

Computer Security

CS 426

Lecture 13



Multi-level Security

Announcements

- Project due on Thursday October 4th
- Mid-term exam on Thursday October 18th
- Collegiate Cyber Defense Competition
 - state (optional), regional, national
 - 2007 participants in Mid West: Indiana Tech, IN; Jackson Community College, MI; Baker College, MI; Madison Area Technical College, WI; DePaul University, IL

Review: Access Control Concepts

- Access Matrices
- Users, Principals, Subjects
- Objects, Rights
- Access Control Lists
- Capabilities
- Security Enhanced Linux
- SubDomain/AppArmor
- UMIP

Review: Discretionary Access Control

- No precise definition. Basically, DAC allows access rights to be propagated at subject's discretion
 - often has the notion of owner of an object
 - used in UNIX, Windows, etc.
- *"A means of restricting access to objects based on the identity and need-to-know of users and/or groups to which the object belongs. Controls are discretionary in the sense that a subject with a certain access permission is capable of passing that permission (directly or indirectly) to any other subject."*

Mandatory Access Control

- Mandatory access controls (MAC) restrict the access of subjects to objects based on a system-wide policy
 - denying users full control over the access to resources that they create. The system security policy (as set by the administrator) entirely determines the access rights granted

Bell-LaPadula Model: A MAC Model for Achieving Multi-level Security

- Introduce in 1973
- Air Force was concerned with security in time-sharing systems
 - Many OS bugs
 - Accidental misuse
- Main Objective:
 - Enable one to formally show that a computer system can securely process classified information

Basic Idea

- There are security classifications or security levels
 - Users/principals/subjects have security clearances
 - Objects have security classifications
- Example
 - Top Secret
 - Secret
 - Confidential
 - Unclassified
- In this case Top Secret > Secret > Confidential > Unclassified
- Security goal (confidentiality): ensures that information do not flow to those not cleared for that level

What is a Security Model?

- A model describes the system
 - e.g., a high level specification or an abstract machine description of what the system does
- A security policy
 - defines the security requirements for a given system
- Verification shows that a policy is satisfied by a system
- System Model + Security Policy = Security Model

Methodology in the BLP Security Model

- Define an abstract model that can be used to describe computer systems.
 - the model
- Define what does it mean for a system in the model to be secure.
 - the policy
- Develop techniques to prove that a system in the model is secure

Approach of BLP

- Use state-transition systems to describe computer systems
- Define a system as secure iff. every reachable state satisfies 3 properties
 - simple-security property, *-property, discretionary-security property
- Prove a Basic Security Theorem (BST)
 - so that one can prove a system is secure by proving things about the system description

The BLP Security Model Simplified

- A computer system is modeled as a state-transition system
 - In each state, there are subjects, objects, an access matrix, and the current access information
 - there are state transition rules describing how a system can go from one state to another
 - each subject is given a level, and each object is given a level

The BLP Security Model Simplified

- A state is secure if it satisfies
 - Simple Security Condition (no read up):
 - S can read O iff $L(S) = L(O)$
 - The Star Property (no write down)
 - S can write O iff $L(S) = L(O)$
 - discretionary-security property
 - every access is allowed by the access matrix
- A system is secure if and only if every reachable state is secure.

More Details in BLP

- Trusted subjects
 - some subjects are identified as trusted subjects, the star property does not apply to trusted subjects
 - why having trusted subjects?
- In the actual model, each subject has two levels: the maximum level and the current level
 - the simple security condition uses the maximum level
 - the *-property uses the current level

STAR-PROPERTY

- applies to subjects (principals) not to users
- users are trusted (must be trusted) not to disclose secret information outside of the computer system
- subjects are not trusted because they may have Trojan Horses embedded in the code they execute
- star-property prevents overt leakage of information and does not address the covert channel problem

Issues with BLP

- Deal only with confidentiality,
 - does not deal with integrity at all
- Does not deal with information flow through covert channels
- The approach of defining a secure system to be one in which every reachable state is secure is flawed
 - in a system that is secure according to BLP, a subject can read high, change current level to low, then write low.
 - to address this problem, need to require
 - subject cannot change current levels
 - or ensures a subject to “forgot” everything when changing levels
 - information flow security is not a per-state property

Overt (Explicit) Channels vs. Covert Channels

- Security objective of MLS in general, BLP in particular
 - high-classified information cannot flow to low-cleared users
- Overt channels of information flow
 - read/write an object
- Covert channels of information flow
 - communication channel based on the use of system resources not normally intended for communication between the subjects (processes) in the system

Examples of Covert Channels

- Using file lock as a shared boolean variable
- By varying its ratio of computing to input/output or its paging rate, the service can transmit information to a concurrently running process
- Covert channels are often noisy
- However, information theory and coding theory can be used to encode and decode information through noisy channels

More on Covert Channels

- Covert channels cannot be blocked by *-property
- It is generally very difficult, if not impossible, to block all cover channels
- One can try to limit the bandwidth of covert channels
- Military requires cryptographic components be implemented in hardware
 - to avoid trojan horse leaking keys through covert channels

