CSCE 465 Computer & Network Security

Instructor: Abner Mendoza

Security Theory II:
Security Policies and Models

Roadmap

- Security policy
  - What they cover
- Access control model: express security policy
  - Confidentiality Policy and BLP Model
  - Integrity Policy and Biba model
  - Hybrid Policy Model
    - Chinese Wall
    - RBAC

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
    - A 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

Example Question

- Policy disallows cheating
  - Includes copying homework, with or without permission
- CSE class has students do homework on computer
- Anne forgets to read-protect her homework file
- Bill copies it
- Who cheated?
  - Anne, Bill, or both?

Answer Part 1

- Bill cheated
  - Policy forbids copying homework assignment
  - Bill did it
  - System entered unauthorized state (Bill having a copy of Anne’s assignment)
- If not explicit in computer security policy, certainly implicit
  - Not credible that a unit of the university allows something that the university as a whole forbids, unless the unit explicitly says so

Answer Part 2

- Anne didn’t protect her homework
  - Not required by security policy
- She didn’t breach security
- If policy said students had to read-protect homework files, then Anne did breach security
  - She didn’t do this

Mechanisms

- Entity or procedure that enforces some part of the security policy
  - Access controls (like bits to prevent someone from reading a homework file)
  - E.g., Disallowing people from bringing flash drives into a computer facility to control what is placed on systems

Access Control

- An access control system determines what rights a particular entity (subjects) has for a set of objects
- It answers the questions like
  - Do you have the right to read /etc/passwd
  - Does Alice have the right to view the CSE website?
  - Do students have the right to share project data?
  - Does TA have the right to change your grades?

Access Control Policy

- An access control policy can be considered as a function:
  - P(S,O,R) -> { accept, deny }
  - Where, set S=subjects, O=objects, R=rights
- The policy is a lot of these tuples, whether explicitly represented that way or not
- Access control matrix (as we learned from last class) is the common way to represent policy

Access Control Administration

There are two central ways to specify a policy

- Discretionary Access Control (DAC) - object "owners" define policy
  - individual user sets access control mechanism to allow or deny access to an object
  - E.g., UNIX file system
    - RWX assigned by file owners
- Mandatory Access Control (MAC) - Environment enforces static policy
  - Environment (system mechanism) controls access to object, and individual cannot alter that access
  - E.g., process labeling
    - System assigns labels for processes, objects, and a dominance calculus is used to evaluate rights

Access Control Models

- What language should I use to express policy?
  - Access Control Model (Security model)
- A security model is a model that represents a particular policy or set of policies
  - Abstracts details relevant for analysis
- Focus on specific characteristics of policies
  - Some specialize in secrecy, e.g., Bell-LaPadula
  - Some specialize in integrity, e.g., Biba
  - Some focus on conflict of interest, e.g., Chinese Wall
  - Some focus on jobs, e.g., RBAC

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

Confidentiality Policy and BLP Model

Confidentiality Policy

- Goal: prevent the unauthorized disclosure of information
  - Deals with information flow
  - Extensive redundancy in military makes integrity/availability less of a problem
- Multi-level security (MLS) models are best-known examples
  - Bell-LaPadula Model basis for many, or most, of these

Bell-LaPadula Model

- Security levels arranged in linear ordering

- Subjects have *security clearance L*(*s*)
- Objects have *security classification L*(*o*)

Example

security level	*subject*	*object*
Top Secret	Tamara	Personnel Files
Secret	Samuel	E-Mail Files
Confidential	Claire	Activity Logs
Unclassified	Ulaley	Telephone Lists

- Tamara can read all files
- Claire cannot read Personnel or E-Mail Files
- Ulaley can only read Telephone Lists

