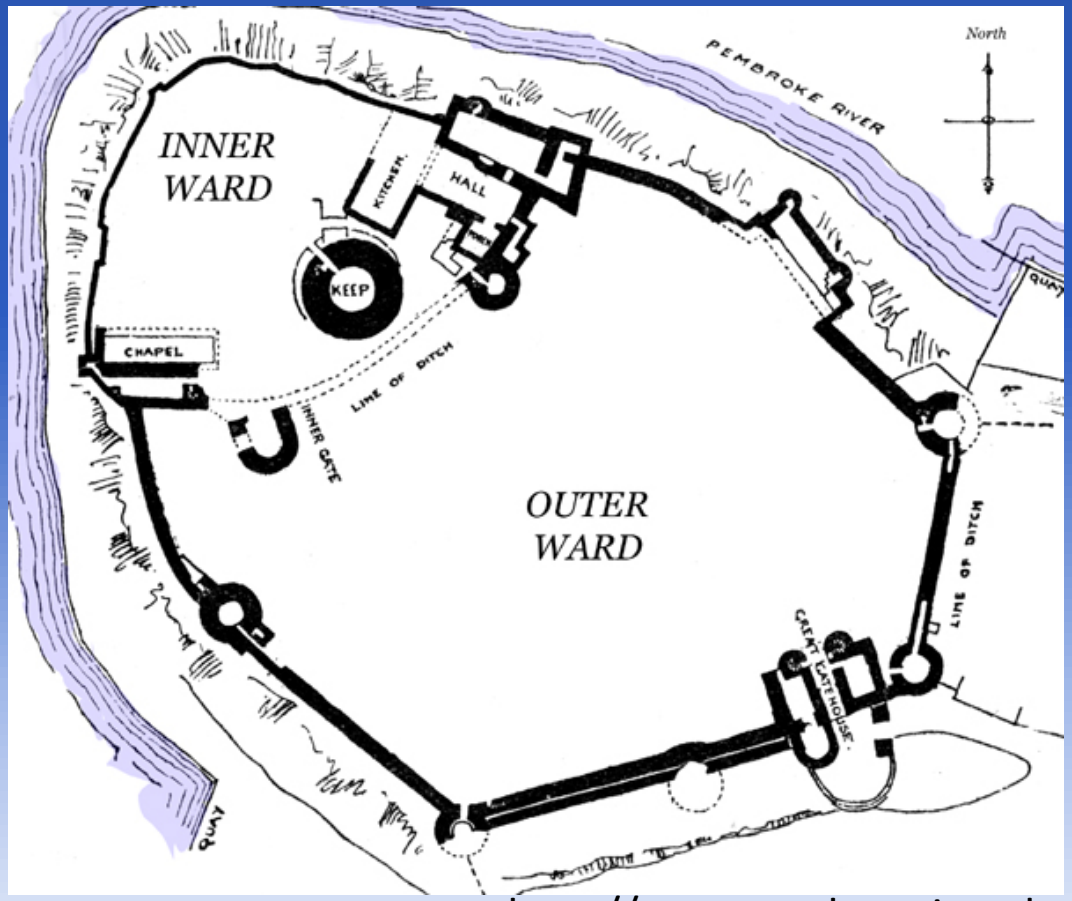
CS 493
Secure Software Systems
Access Control



http://www.medart.pitt.edu

Dr. Williams Central Connecticut State University

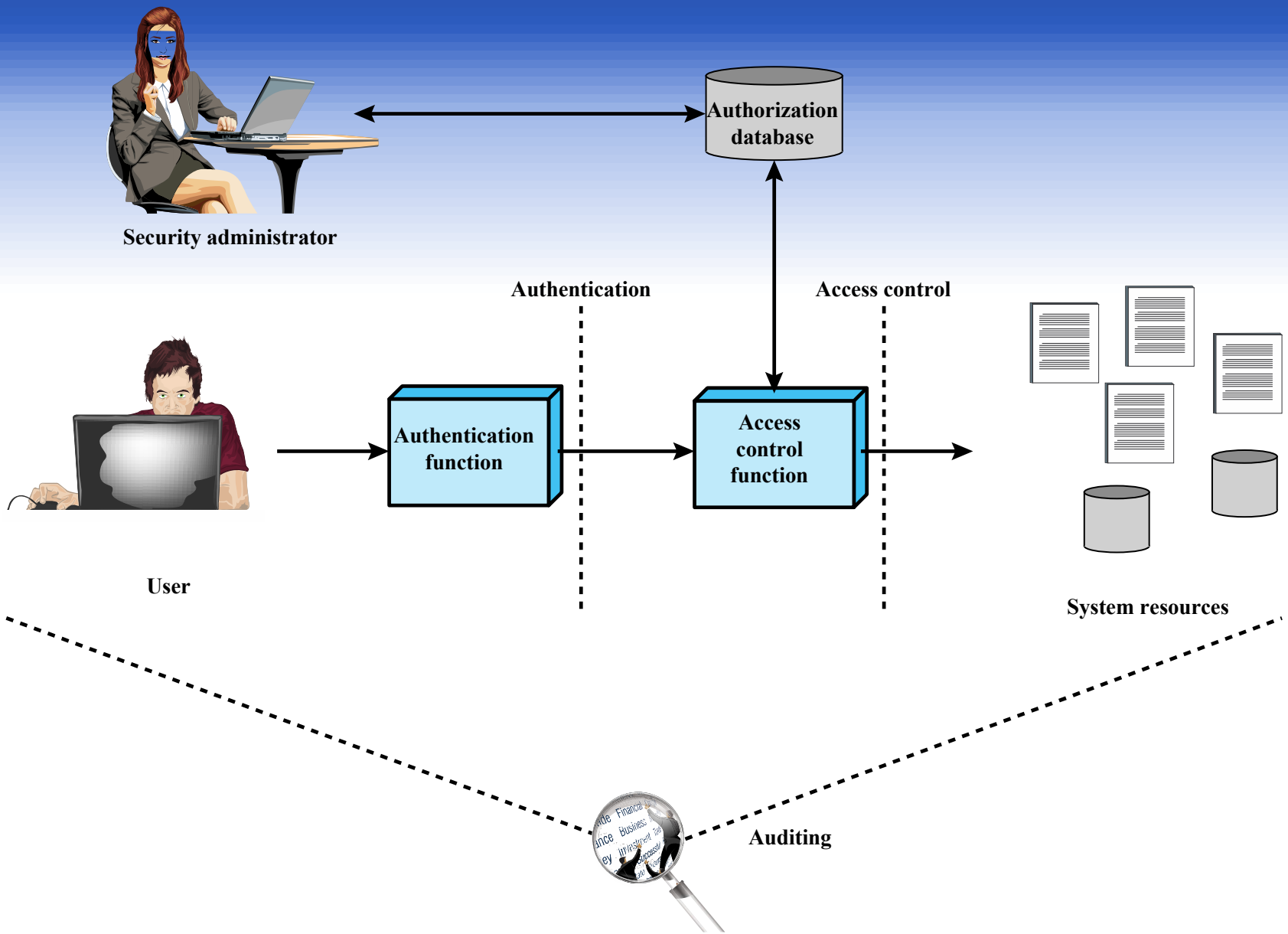
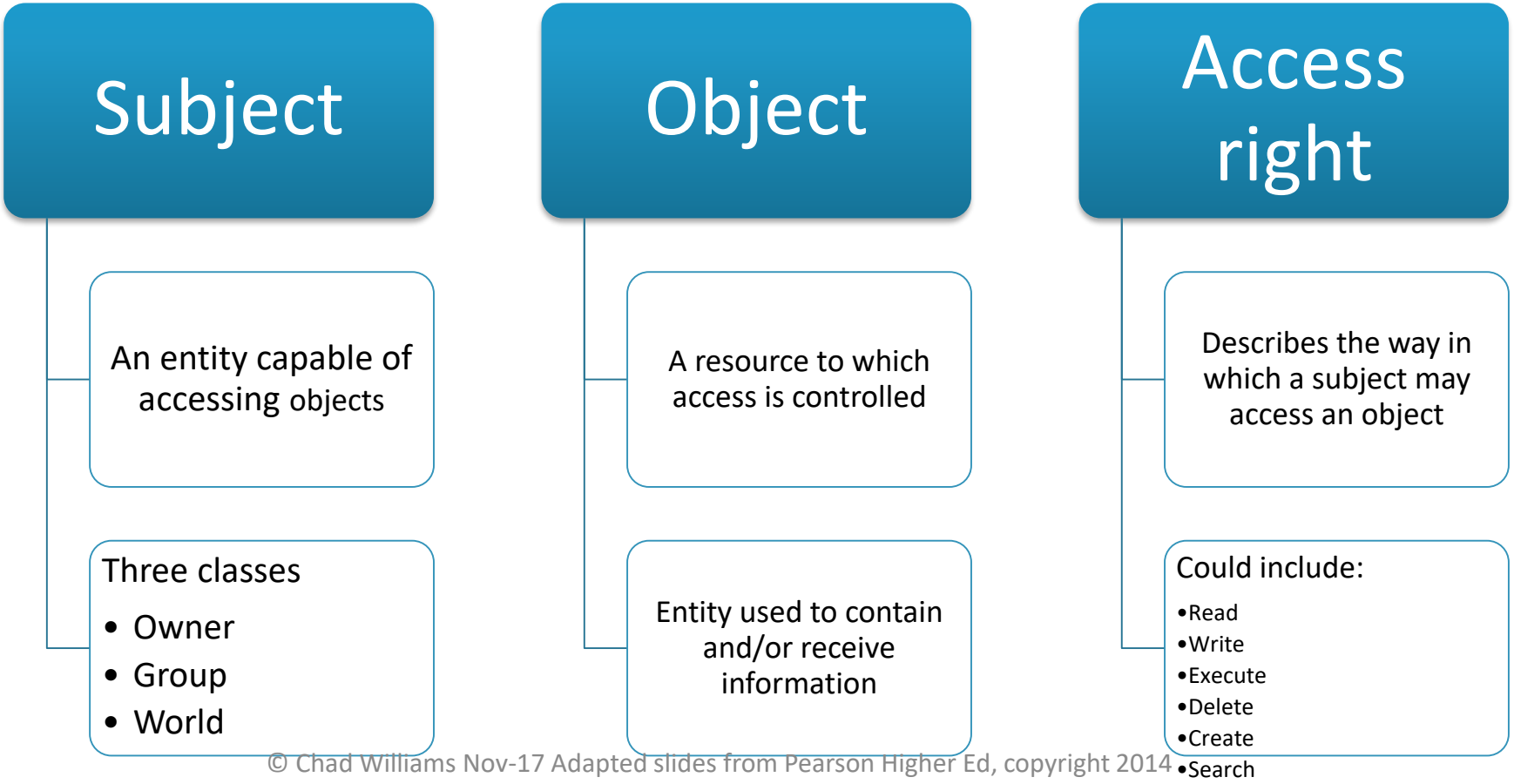


