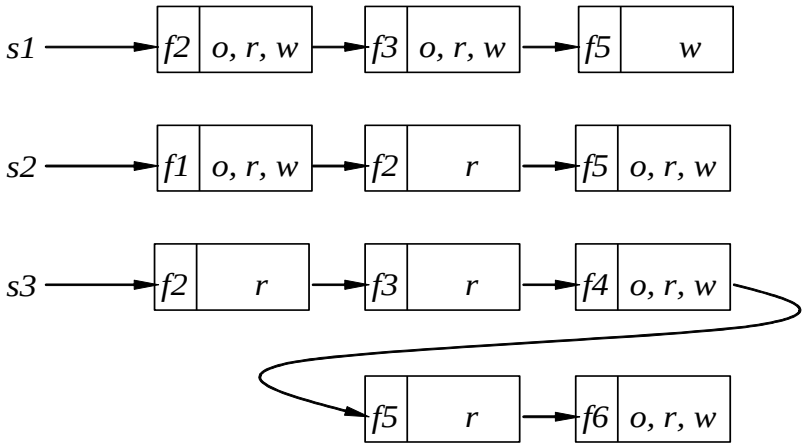
- Access control matrix model
  - Describes the protection state of a system.
  - Elements indicate the access rights that subjects have on objects
  - Is an abstract model - what does it mean?
- ACM implementation
  - What is the disadvantage of maintaining a matrix?
  - Two ways implement:
    - Capability based
    - Access control list

o: own
r: read
w:write

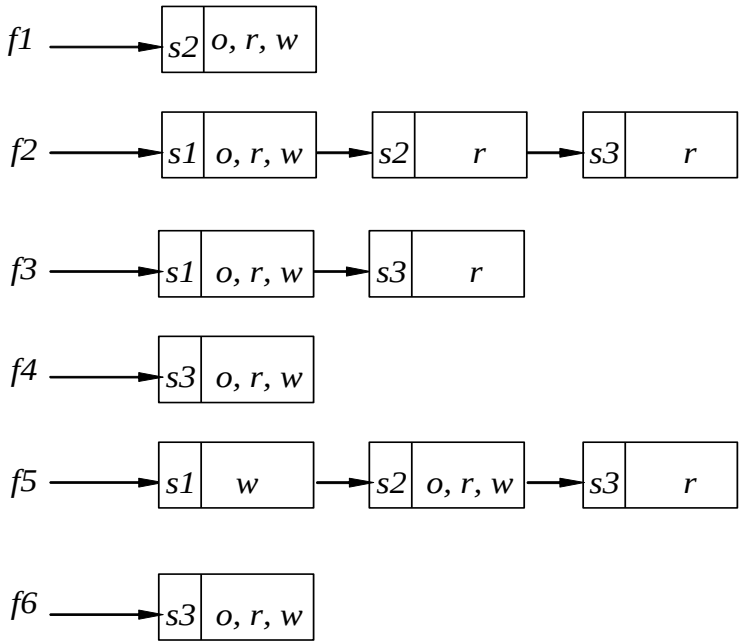
	f1	f2	f3	f4	f5	f6
s1		o, r, w	o, r, w		w	
s2	o, r, w	r			o, r, w	
s3		r	r	o, r, w	r	o, r, w

Access Matrix

Capabilities



Access Control List

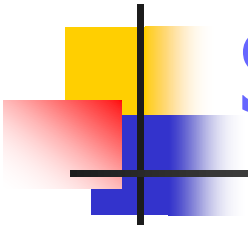


Access Control Matrix

Hostnames	*Telegraph*	*Nob*	*Toadflax*
Telegraph	*own*	*ftp*	*ftp*
Nob		*ftp, nsf, mail, own*	*ftp, nfs, mail*
Toadflax		*ftp, mail*	*ftp, nsf, mail, own*

- *telegraph* is a PC with ftp client but no server
- *nob* provides NFS but not to Toadfax
- *nob* and *toadfax* can exchange mail

	Counter	*Inc_ctr*	*Dcr_ctr*	*Manager*
Inc_ctr	+			
Dcr_ctr	-			
manager		*Call*	*Call*	*Call*



Summary

- Course outline
- Overview of security
  - Basic components:
    - CIA, Assurance
  - Policy/Mechanisms
  - Operational and human issues
- Key Secure Design Principles