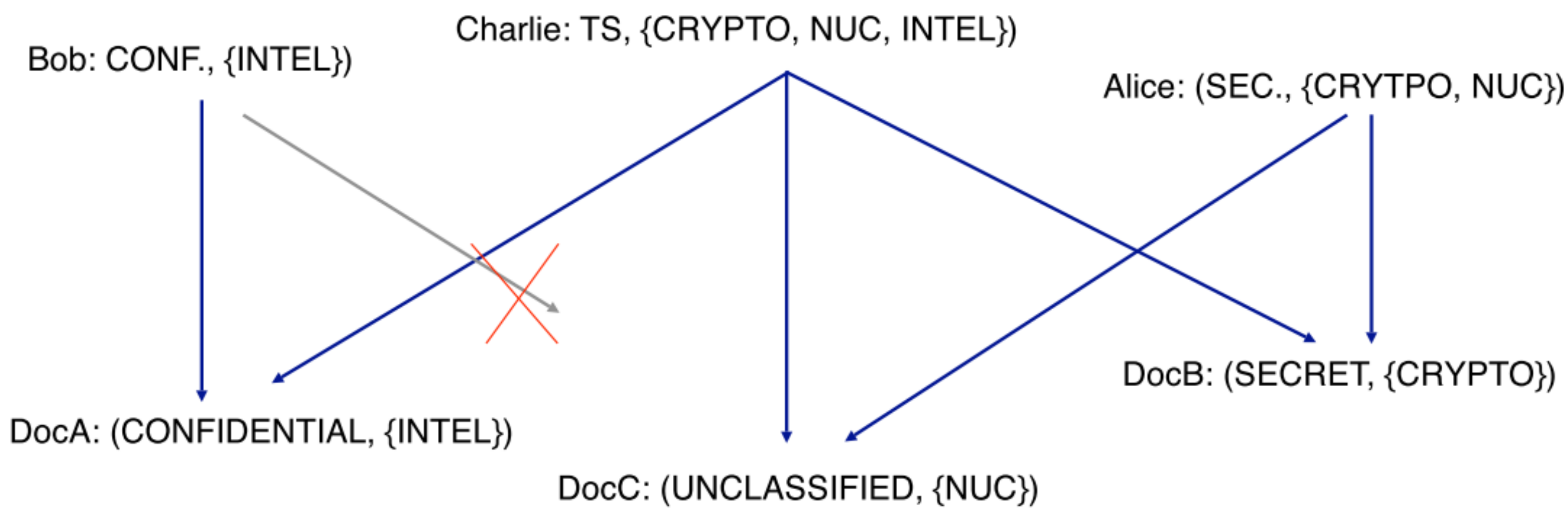
Lattice Model

- Used by the US military (and many others), the Lattice model uses MLS to define policy
- Expand notion of security level to include categories
  - Categories (actually unbounded set)
    - NUC(lear), INTEL(igence), CRYPTO(graphy)
    - Note that these levels are used for physical documents in the governments as well.
- Security level is (*clearance*, *category set*), or formally (*L*,*C*) where *L* is the clearance level, and *C* is the set of categories
- Examples
  - Alice: (SECRET, {CRYTPO, NUC})
  - Bob: (CONFIDENTIAL, {INTEL})
  - Charlie: (TOP SECRET, {CRYPTO, NUC, INTEL})
  - DocA: (CONFIDENTIAL, {INTEL})
  - DocB: (SECRET, {CRYPTO})

Reading Information

- Information flows *up*, not *down*
  - "Reads up" disallowed, "reads down" allowed
- Simple Security Condition
  - Subject *s* can read object *o* iff $L(o) \le L(s)$ and $C(o) \subseteq C(s)$, and *s* has permission to read *o*
    - The security level (L,C) dominates the security level (L',C') if $L' \le L$ and $C' \subseteq C$
  - Sometimes called "no reads up" rule

Example


Charlie: TS, {CRYPTO, NUC, INTEL})
Bob: CONF., {INTEL})
Alice: (SEC., {CRYTPO, NUC})
DocB: (SECRET, {CRYPTO})
DocA: (CONFIDENTIAL, {INTEL})
DocC: (UNCLASSIFIED, {NUC})

Writing Information

- Information flows up, not down
  - "Writes up" allowed, "writes down" disallowed

- *-Property
  - Subject s can write object o iff $L(s) \leq L(o)$ and $C(s) \subseteq C(o)$, and s has permission to write o
  - Sometimes called "no writes down" rule

Basic Security Theorem

- If a system is initially in a secure state, and every transition of the system satisfies the simple security condition and the *-property, then every state of the system is secure

Integrity Policy Model

Integrity Policy

- MLS as presented before talks about who can "read" a document (confidentiality)
- Integrity is considered who can "write" to a document
  - Thus, who can affect the integrity (content) of a document
  - Example: You may not care who can read DNS records, but you better care who writes to them!

Biba Integrity Model

- I is a set of integrity levels
  - Function i: $S \cup O \rightarrow I$
- Important point: Integrity labels are different from security labels
  - Security labels limit the flow of information; integrity labels inhibit modification of information
- The higher the level, the more confidence
  - That a program will execute correctly
  - That data is accurate and/or reliable
- Note relationship between integrity and trustworthiness

Biba Model

- Biba defined a dual of secrecy for integrity
  - Lattice policy with, "no read down, no write up"
  - Users can only create content at or below their own integrity level (a monk may write a prayer book that can be read by commoners, but not one to be read by a high priest).
  - Users can only view content at or above their own integrity level (a monk may read a book written by the high priest, but may not read a pamphlet written by a lowly commoner).

Examples

- Which users can modify what documents?

Charlie: TS, {CRYPTO, NUC, INTEL})

Bob: CONF., {INTEL})

Alice: (SEC., {CRYTPO, NUC})

?????

