Access Control

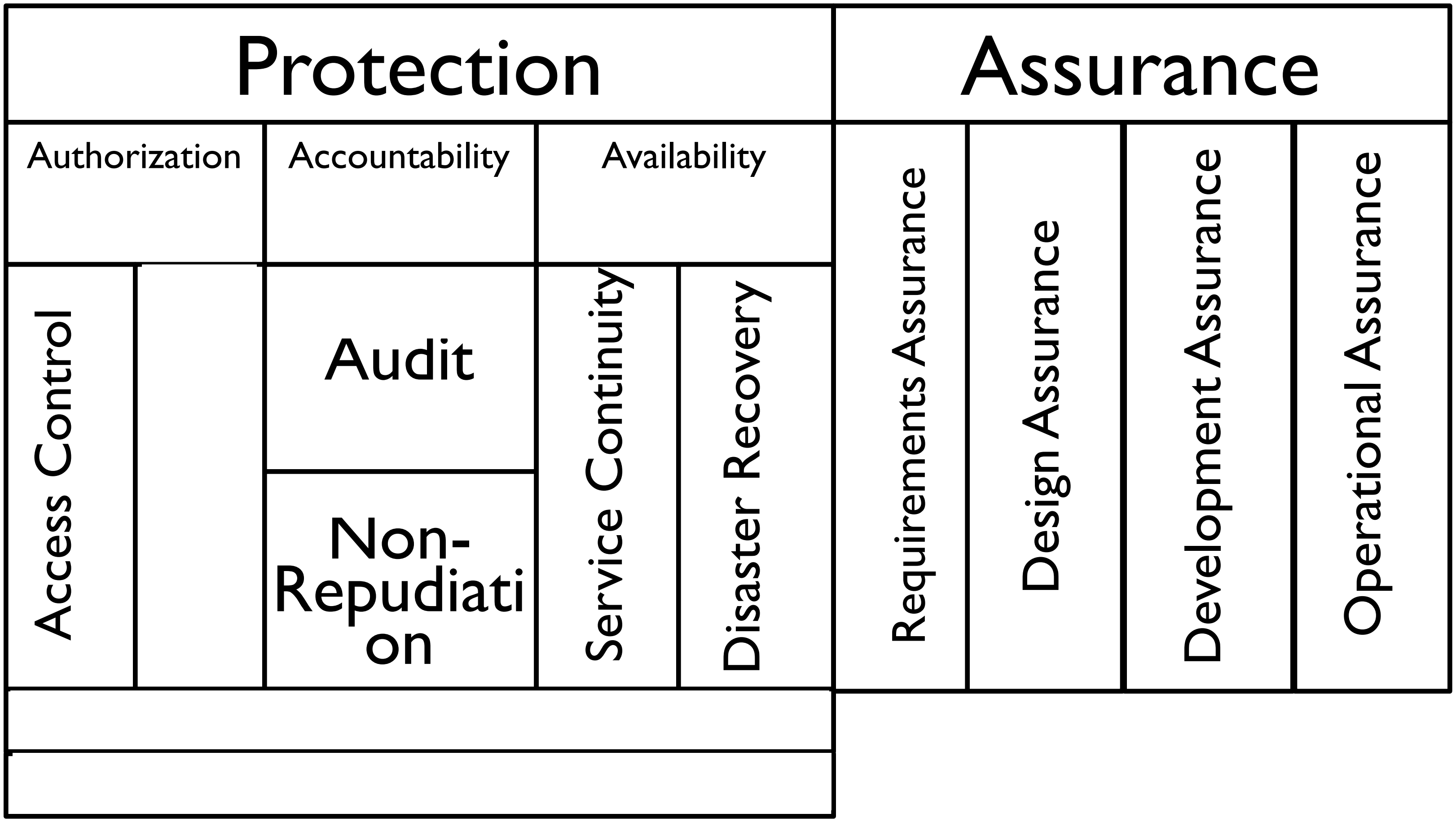
read:
Stamp: sections 8.1-8.4, 8.8-8.10
Anderson: chapters 4, 8, 9, 10.

learning objectives

you should be able to

- explain confidentiality and integrity in terms of security policies
- explain c-lists and ACLs and differences between the two
- explain main access control poly models (BLP, CW, RBAC, DAC)
- convert a policy from one model to another

