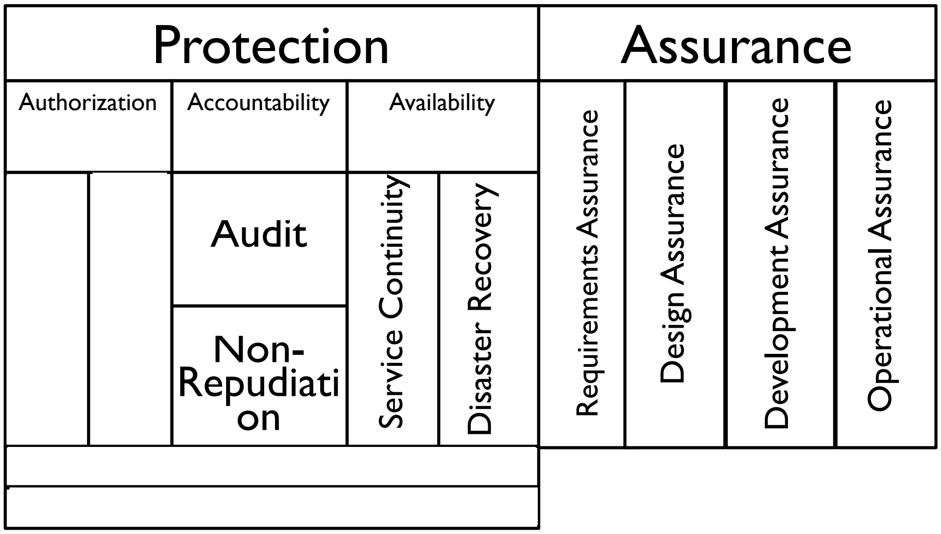
THE UNIVERSITY OF BRITISH COLUMBIA

Access Control

read:
Stamp: sections 8.1-8.4, 8.8-8.10
Anderson: chapters 4, 7, 8.

Where We Are



Protection

Assurance

Authorization

Accountability

Availability

Audit

Non-Repudiation

Service Continuity

Disaster Recovery

Requirements Assurance

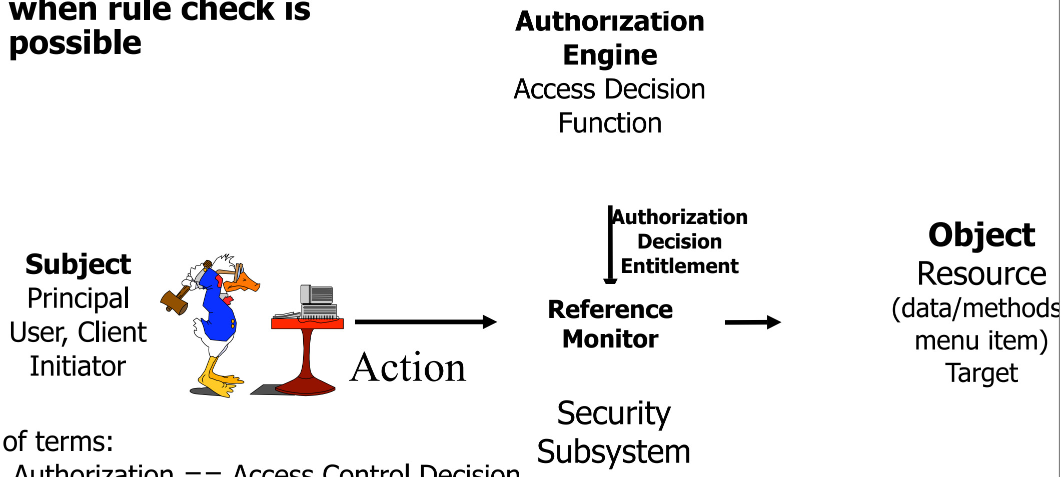
Design Assurance

Development Assurance

Operational Assurance

Authorization Mechanisms:
Access Control

Definition: **enforces the rules, when rule check is possible**



Authorization Engine
Access Decision Function

Authorization Decision Entitlement

Subject
Principal
User, Client
Initiator

Action

Reference Monitor

Security Subsystem

Object
Resource
(data/methods
menu item)
Target

Mix of terms:

Authorization == Access Control Decision
Authorization Engine == Policy Engine

Policies and Mechanisms

- Policies describe what is allowed
- Mechanisms control how policies are enforced