Figure 4.1 Relationship Among Access Control and Other Security Functions

Access Control Policies

- Discretionary access control (DAC)
  - Controls access based on the identity of the requestor and on access rules (authorizations) stating what requestors are (or are not) allowed to do
- Mandatory access control (MAC)
  - Controls access based on comparing security labels with security clearances
- Role-based access control (RBAC)
  - Controls access based on the roles that users have within the system and on rules stating what accesses are allowed to users in given roles
- Attribute-based access control (ABAC)
  - Controls access based on attributes of the user, the resource to be accessed, and current environmental conditions

Subjects, Objects, and Access Rights

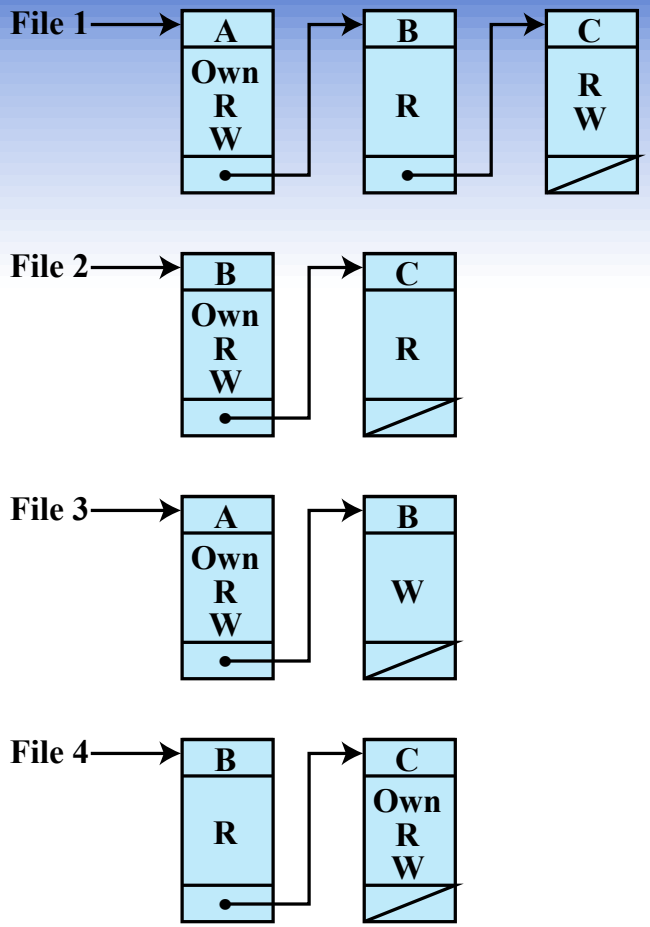


Discretionary Access Control (DAC)

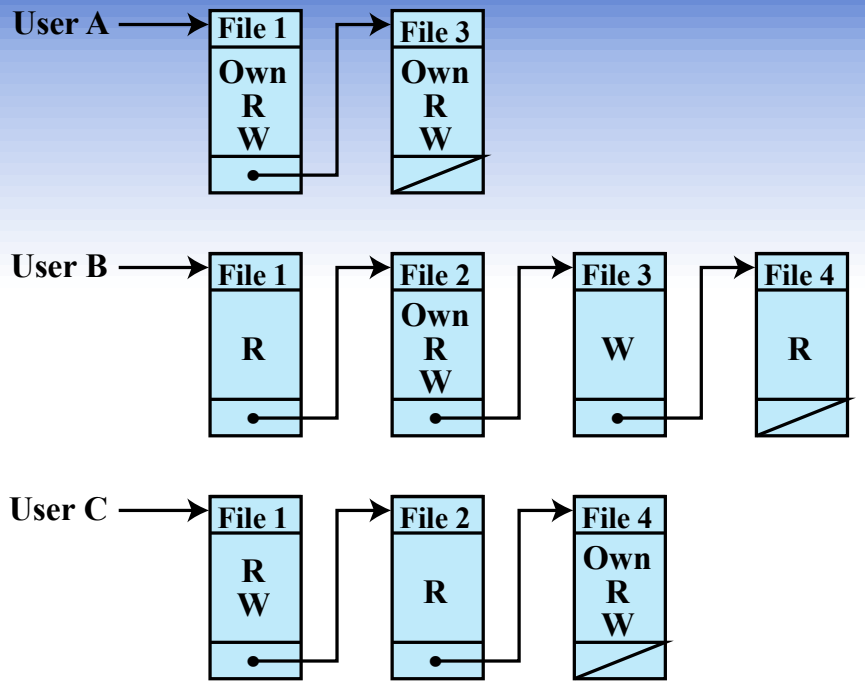
- Scheme in which an entity may enable another entity to access some resource
- Often provided using an access matrix
  - One dimension consists of identified subjects that may attempt data access to the resources
  - The other dimension lists the objects that may be accessed
- Each entry in the matrix indicates the access rights of a particular subject for a particular object

		OBJECTS			
		File 1	File 2	File 3	File 4
SUBJECTS	User A	Own Read Write		Own Read Write	
	User B	Read	Own Read Write	Write	Read
	User C	Read Write	Read		Own Read Write

(a) Access matrix



(b) Access control lists for files of part (a)



(c) Capability lists for files of part (a)

Figure 4.2 Example of Access Control Structures

Subject	Access Mode	Object
A	Own	File 1
A	Read	File 1
A	Write	File 1
A	Own	File 3
A	Read	File 3
A	Write	File 3
B	Read	File 1
B	Own	File 2
B	Read	File 2
B	Write	File 2
B	Write	File 3
B	Read	File 4
C	Read	File 1
C	Write	File 1
C	Read	File 2
C	Own	File 4
C	Read	File 4
C	Write	File 4

Table 4.1

Authorization Table for Files in Figure 4.2

		OBJECTS								
		subjects			files		processes		disk drives	
		S_1	S_2	S_3	F_1	F_2	P_1	P_2	D_1	D_2
SUBJECTS	S_1	control	owner	owner control	read *	read owner	wakeup	wakeup	seek	owner
	S_2		control		write *	execute			owner	seek *
	S_3			control		write	stop			