Where We Are



Protection

Authorization

Accountability

Availability

Access Control

Audit

Non-Repudiation

Service Continuity

Disaster Recovery

Assurance

Requirements Assurance

Design Assurance

Development Assurance

Operational Assurance

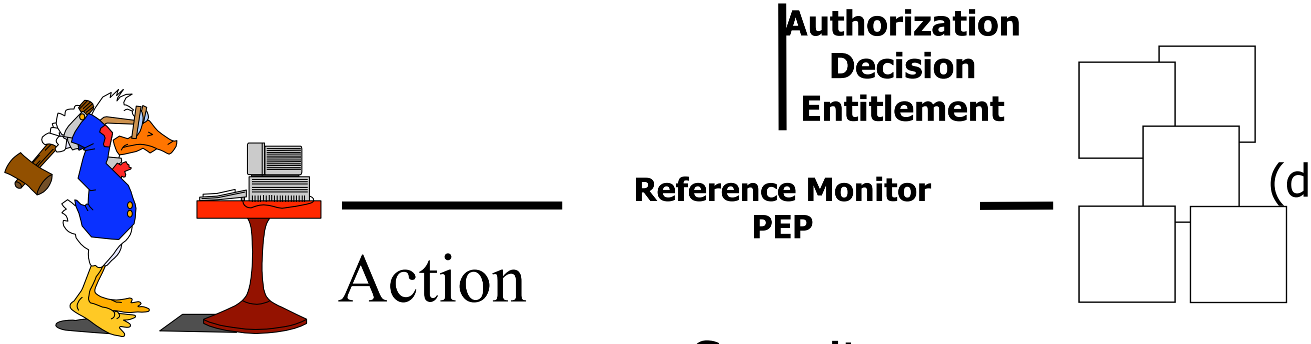
Ross Anderson

"If you say that your problem can be solved with cryptography, then you don't understand your problem and you don't understand cryptography."

Roger Needham

Authorization Mechanisms:
Access Control

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



Authorization Engine
Access Decision
Function
PDP

Authorization
Decision
Entitlement

Subject
Principal
User, Client
Initiator

Action

Reference Monitor
PEP

Object
Resource
(data/methods
menu item)
Target

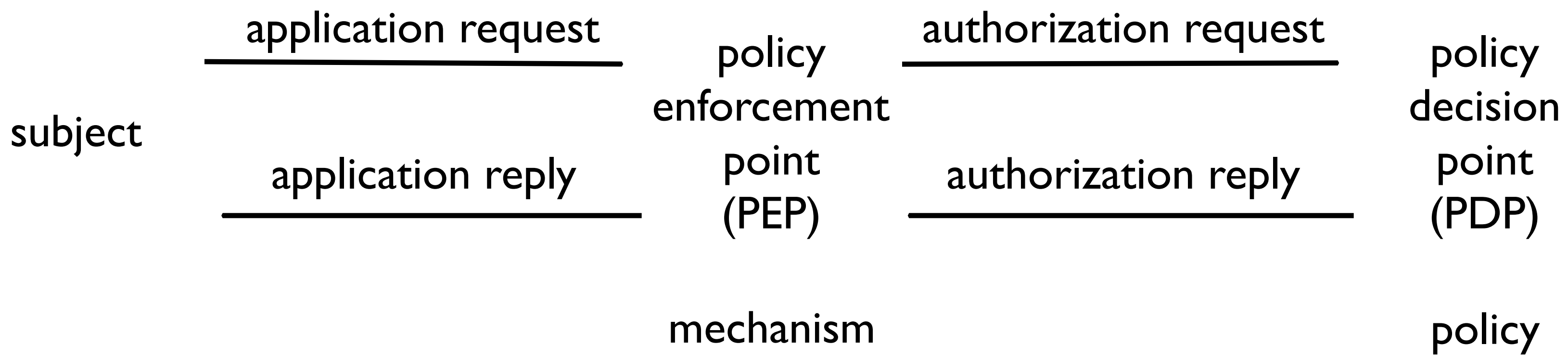
Security
Subsystem

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



subject

application request

application reply

policy
enforcement
point
(PEP)

authorization request

authorization reply

policy
decision
point
(PDP)

mechanism

policy

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 1,000's of users, 1,000'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**

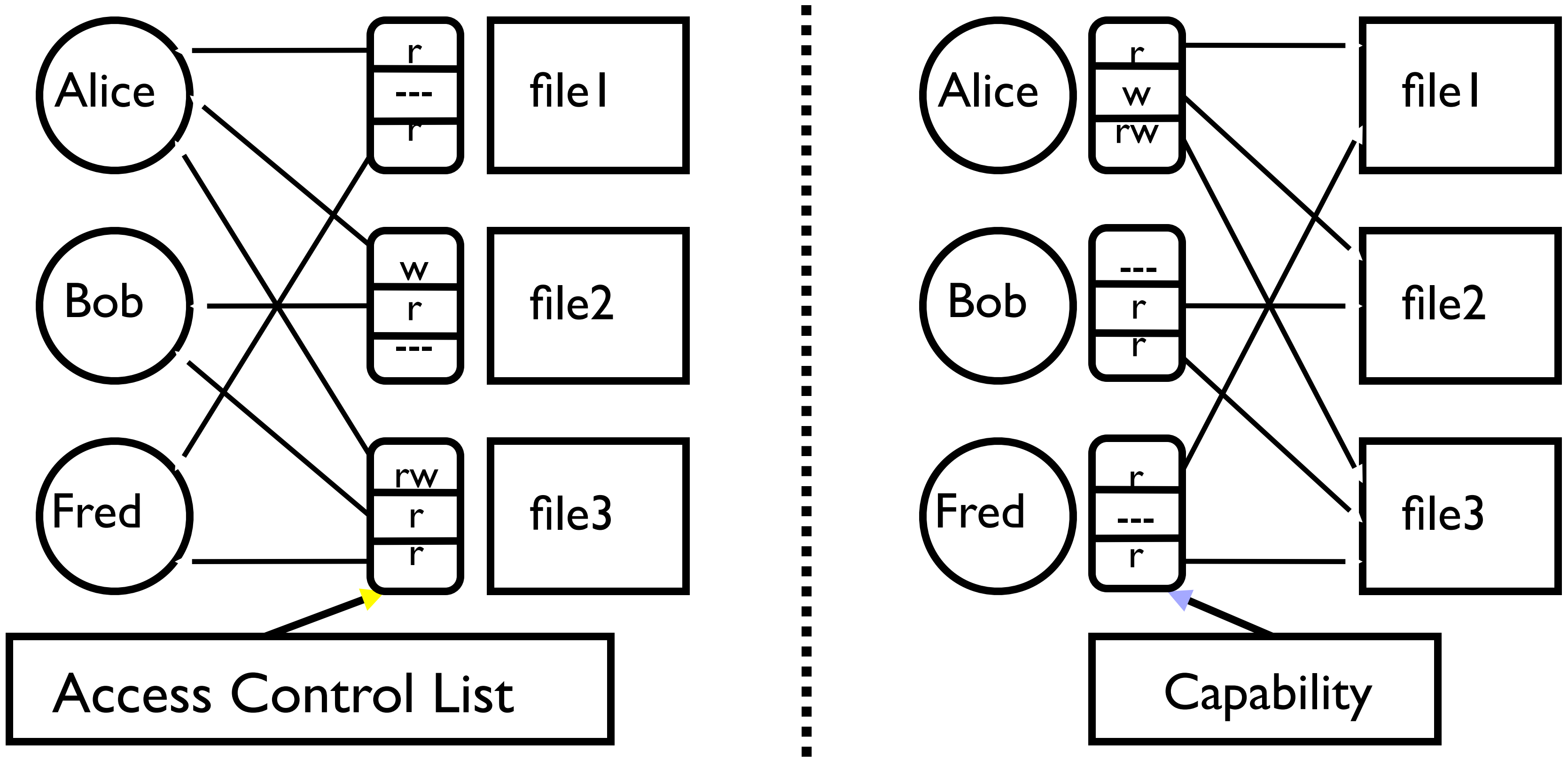
	OS	Accounting program	Accounting data	**Insurance data**	Payroll data
Bob	rx	rx	r	---	---
Alice	rx	rx	r	rw	rw
Sam	rwx	rwx	r	rw	rw
Accounting program	rx	rx	rw	rw	rw