Access Matrix

Lampson’s Access Control Matrix

Subjects (users) index the rows

Objects (resources) index the columns

	OS	Accounting program	Accounting data	Insurance data	Payroll data
Bob	rx	rx	r	---	---
Alice	rx	rx	r	rw	rw
Sam	rwx	rwx	r	rw	rw
Accounting program	rx	rx	rw	rw	rw

why access matrix is not used

- **Access control matrix** has all relevant info
- But how to manage a large access control (AC) matrix?
- Could be 1000’s of users, 1000’s of resources
- Then AC matrix with 1,000,000’s of entries
- Need to check this matrix before access to any resource is allowed
- Hopelessly inefficient

Access Control Lists

- ACL: store access control matrix by **column**
- Example: ACL for **insurance data** is in **yellow**

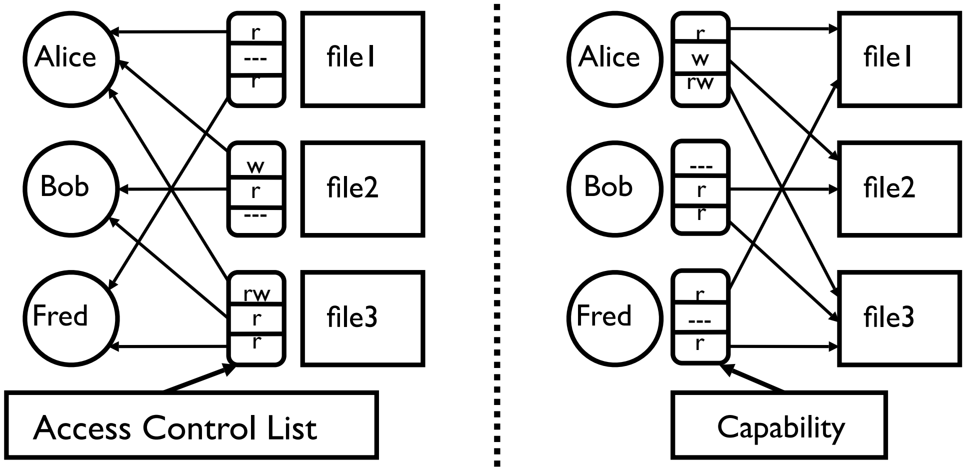
	OS	Accounting program	Accounting data	**Insurance data**	Payroll data
Bob	rx	rx	r	---	---
Alice	rx	rx	r	rw	rw
Sam	rwx	rwx	r	rw	rw
Accounting program	rx	rx	rw	rw	rw

Capabilities (or C-Lists)

- Store access control matrix by **row**
- Example: Capability for **Alice** is in **blue**

	OS	Accounting program	Accounting data	Insurance data	Payroll data
Bob	rx	rx	r	---	---
Alice	rx	rx	r	rw	rw
Sam	rwx	rwx	r	rw	rw
Accounting program	rx	rx	rw	rw	rw

ACLs vs Capabilities



- Note that arrows point in opposite directions!
- With ACLs, still need to associate users to files

ACLs vs Capabilities

- ACLs
  - Good when users manage their own files
  - Protection is data-oriented
  - Easy to change rights to a resource
- Capabilities
  - Easy to delegate
  - Easy to add/delete users
  - Easier to delegate rights
  - Harder to control the delegation
  - More difficult to implement
  - The "Zen of information security"

Security Policies

what's secure system?

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

THE UNIVERSITY OF BRITISH COLUMBIA

CIA

What’s Confidentiality?

- X set of entities, I information
- I has confidentiality property with respect to X if no x ∈ X can obtain information from I
- I can be disclosed to others
- Example:

what’s confidentiality policy?

- Goal: prevent the unauthorized disclosure of information
  - Deals with information flow
  - Integrity incidental
- Multi-level security models are best-known examples
  - Bell-LaPadula Model basis for many, or most, of these

What’s Integrity?

- X set of entities, I information
- I has integrity property with respect to X if all x ∈ X trust information in I
- Examples?