* - copy flag set

Figure 4.3 Extended Access Control Matrix

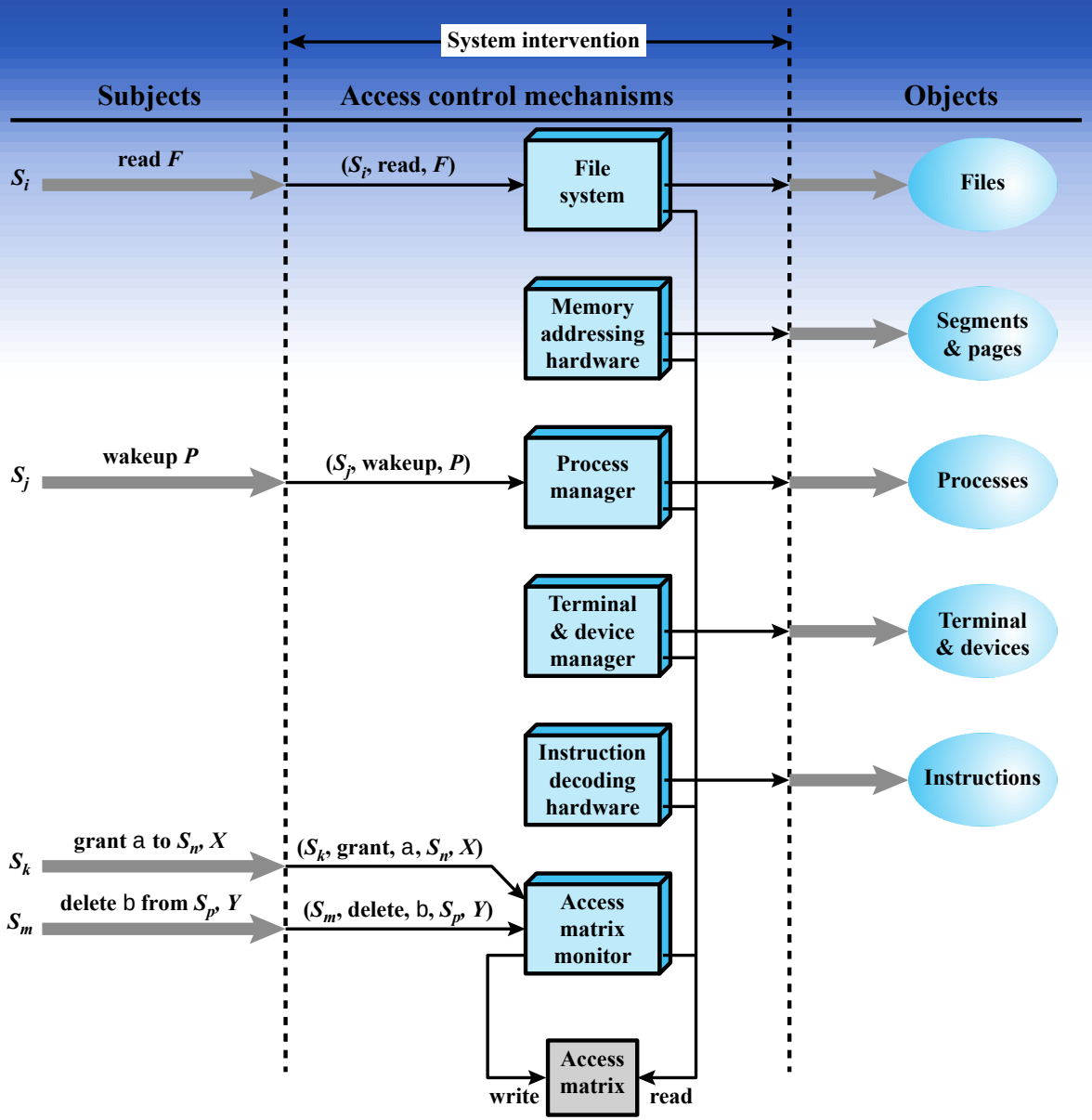
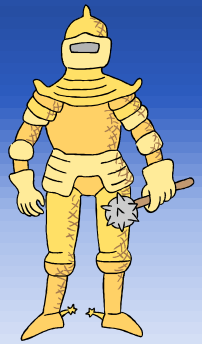


Figure 4.4 An Organization of the Access Control Function

Rule	Command (by S_0)	Authorization	Operation
R1	**transfer** $\begin{Bmatrix}\alpha* \\ \alpha\end{Bmatrix}$ **to** S, X	'α*' in $A[S_0, X]$	store $\begin{Bmatrix}\alpha* \\ \alpha\end{Bmatrix}$ in $A[S, X]$
R2	**grant** $\begin{Bmatrix}\alpha* \\ \alpha\end{Bmatrix}$ **to** S, X	'owner' in $A[S_0, X]$	store $\begin{Bmatrix}\alpha* \\ \alpha\end{Bmatrix}$ in $A[S, X]$
R3	**delete** α **from** S, X	'control' in $A[S_0, S]$ or 'owner' in $A[S_0, X]$	delete α from $A[S, X]$
R4	$w \leftarrow$ **read** S, X	'control' in $A[S_0, S]$ or 'owner' in $A[S_0, X]$	copy $A[S, X]$ into w
R5	**create object** X	None	add column for X to A; store 'owner' in $A[S_0, X]$
R6	**destroy object** X	'owner' in $A[S_0, X]$	delete column for X from A
R7	**create subject** S	none	add row for S to A; execute **create object** S; store 'control' in $A[S, S]$
R8	**destroy subject** S	'owner' in $A[S_0, S]$	delete row for S from A; execute **destroy object** S

Table 4.2

Access Control System Commands



Protection Domains

- Set of objects together with access rights to those objects
- More flexibility when associating capabilities with protection domains
- In terms of the access matrix, a row defines a protection domain
- User can spawn processes with a subset of the access rights of the user
- Association between a process and a domain can be static or dynamic
- In user mode certain areas of memory are protected from use and certain instructions may not be executed
- In kernel mode privileged instructions may be executed and protected areas of memory may be accessed

UNIX File Access Control

UNIX files are administered using inodes (index nodes)

- Control structures with key information needed for a particular file
- Several file names may be associated with a single inode
- An active inode is associated with exactly one file
- File attributes, permissions and control information are sorted in the inode
- On the disk there is an inode table, or inode list, that contains the inodes of all the files in the file system
- When a file is opened its inode is brought into main memory and stored in a memory resident inode table

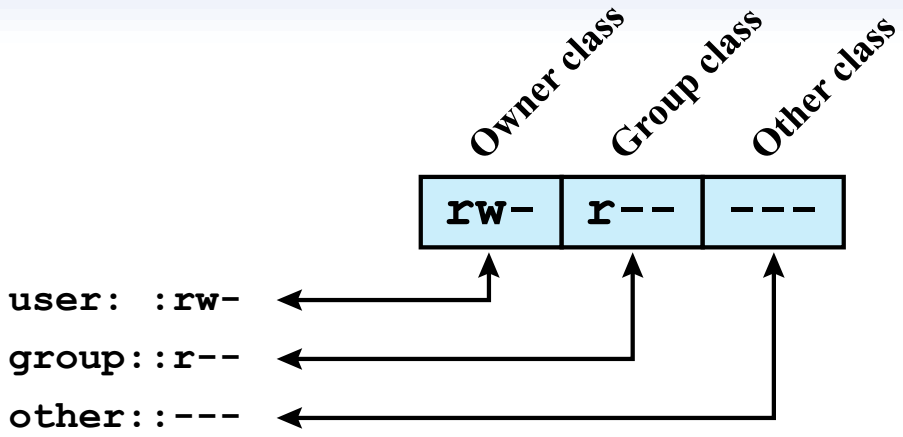
Directories are structured in a hierarchical tree

- May contain files and/or other directories
- Contains file names plus pointers to associated inodes

UNIX

File Access Control

- Unique user identification number (user ID)
- Member of a primary group identified by a group ID
- Belongs to a specific group
- 12 protection bits
  - Specify read, write, and execute permission for the owner of the file, members of the group and all other users
- The owner ID, group ID, and protection bits are part of the file's inode



(a) Traditional UNIX approach (minimal access control list)

Traditional UNIX File Access Control

- "Set user ID"(SetUID)
- "Set group ID"(SetGID)
  - System temporarily uses rights of the file owner/group in addition to the real user's rights when making access control decisions
  - Enables privileged programs to access files/resources not generally accessible
- Sticky bit
  - When applied to a directory it specifies that only the owner of any file in the directory can rename, move, or delete that file
- Superuser
  - Is exempt from usual access control restrictions
  - Has system-wide access