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



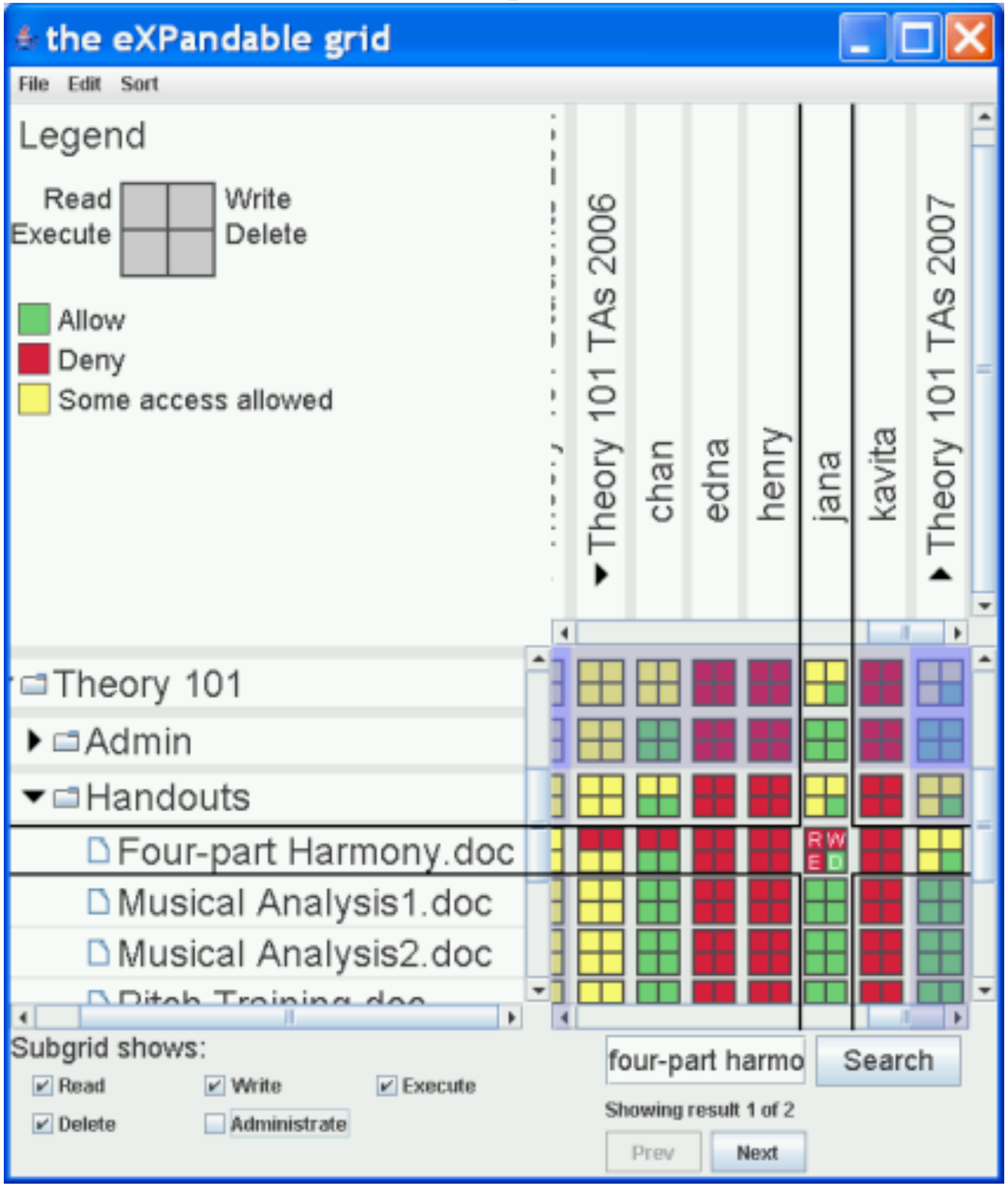
- Note that arrows point in opposite directions!
- With ACLs, still need to associate users to files

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"

can jana read Four-part Harmony.doc?

can jana read Four-part Harmony.doc?