Types of Access Control

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

Multilevel Security (MLS) Models

Classifications and Clearances

- **Classifications** apply to **objects**
- **Clearances** apply to **subjects**
- US Department of Defense uses 4 levels of classifications/clearances

  TOP SECRET

  SECRET

  CONFIDENTIAL

  UNCLASSIFIED

Clearances and Classification

- To obtain a **SECRET** clearance requires a routine background check
- A **TOP SECRET** clearance requires extensive background check
- Practical classification problems
  - Proper classification not always clear
  - Level of granularity to apply classifications
  - Aggregation — flipside of granularity

Subjects and Objects

- Let O be an **object**, S a **subject**
  - O has a classification
  - S has a clearance
- o Security **level** denoted L(O) and L(S)
- For DoD levels, we have

TOP SECRET > SECRET > CONFIDENTIAL > UNCLASSIFIED

Multilevel Security (MLS)

- MLS needed when subjects/objects at different levels use same system
- MLS is a form of **Access Control**
- Classified government/military information
- **Business example:** info restricted to
  - Senior management only
  - All management
  - Everyone in company
  - General public
- Network firewall
  - Keep intruders at low level to limit damage
- Confidential medical info, databases, etc.

Example

security level	subject	object
Top Secret	Alice	Personnel Files
Secret	Bob	E-Mail Files
Confidential	Chiang	Activity Logs
Unclassified	Fred	Telephone Lists

- Alice can read all files
- Chiang cannot read Personnel or E-Mail Files
- Fred can only read Telephone Lists

Bell-LaPadula

- BLP security model designed to express essential requirements for MLS
- BLP deals with **confidentiality**
  - To prevent unauthorized reading
- Recall that O is an object, S a subject
  - Object O has a classification
  - Subject S has a clearance
  - Security level denoted L(O) and L(S)

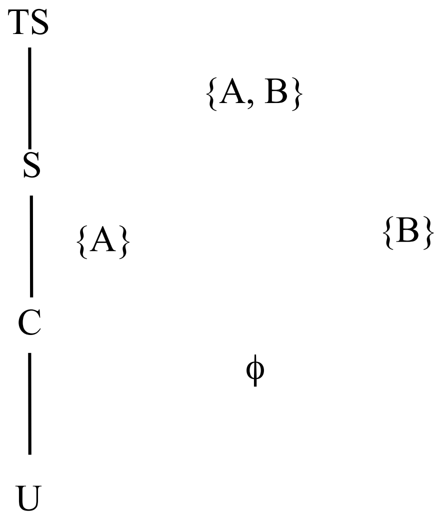
BLP rules

Simple Security Condition: S can read O if and only if L(O) ≤ L(S)

***-Property** (**Star Property**): S can write O if and only if L(S) ≤ L(O)

- **No read up, no write down**

The Military Lattice



TS

S

{A, B}

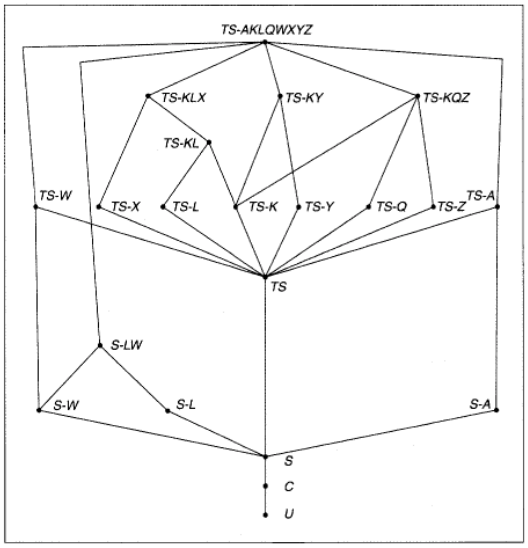
{A}

{B}

C

φ

U



Key Points Regarding Confidentiality Policies

- Confidentiality policies restrict flow of information
- Bell-LaPadula model supports multilevel security
  - Cornerstone of much work in computer security

Chinese Wall Model

What’s Chinese Wall Model

Problem:

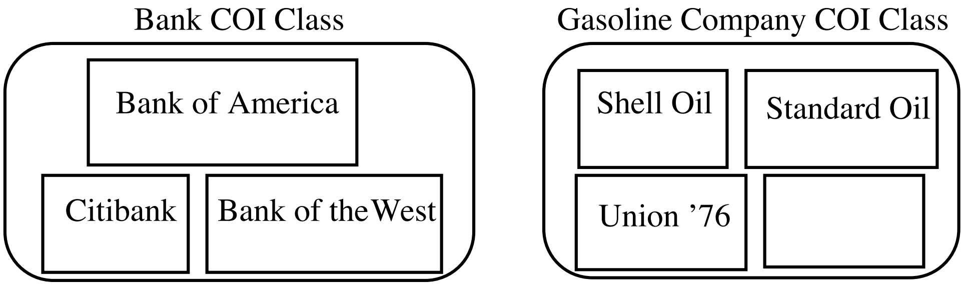
- Tony advises American Bank about investments
- He is asked to advise Toyland Bank about investments