Access Control Lists (ACLs) in UNIX

Modern UNIX systems support ACLs

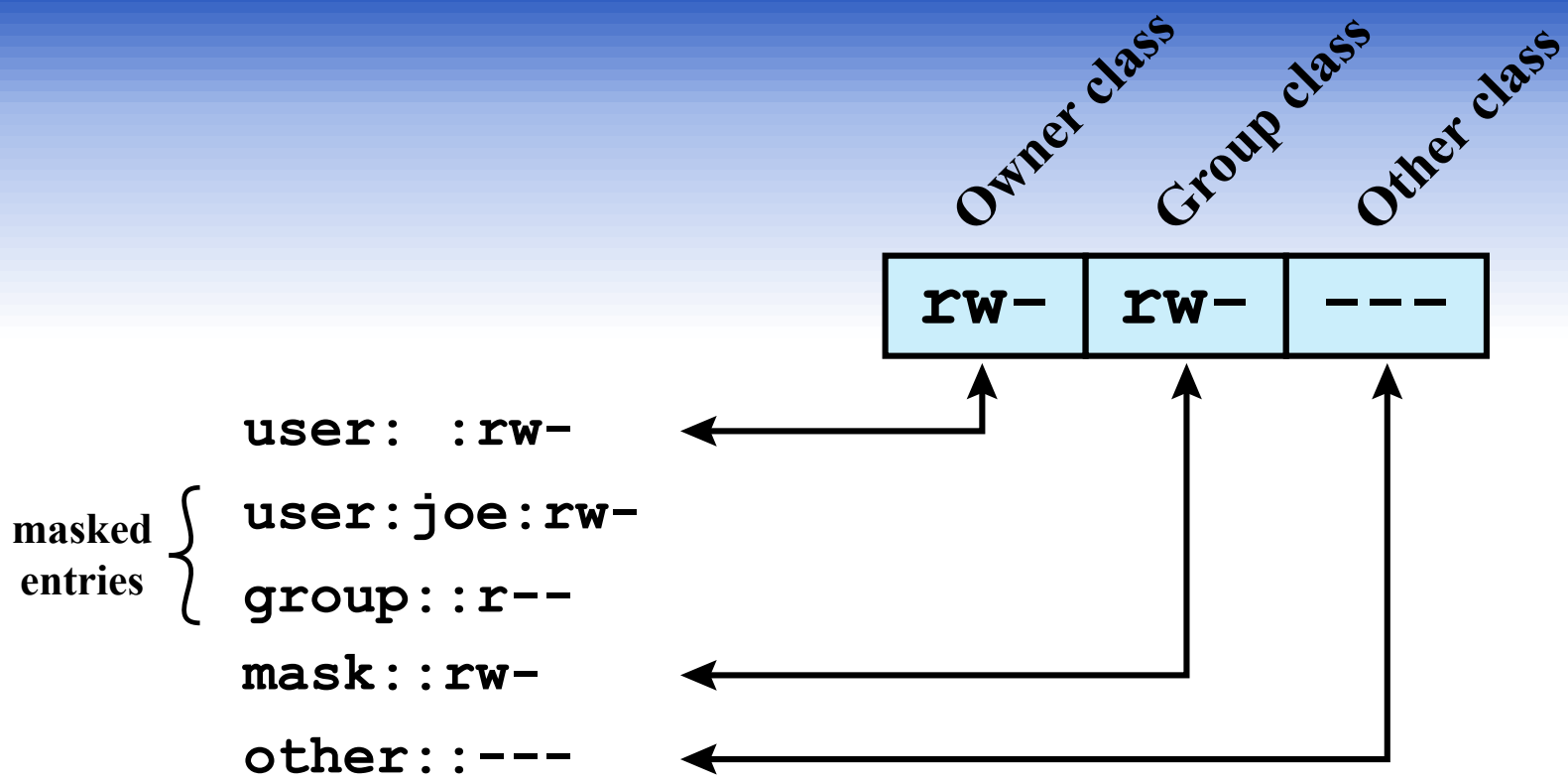
- FreeBSD, OpenBSD, Linux, Solaris

FreeBSD

- Setfacl command assigns a list of UNIX user IDs and groups
- Any number of users and groups can be associated with a file
- Read, write, execute protection bits
- A file does not need to have an ACL
- Includes an additional protection bit that indicates whether the file has an extended ACL

When a process requests access to a file system object two steps are performed:

- Step 1 selects the most appropriate ACL
- Step 2 checks if the matching entry contains sufficient permissions



(b) Extended access control list

Figure 4.5 UNIX File Access Control

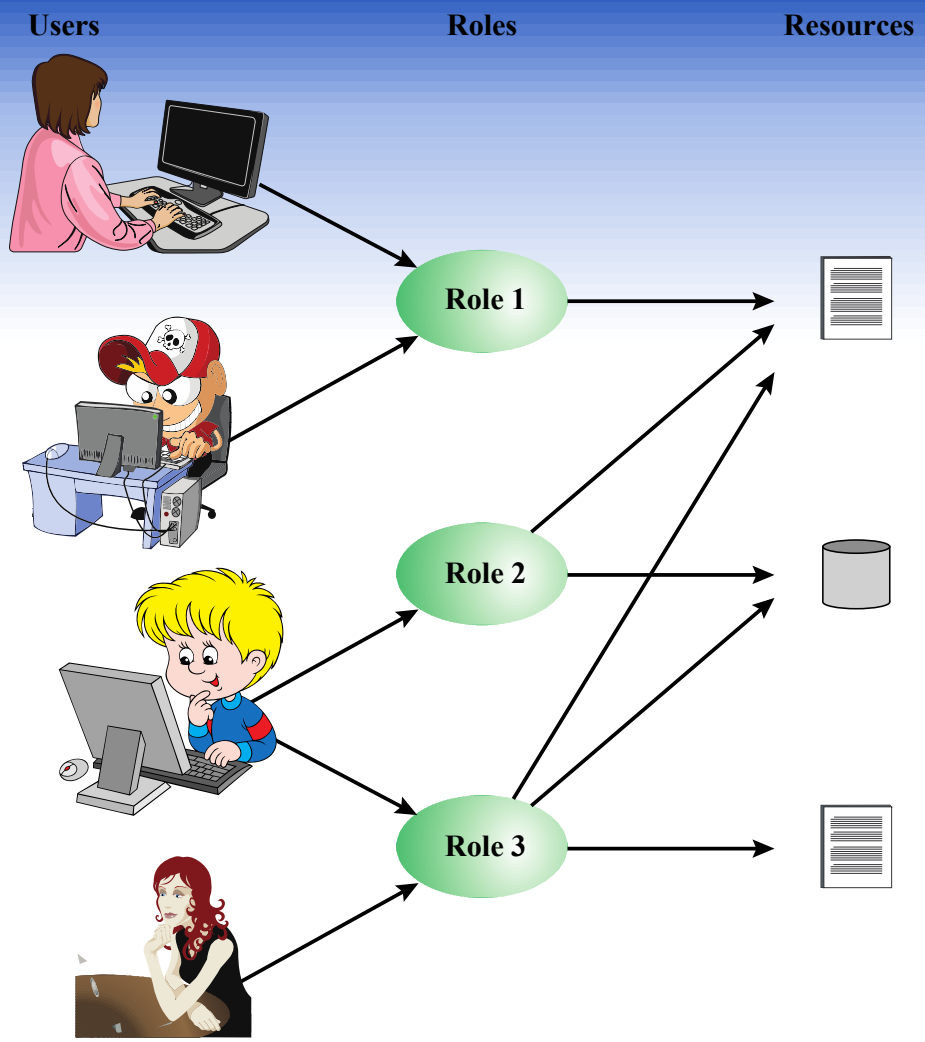
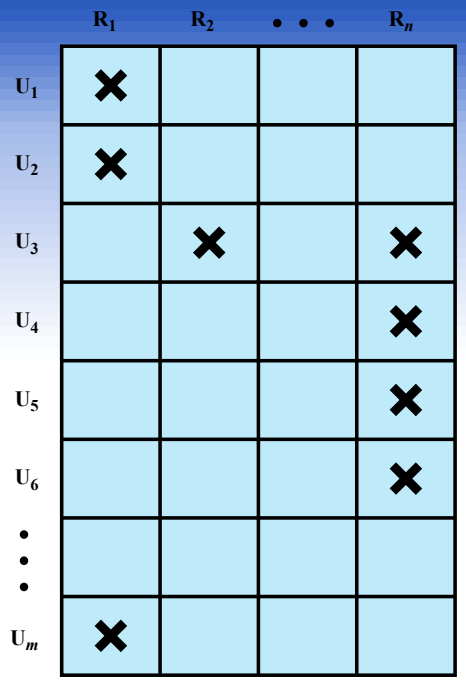