Legend

Read Write
Execute Delete

🟩 Allow
🟥 Deny
🟨 Some access allowed

Theory 101 TAs 2006, chan, edna, henry, jana, kavita, Theory 101 TAs 2007

Theory 101
Admin
Handouts
Four-part Harmony.doc
Musical Analysis1.doc
Musical Analysis2.doc
Pitch Training.doc

R W
E D

Subgrid shows:

☑ Read ☑ Write ☑ Execute
☑ Delete ☐ Administrate

four-part harmo

Showing result 1 of 2

source: http://www.robreeder.com/projects/xgrids.html

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)

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:
  - X set of students
  - I final exam answer key
  - I is confidential with respect to X if students cannot obtain final exam answer key

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 Policies

- 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 $\mathrm{L(O)}$ and $\mathrm{L(S)}$

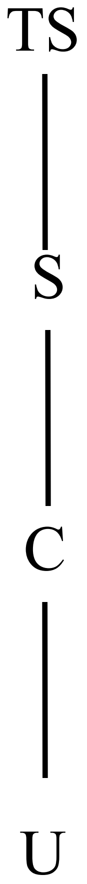
BLP rules

Simple Security Condition: S can read O if and only if $L(O) \leq L(S)$

***-Property** (**Star Property**): S can write O if and only if $L(S) \leq L(O)$

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

The Military Lattice

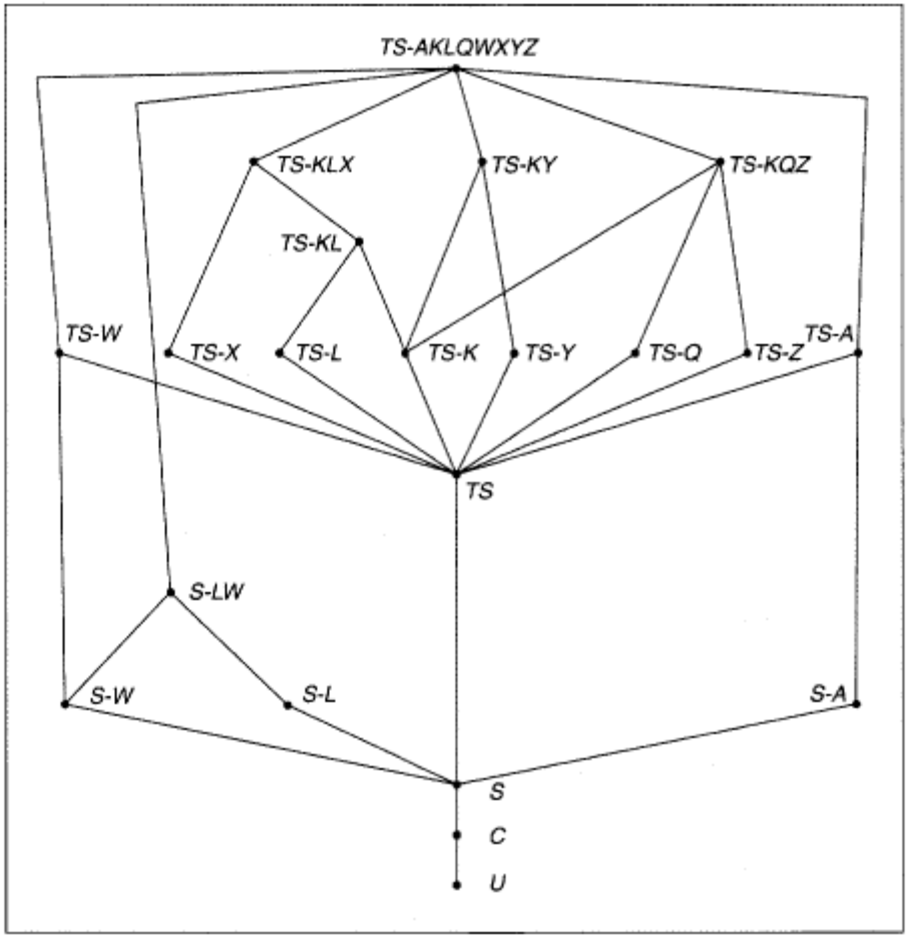


{A, B}

{A}

{B}

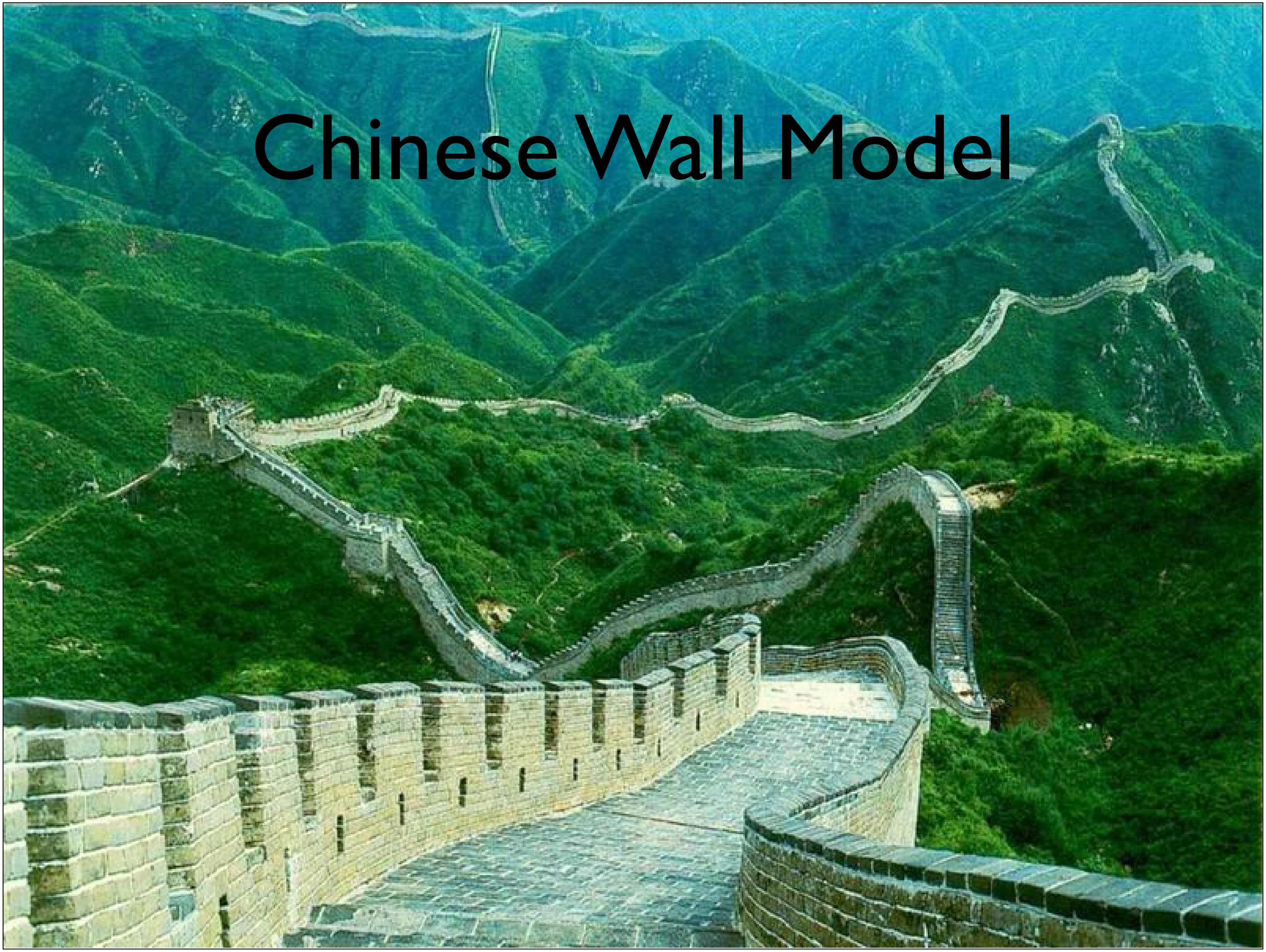
ϕ



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

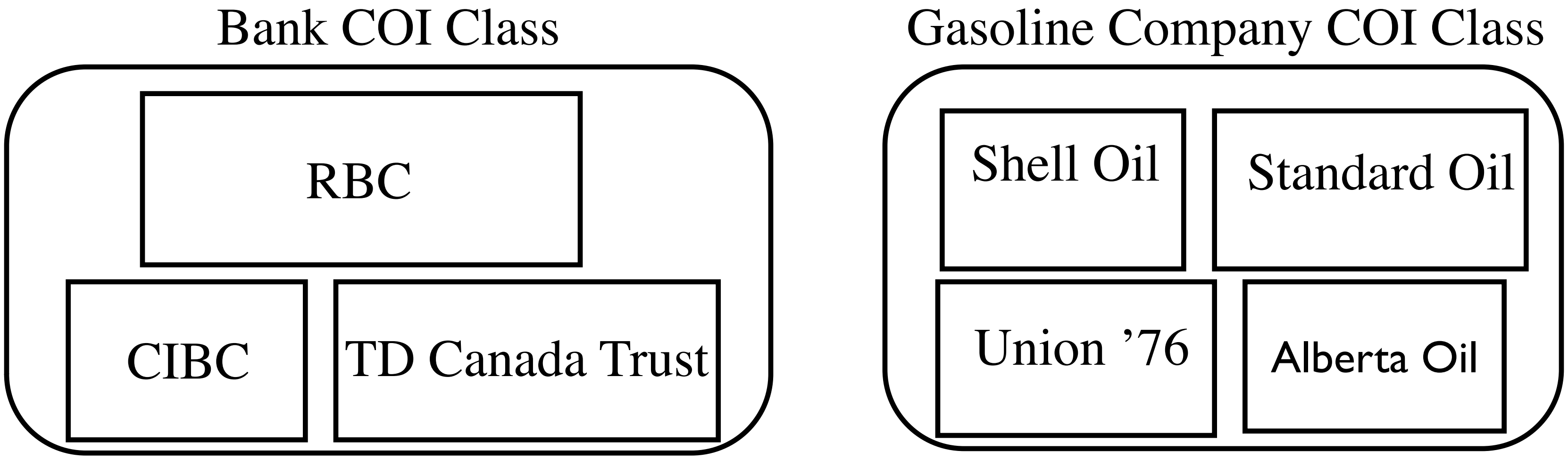
Problem:

  - 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



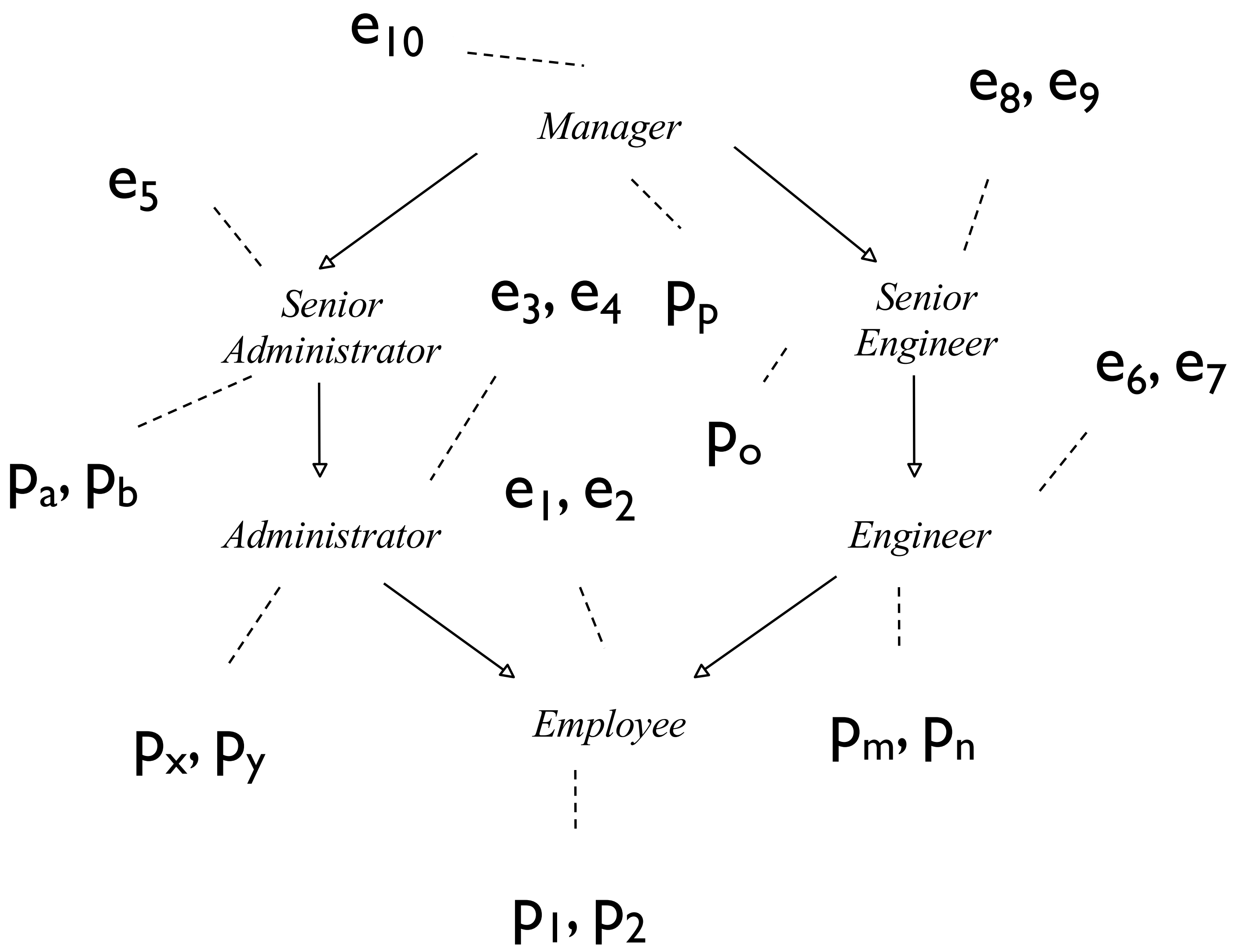
- If Anthony reads any Company dataset (CD) in a conflict of interest (COI), he can never read another CD in that COI
  - Possible that information learned earlier may allow him to make decisions later

Role-based Access Control (RBAC)