More on MLS: Security Levels

- Used as attributes of both subjects & objects
 - clearance & classification
- Typical military security levels:
 - top secret \geq secret \geq confidential \geq unclassified
- Typical commercial security levels
 - restricted \geq proprietary \geq sensitive \geq public

Security Categories

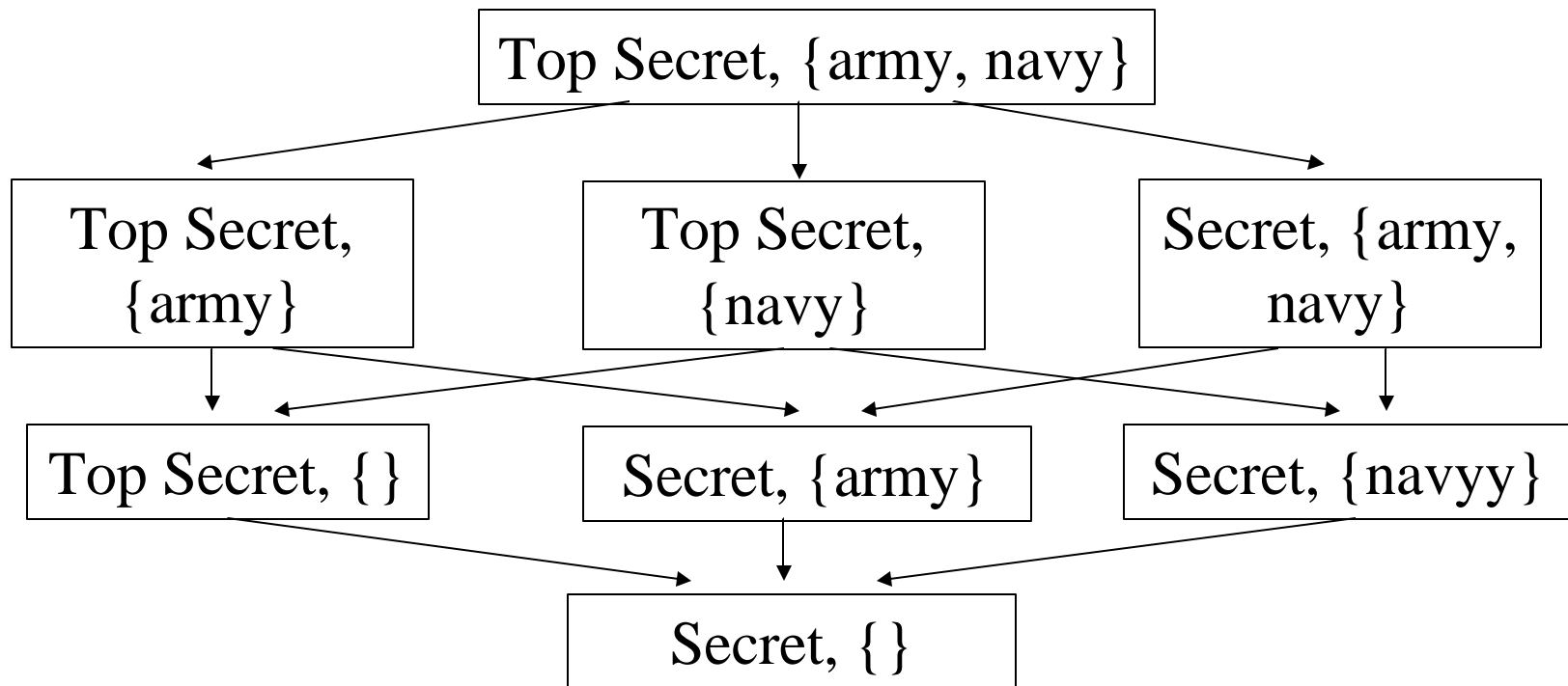
- Also known as compartments
- Typical military security categories
 - army, navy, air force
 - nato, nasa, nofor
- Typical commercial security categories
 - Sales, R&D, HR
 - Dept A, Dept B, Dept C

Security Labels

- Labels = Levels \times P (Categories)
- Define an ordering relationship among Labels
 - $(e1, C1) \leq (e2, C2)$ iff. $e1 \leq e2$ and $C1 \subseteq C2$
- This ordering relation is a partial order
 - reflexive, transitive, anti-symmetric
 - e.g., \subseteq
- All security labels form a lattice

An Example Security Lattice

- levels={top secret, secret}
- categories={army,navy}



The need-to-know principle

- Even if someone has all the necessary official approvals (such as a security clearance) to access certain information they should not be given access to such information unless they have a *need to know*: that is, unless access to the specific information necessary for the conduct of one's official duties.
- Can be implemented using categories and or DAC

Coming Attractions ...

- October 4:
 - Trusted Operating Systems & Assurance
- Readings for this lecture
 - Security Engineering: Chapter 7
- Readings for next lecture
 - Security Engineering: Chapter 23