Figure 4.6 Users, Roles, and Resources



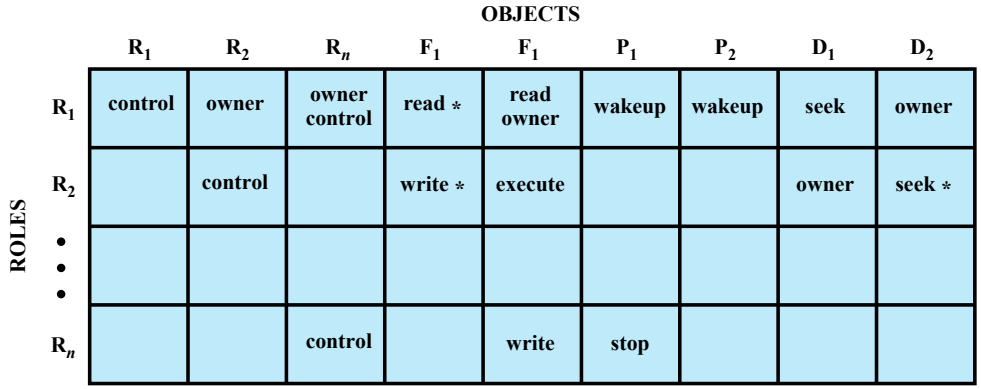
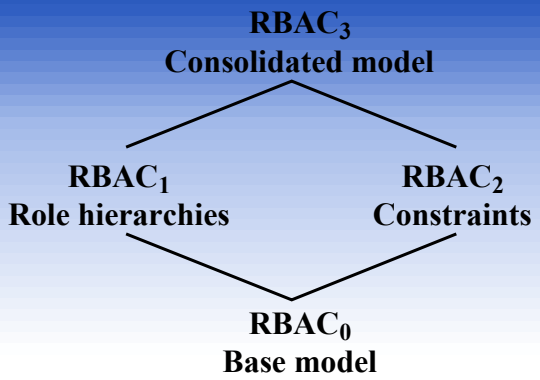
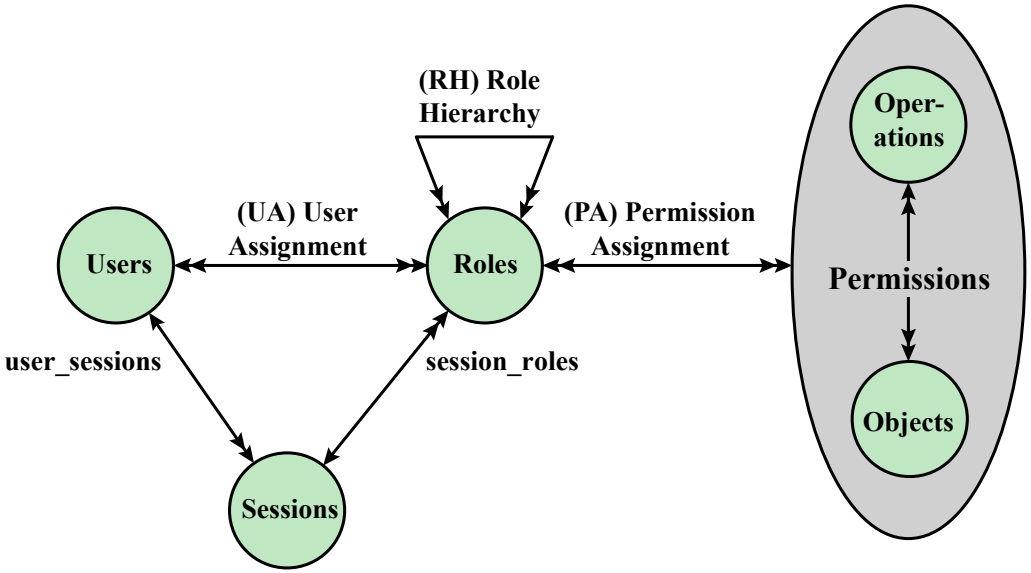


Figure 4.7 Access Control Matrix Representation of RBAC



(a) Relationship among RBAC models



(b) RBAC models

Figure 4.8 A Family of Role-Based Access Control Models.

Table 4.3
Scope RBAC Models

Models	Hierarchies	Constraints
$RBAC_0$	No	No
$RBAC_1$	Yes	No
$RBAC_2$	No	Yes
$RBAC_3$	Yes	Yes