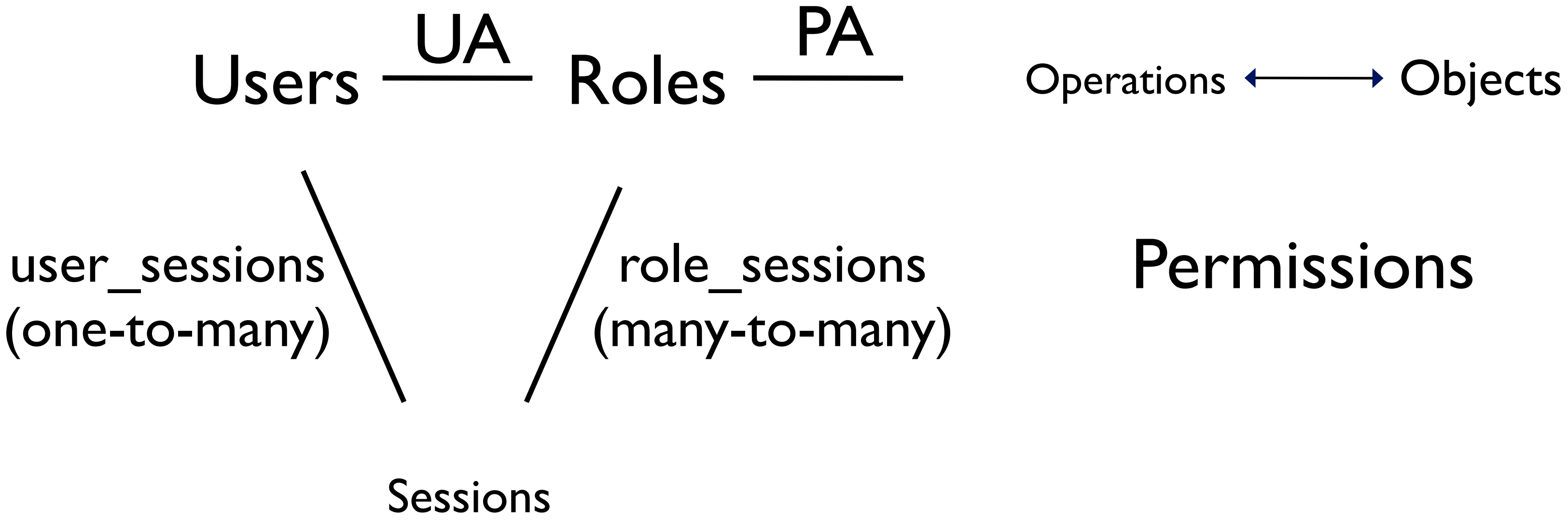
RBAC

- 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
  - The role of "administrator" dictates access, not the identity of the individual.

Example


e_{10}
e_8, e_9
Manager
e_5
Senior Administrator
e_3, e_4
p_p
Senior Engineer
e_6, e_7
p_o
p_a, p_b
e_1, e_2
Administrator
Engineer
Employee
p_x, p_y
p_m, p_n
p_1, p_2

RBAC (ANSI Standard)



Users UA Roles PA

Operations ⟷ Objects

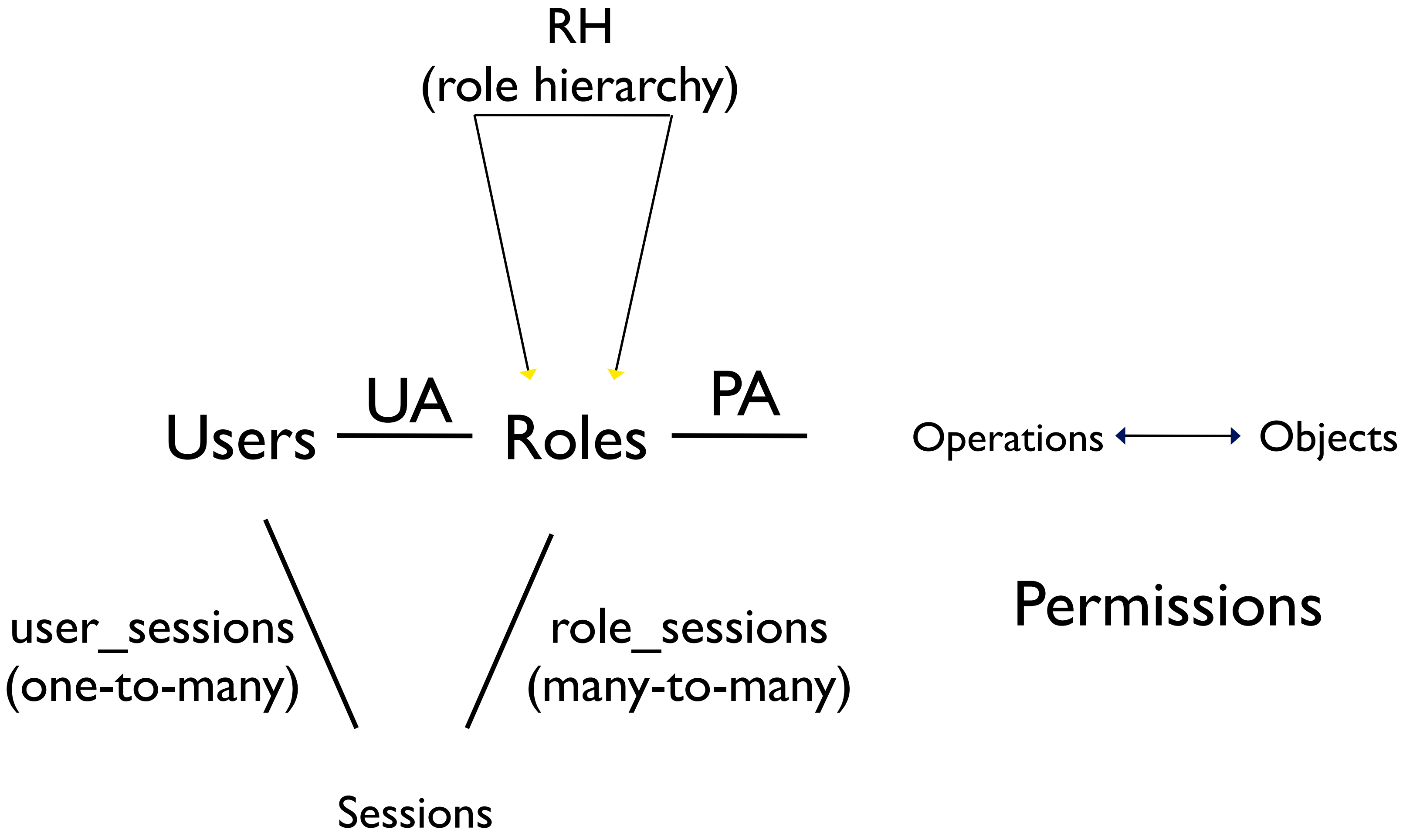
user_sessions
(one-to-many)