- 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

Example



Bank COI Class

Bank of America

Citibank

Bank of the West

Gasoline Company COI Class

Shell Oil

Standard Oil

Union ’76

- If Anthony reads any Company dataset (CD) in a conflict of interest (COI), he can never read another CD in that COI
  - Possible that information learned earlier may allow him to make decisions later

CW-Simple Security Condition

- s can read o iff either condition holds:
    1. There is an o' such that s has accessed o' and $CD(o') = CD(o)$
        1. Meaning s has read something in o's dataset
    2. For all $o' \in O$, $o' \in PR(s) \Rightarrow COI(o') \neq COI(o)$
        1. Meaning s has not read any objects in o's conflict of interest class

1. Ignores sanitized data (see below)

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

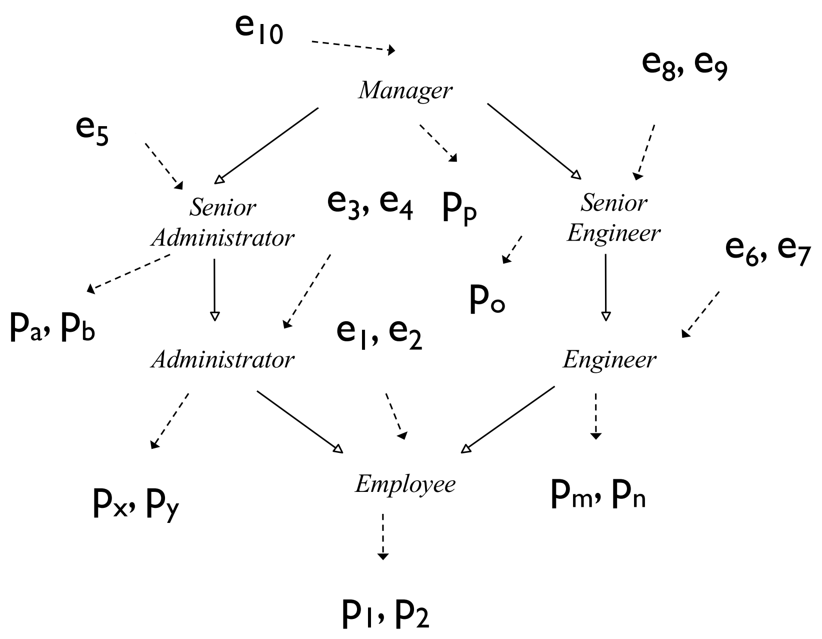
THE UNIVERSITY OF BRITISH COLUMBIA

Role-based Access Control (RBAC)

RBAC

- Access depends on role, not identity or label
  - Example:
    - Allison, administrator for a department, has access to financial records.
    - She leaves.
    - Betty hired as the new administrator, so she now has access to those records

Example



e_{10}

Manager

e_8, e_9

e_5

Senior Administrator

e_3, e_4

p_p

Senior Engineer

p_o

e_6, e_7

p_a, p_b

Administrator

e_1, e_2

Engineer

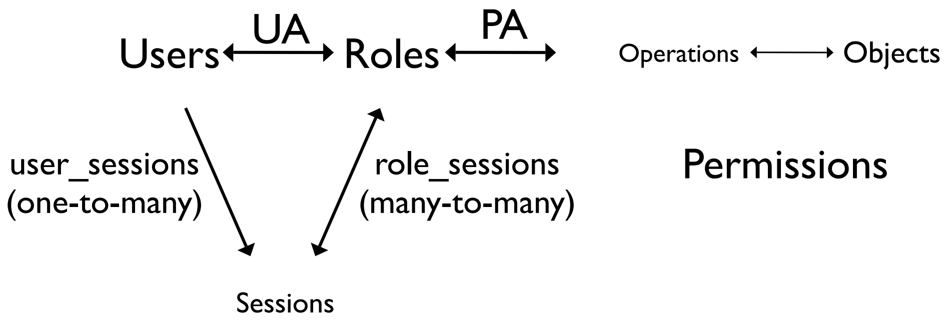
Employee

p_x, p_y

p_m, p_n

p_1, p_2

RBAC (ANSI Standard)