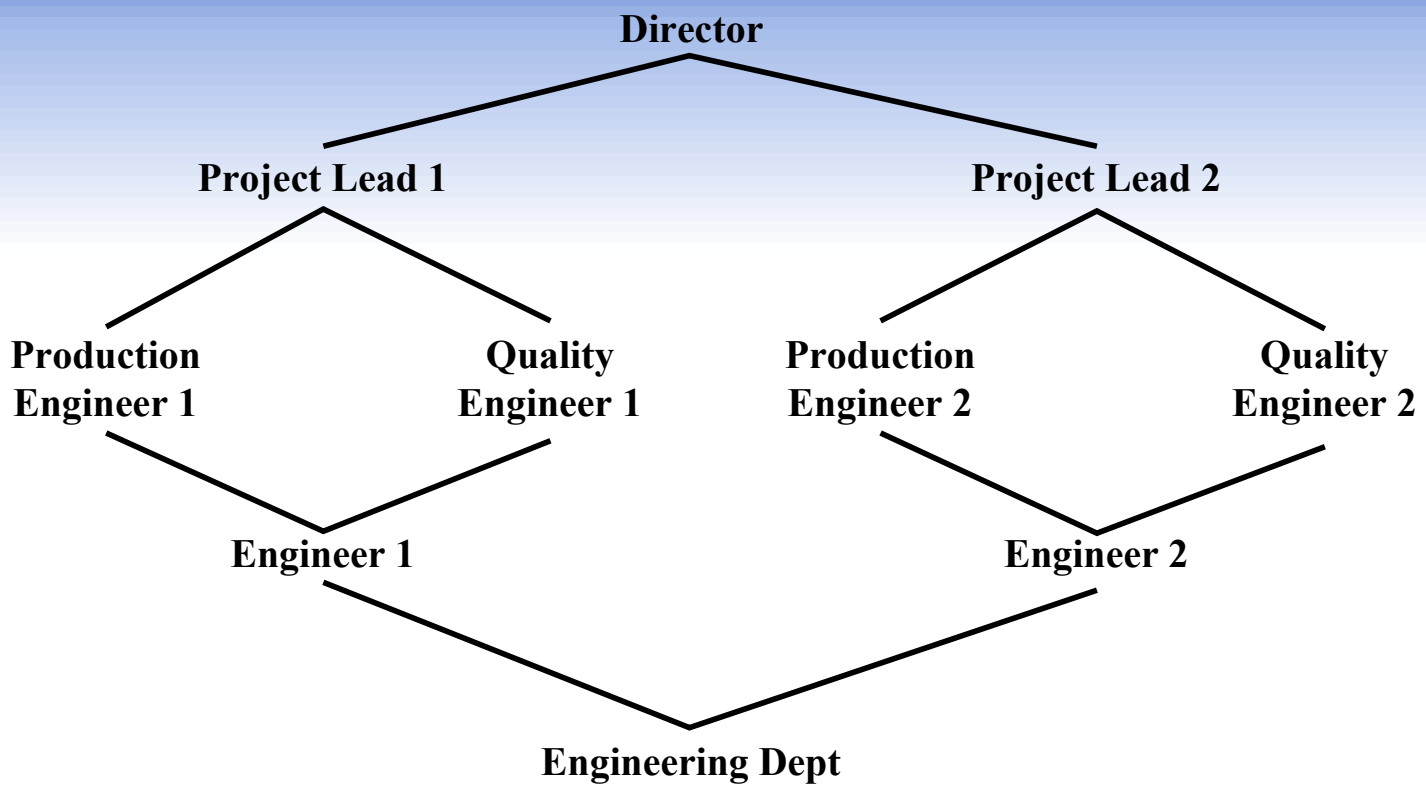


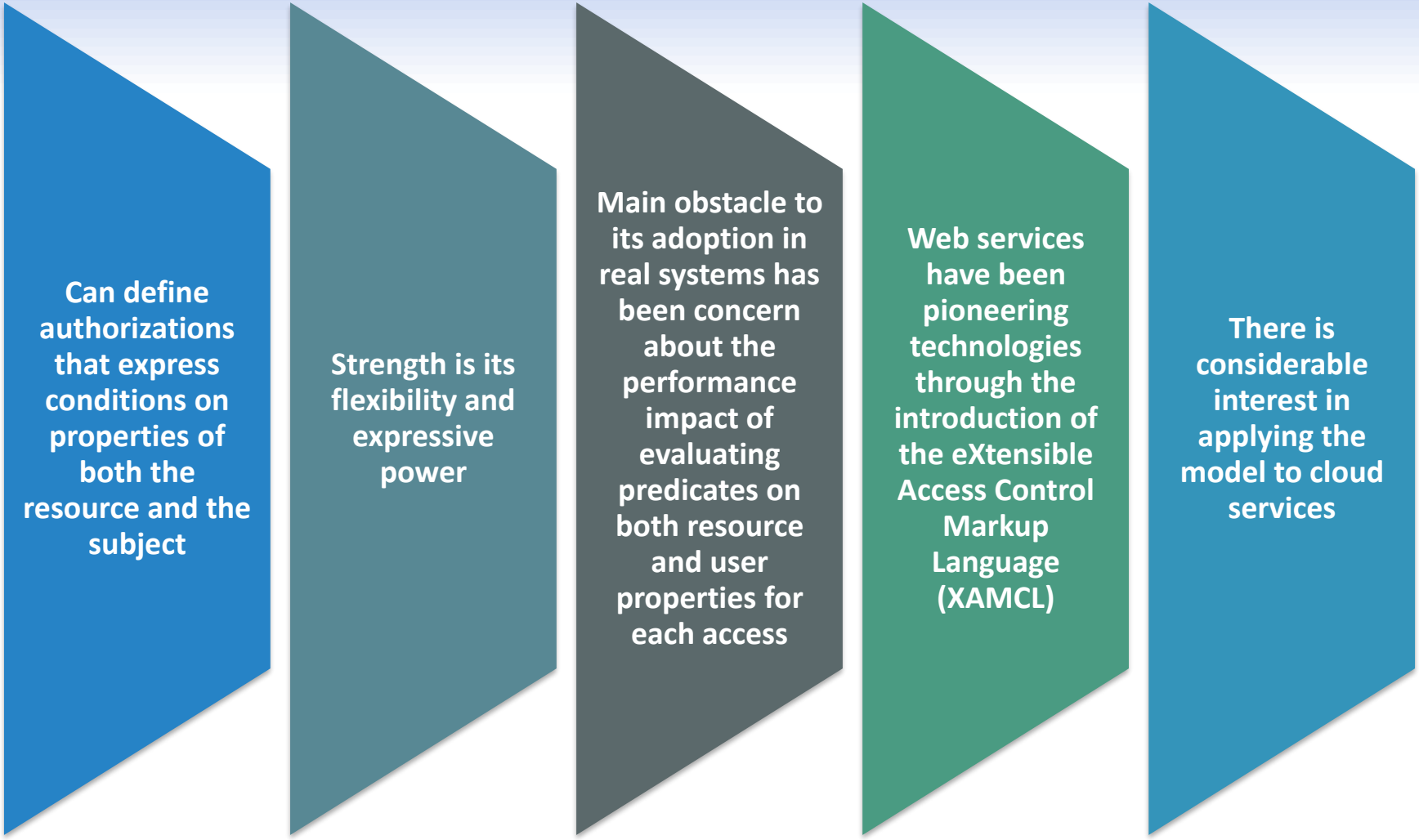
Figure 4.9 Example of Role Hierarchy

Constraints - RBAC

- Provide a means of adapting RBAC to the specifics of administrative and security policies of an organization
- A defined relationship among roles or a condition related to roles
- Types:

Mutually exclusive roles	Cardinality	Prerequisite roles
• A user can only be assigned to one role in the set (either during a session or statically) • Any permission (access right) can be granted to only one role in the set	• Setting a maximum number with respect to roles	• Dictates that a user can only be assigned to a particular role if it is already assigned to some other specified role

Attribute-Based Access Control (ABAC)



ABAC Model: Attributes

Subject attributes

- A subject is an active entity that causes information to flow among objects or changes the system state
- Attributes define the identity and characteristics of the subject

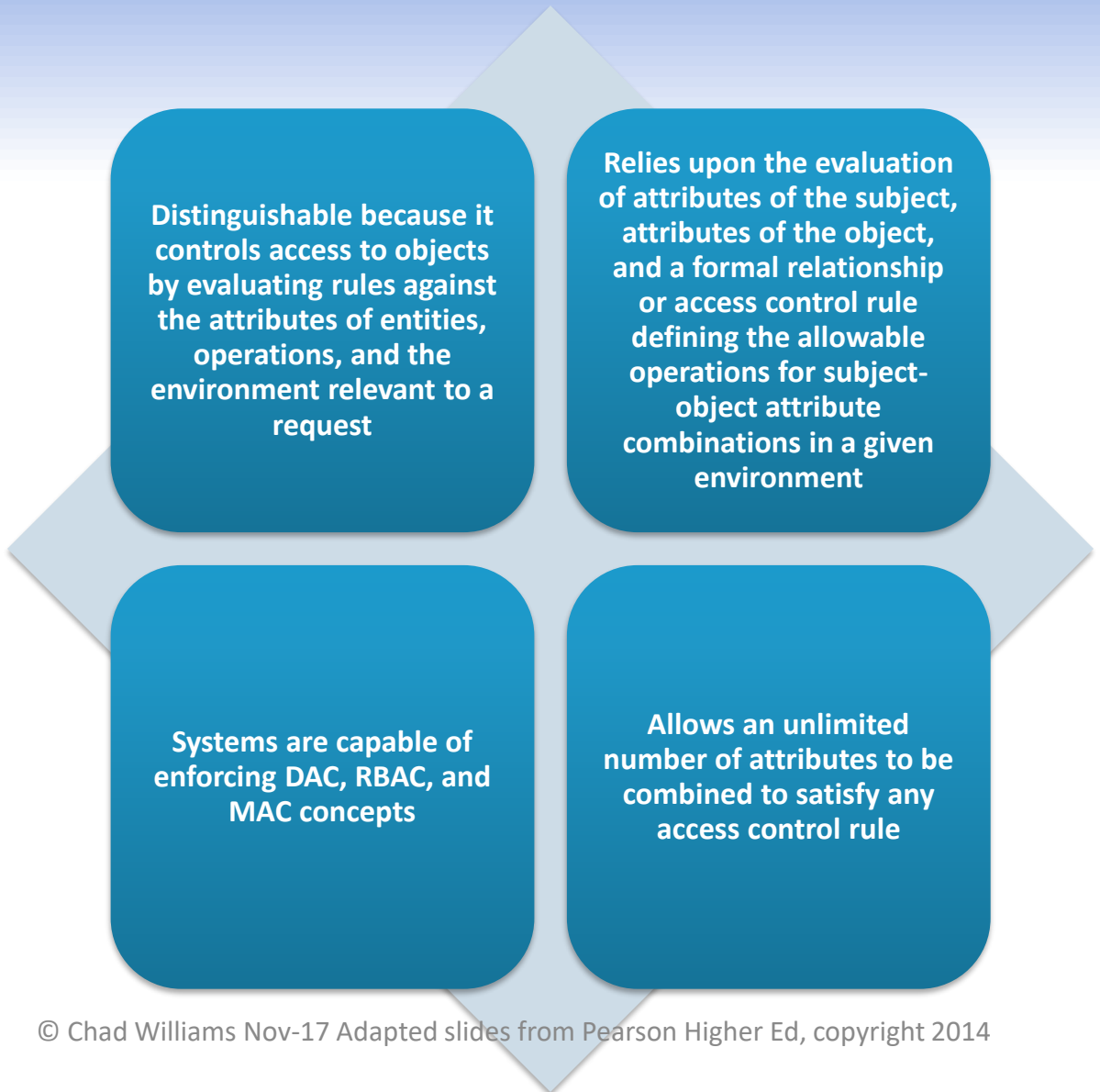
Object attributes

- An object (or resource) is a passive information system-related entity containing or receiving information
- Objects have attributes that can be leverages to make access control decisions

Environment attributes

- Describe the operational, technical, and even situational environment or context in which the information access occurs
- These attributes have so far been largely ignored in most access control policies