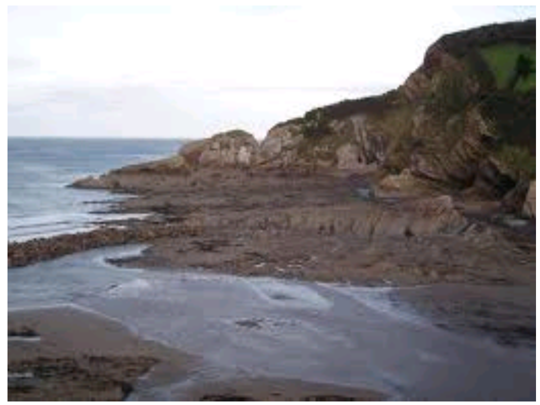
DocB: (SECRET, {CRYPTO})

DocA: (CONFIDENTIAL, {INTEL})

DocC: (UNCLASSIFIED, {NUC})

Low-Water Mark integrity

- Change integrity level based on actual dependencies



- Subject is initially at the highest integrity
  - But integrity level can change based on objects accessed
- Ultimately, subject has integrity of lowest object read

Hybrid Policy Model: Chinese Wall and RBAC

Chinese Wall Model

- A model of a security policy that refers equally to confidentiality and integrity
  - Deals with conflict of interest situations

- Problem:
  - Tony advises American Bank about investments
  - He is asked to advise Toyland Bank about investments
  - Conflict of interest to accept, because his advice for either bank would affect his advice to the other bank

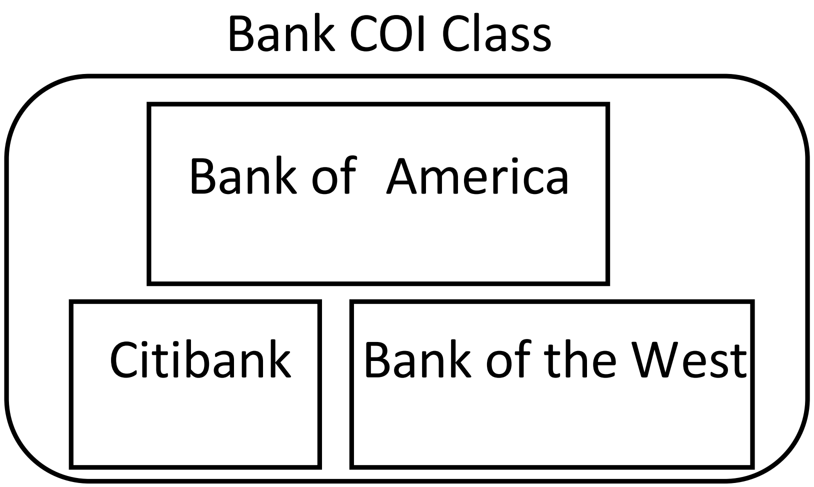
Organization

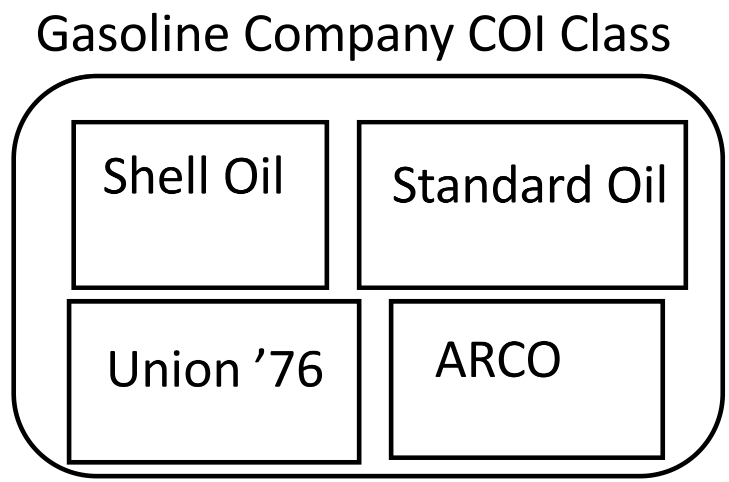
- Organize entities into “conflict of interest” classes
- Control subject accesses to each class
- Control writing to all classes to ensure information is not passed along in violation of rules
- Allow sanitized data to be viewed by everyone

Definitions

- The objects are items of information related to a company
- A company dataset (CD) contains objects related to a single company
- A conflict of interest (COI) class contains the datasets of companies in competition
- CD(O): the company dataset that contains object O; COI(O): the COI class that contains object O

Example


Bank COI Class
Bank of America
Citibank
Bank of the West


Gasoline Company COI Class
Shell Oil
Standard Oil
Union ’76
ARCO

CW-Simple Security Condition

- If Anthony reads any CD in a COI, he can *never* read another CD in that COI
  - Possible that information learned earlier may allow him to make decisions later
- S can read O if and only if any of the following holds
  - There is an object O' such that S has accessed O' and CD(O') = CD(O)
  - For all objects O', $O' \in PR(S) \rightarrow COI(O') \neq COI(O)$
    - PR(S) is the set of objects S has read
  - O is a sanitized object