Users

UA

Roles

PA

Operations

Objects

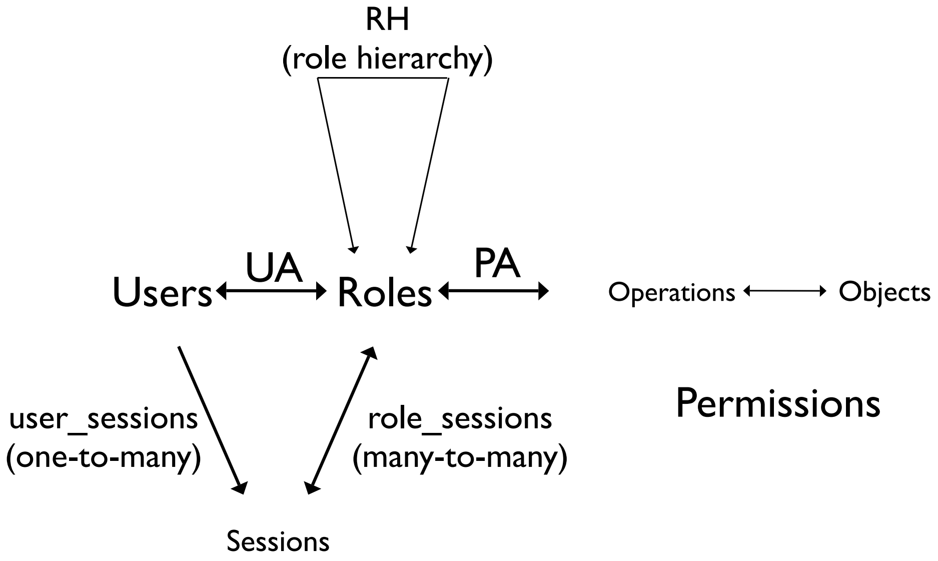
user_sessions (one-to-many)

role_sessions (many-to-many)

Permissions

Sessions

RBAC with General Role Hierarchy



RH
(role hierarchy)

Users UA Roles PA Operations Objects

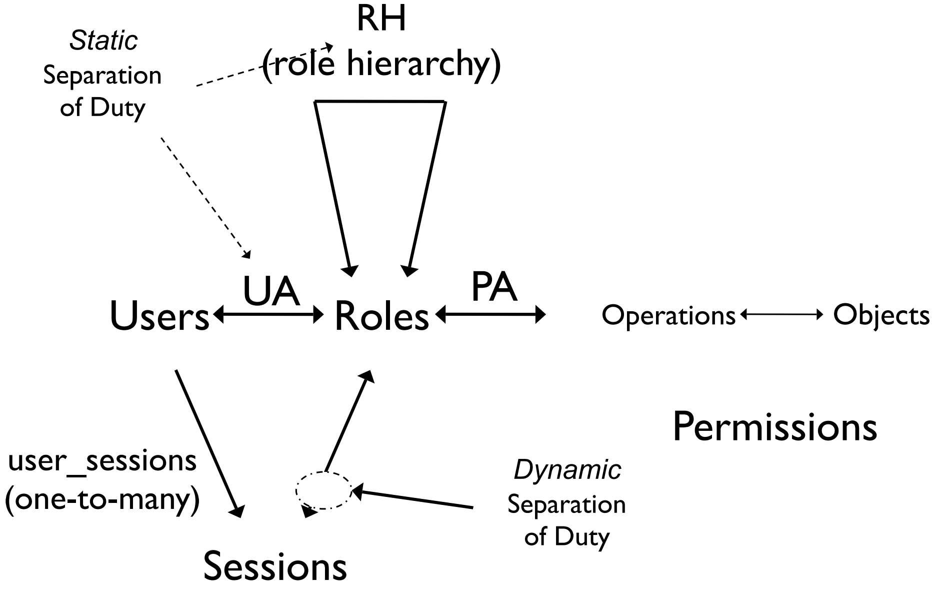
Permissions

user_sessions
(one-to-many)

role_sessions
(many-to-many)

Sessions

Constrained RBAC



RH
(role hierarchy)

Static
Separation
of Duty

Users UA Roles PA Operations Objects

Permissions

user_sessions
(one-to-many)

Dynamic
Separation
of Duty

Sessions