ABAC



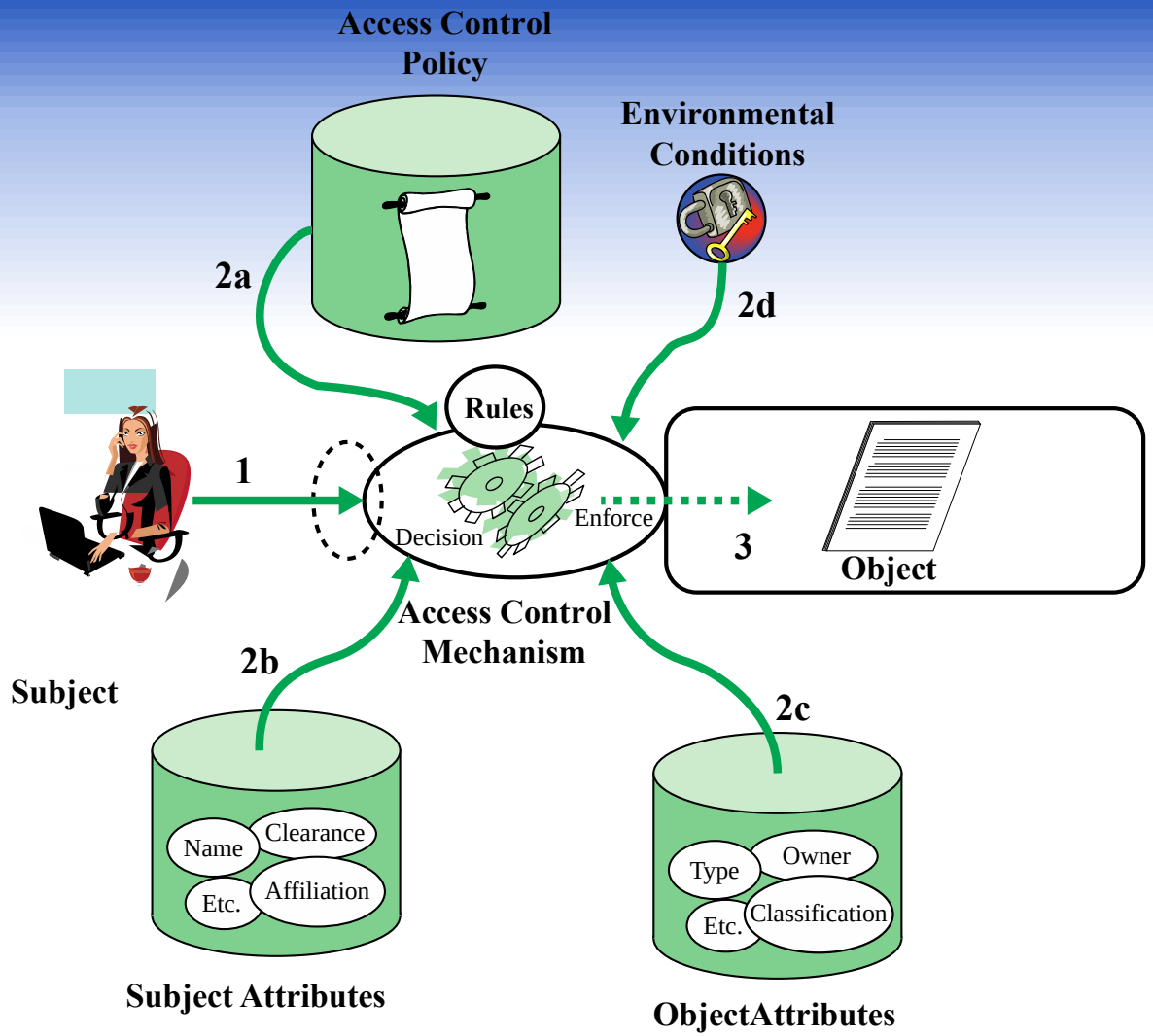
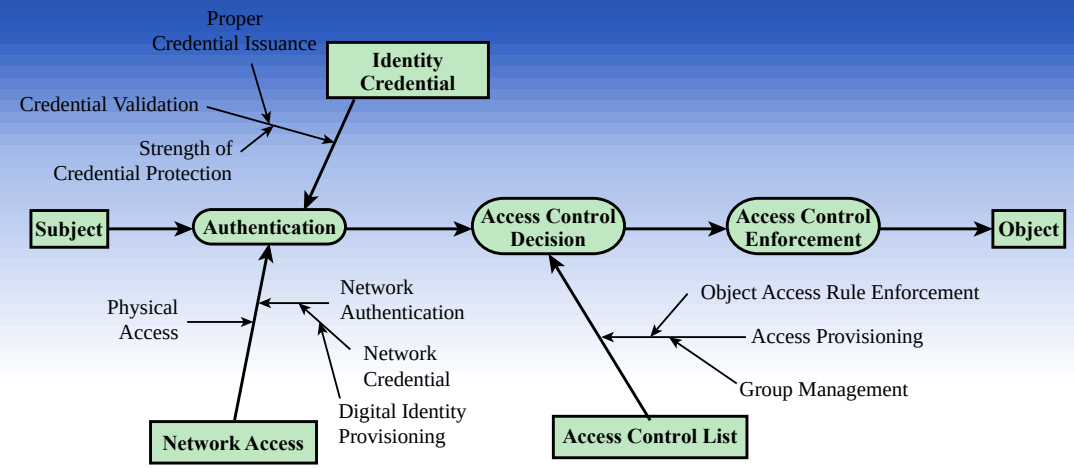
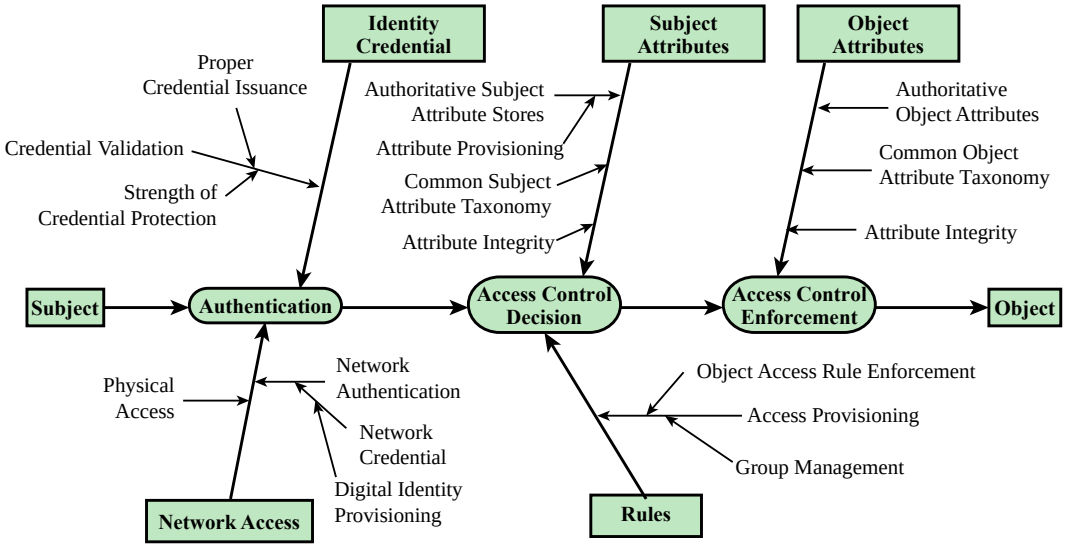


Figure 4.10 Simple ABAC Scenario



(a) ACL Trust Chain



(b) ABAC Trust Chain

Figure 4.11 ACL and ABAC Trust Relationships

ABAC Policies

A policy is a set of rules and relationships that govern allowable behavior within an organization, based on the privileges of subjects and how resources or objects are to be protected under which environment conditions

Typically written from the perspective of the object that needs protecting and the privileges available to subjects

Privileges represent the authorized behavior of a subject and are defined by an authority and embodied in a policy

Other terms commonly used instead of privileges are: rights, authorizations, and entitlements

Identity, Credential, and Access Management (ICAM)

- A comprehensive approach to managing and implementing digital identities, credentials, and access control
- Developed by the U.S. government
- Designed to:
  - Create trusted digital identity representations of individuals and nonperson entities (NPEs)
  - Bind those identities to credentials that may serve as a proxy for the individual of NPE in access transactions
    - A credential is an object or data structure that authoritatively binds an identity to a token possessed and controlled by a subscriber
  - Use the credentials to provide authorized access to an agency's resources