Writing

- Anthony, Susan work in same trading house
- Anthony can read Bank 1’s CD, Gas’ CD
- Susan can read Bank 2’s CD, Gas’ CD
- If Anthony could write to Gas’ CD, Susan can read it
  - Hence, indirectly, she can read information from Bank 1’s CD, a clear conflict of interest

CW-*-Property

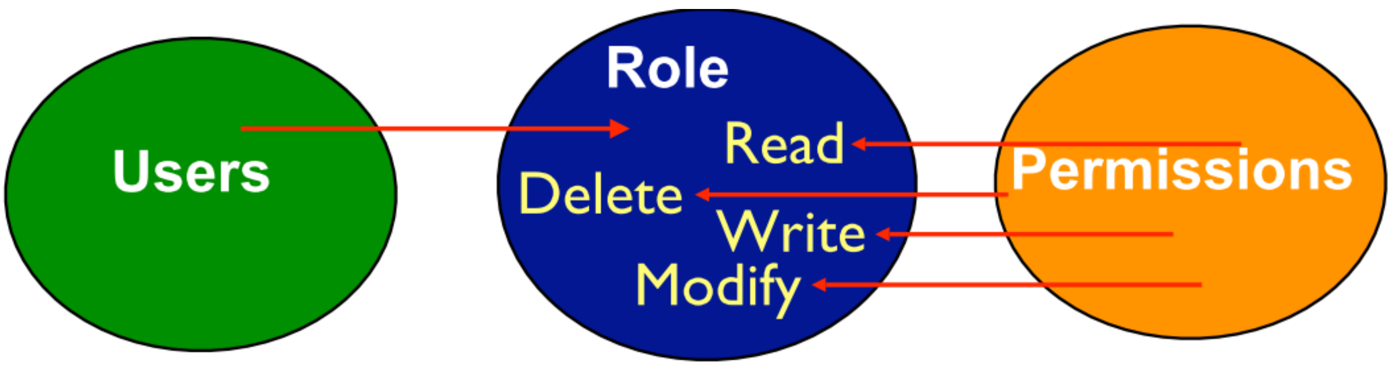
- s can write to o iff both of the following hold:
    1. The CW-simple security condition permits s to read o; and
    2. For all *unsanitized* objects o', if s can read o', then $CD(o') = CD(o)$
- Says that s can write to an object if all the (unsanitized) objects it can read are in the same dataset

RBAC: Role-Based Access Control

- In an enterprise, we don’t really do anything as ourselves, we do things as some job function
  - E.g., student, professor, doctor
- Access depends on function, not identity
  - Example:
    - Allison, bookkeeper for Math Dept, has access to financial records.
    - She leaves.
    - Betty hired as the new bookkeeper, so she now has access to those records
  - The role of “bookkeeper” dictates access, not the identity of the individual

Roles

- A role is a collection of privileges/permissions associated with some function or affiliation



- Important: the permissions are static, the user-role membership is transient
- Not direct MAC and DAC, but may one or either of these.

Summary

- Access control model: express security policy

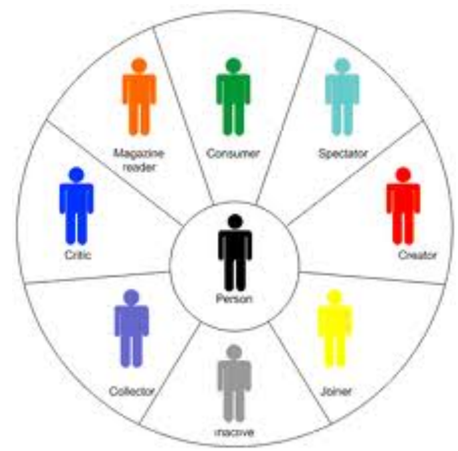
- Confidentiality Policy and BLP Model (no read up, no write down)
- Integrity Policy and Bida model (no read down, no write up)

- Hybrid Policy Model
  - Chinese Wall (conflict of interest)
  - RBAC (the importance of role)

Appendix

Role-Based Access Control

- Definitions
  - A role is a collection of job functions. Each role r is authorized to perform one or more transactions. The set of authorized transactions for r is trans(r)
  - The active role of subject s, actr(s), is the role that s is currently performing
  - The authorized roles of a subject s, authr(s), is the set of roles that s is authorized to assume



Role-Based Access Control

- Axioms
  - If a subject can execute any transaction, then that subject has an active role
    - Binds the notion of execution to role rather than user
  - A subject must be authorized to assume its active role
    - Cannot assume an unauthorized role
  - A subject cannot execute a transaction for which its current role is not authorized