role_sessions
(many-to-many)

Permissions

Sessions

RBAC with General Role Hierarchy



RH
(role hierarchy)

Users

UA

Roles

PA

Operations

Objects

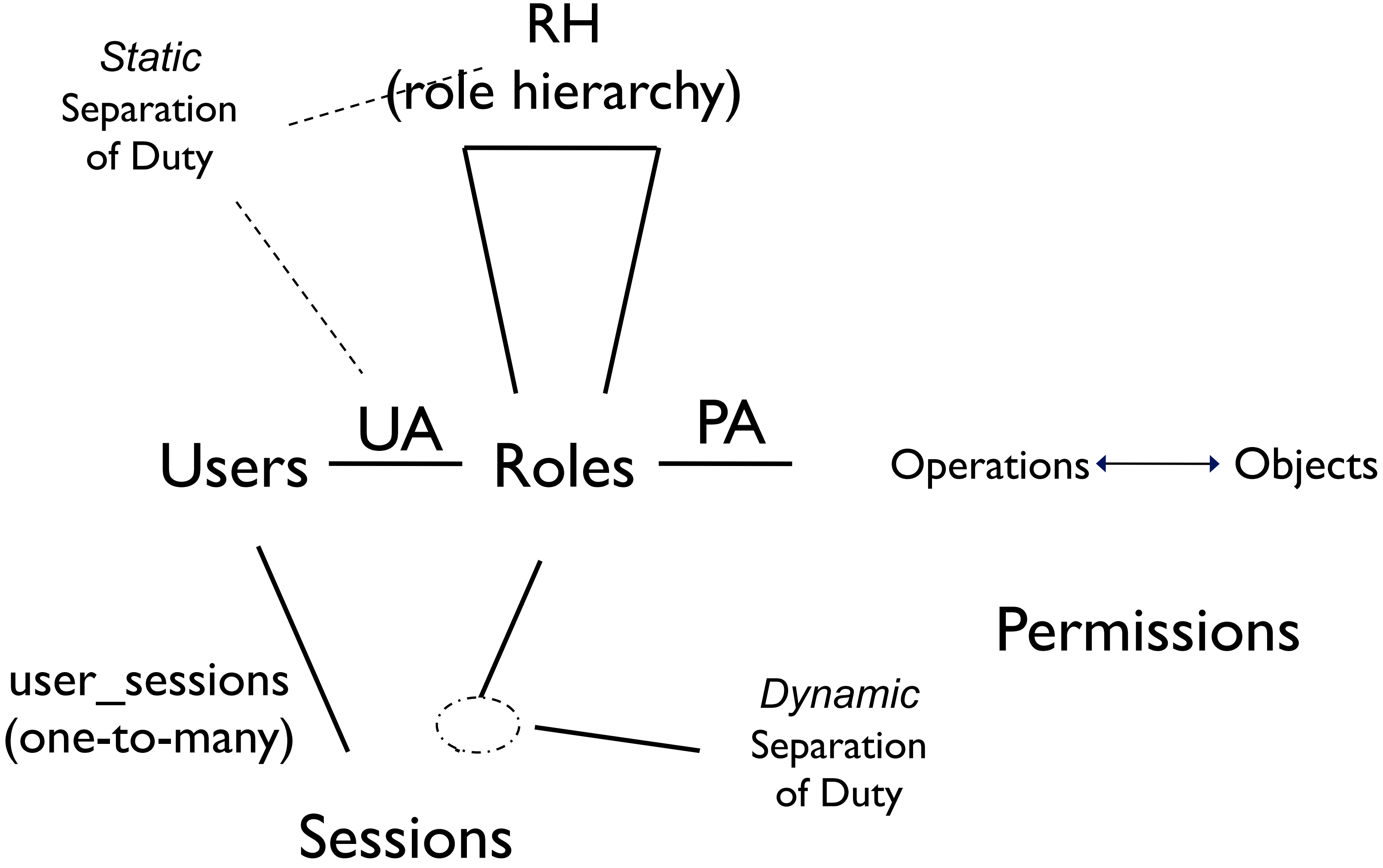
user_sessions
(one-to-many)

role_sessions
(many-to-many)

Permissions

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

what we learned so far

- structure of access controls (PEP & PDP)
- access matrix
    - ACLs and capability lists
- security policies
    - confidentiality & integrity
    - types of policies (DAC, MAC, OrCon)
- BLP model
- Chinese Wall model
- RBAC model