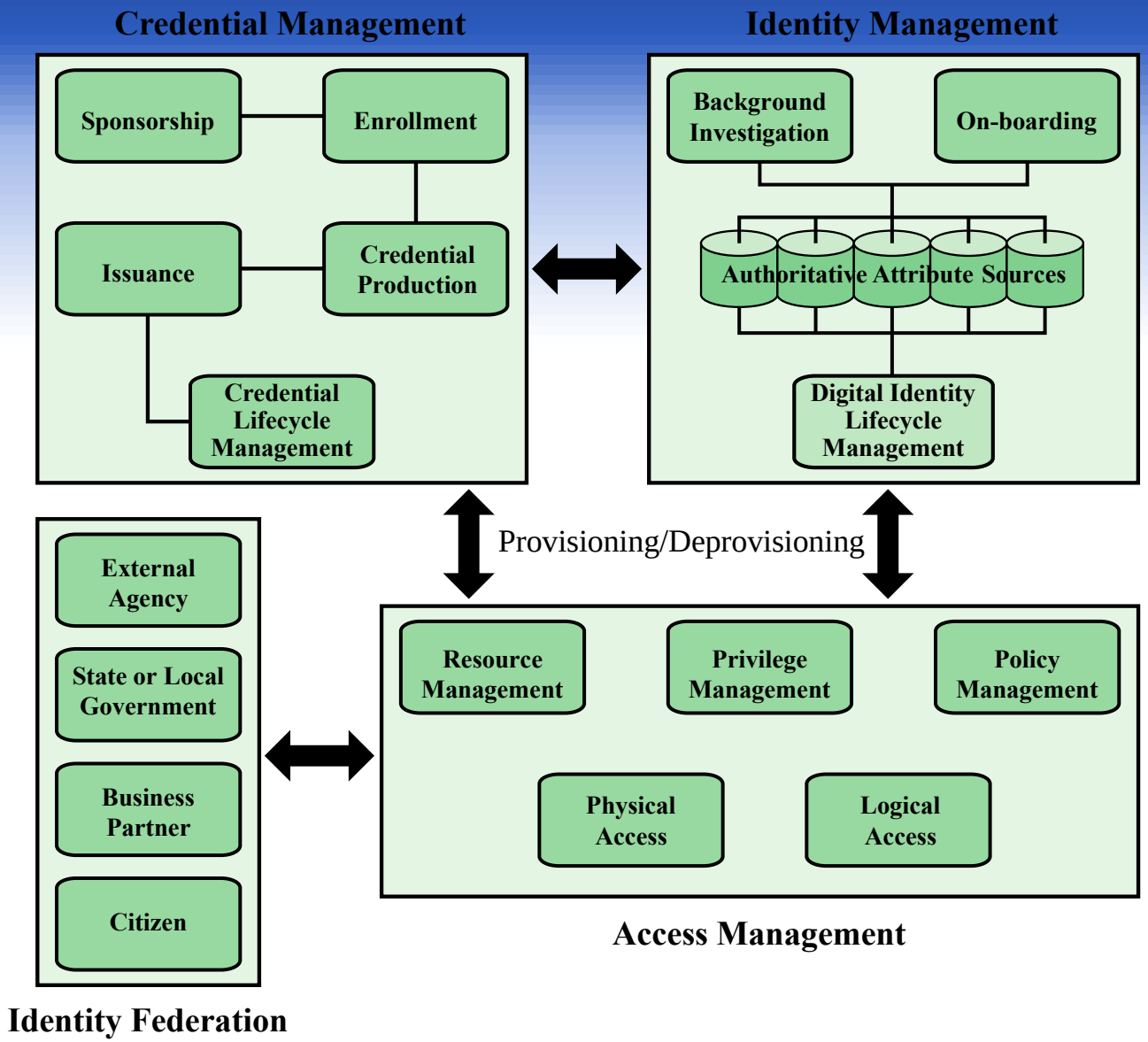


Figure 4.12 Identity, Credential, and Access Management (ICAM)

Credential Management

The management of the life cycle of the credential

Examples of credentials are smart cards, private/public cryptographic keys, and digital certificates

Encompasses five logical components:

An authorized individual sponsors an individual or entity for a credential to establish the need for the credential

The sponsored individual enrolls for the credential

- Process typically consists of identity proofing and the capture of biographic and biometric data
- This step may also involve incorporating authoritative attribute data, maintained by the identity management component

A credential is produced

- Depending on the credential type, production may involve encryption, the use of a digital signature, the production of a smart card or other functions

The credential is issued to the individual or NPE

A credential must be maintained over its life cycle

- Might include revocation, reissuance/replacement, reenrollment, expiration, personal identification number (PIN) reset, suspension, or reinstatement

Access Management

Deals with the management and control of the ways entities are granted access to resources

Covers both logical and physical access

May be internal to a system or an external element

Purpose is to ensure that the proper identity verification is made when an individual attempts to access a security sensitive building, computer systems, or data

Three support elements are needed for an enterprise-wide access control facility:

- Resource management
- Privilege management
- Policy management

Three support elements are needed for an enterprise-wide access control facility:

Resource management

- Concerned with defining rules for a resource that requires access control
- Rules would include credential requirements and what user attributes, resource attributes, and environmental conditions are required for access of a given resource for a given function

Privilege management

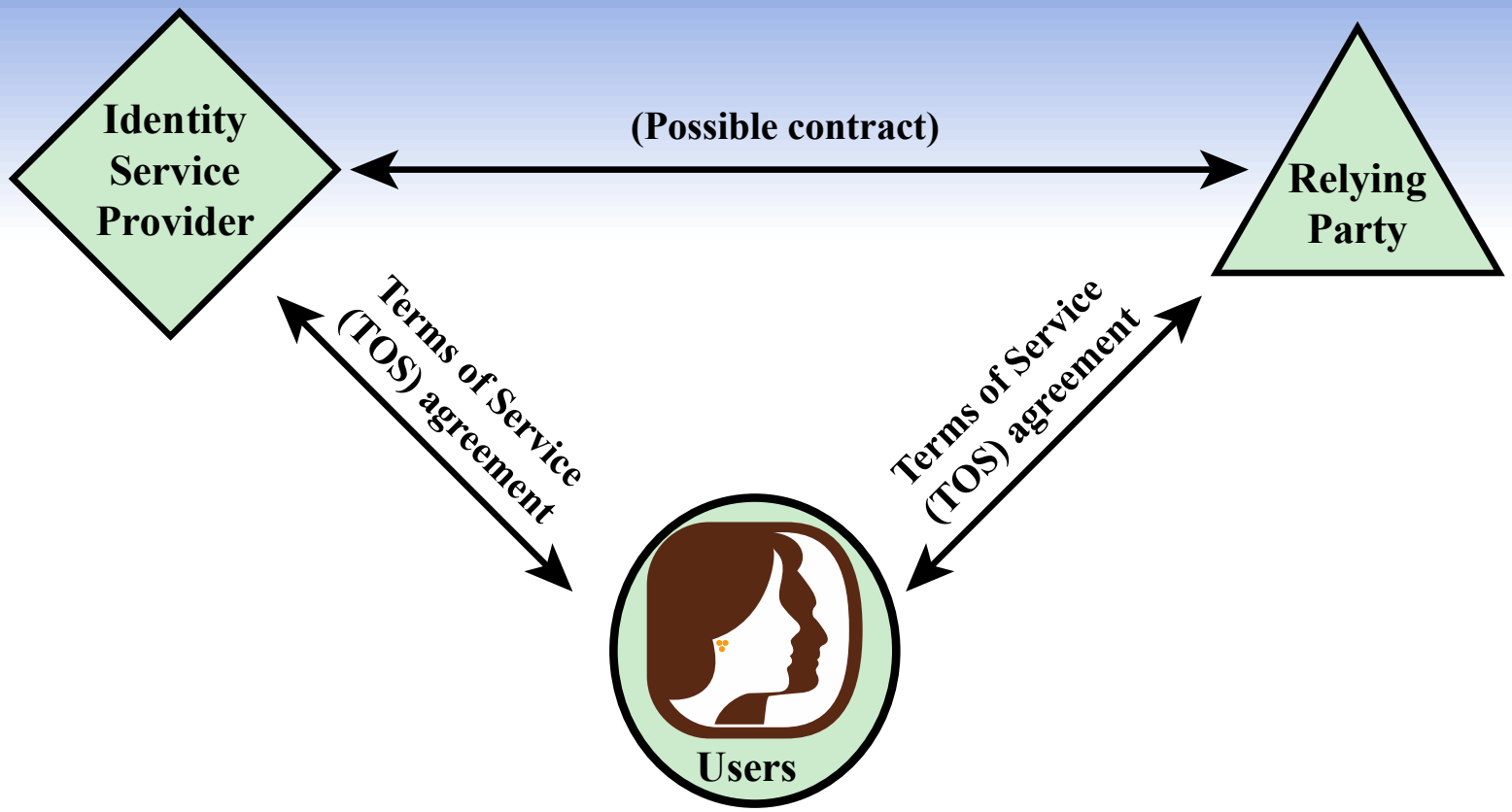
- Concerned with establishing and maintaining the entitlement or privilege attributes that comprise an individual's access profile
- These attributes represent features of an individual that can be used as the basis for determining access decisions to both physical and logical resources
- Privileges are considered attributes that can be linked to a digital identity

Policy management

- Governs what is allowable and unallowable in an access transaction

Identity Federation

- Term used to describe the technology, standards, policies, and processes that allow an organization to trust digital identities, identity attributes, and credentials created and issued by another organization
- Addresses two questions:
  - How do you trust identities of individuals from external organizations who need access to your systems
  - How do you vouch for identities of individuals in your organization when they need to collaborate with external organizations



(a) Traditional triangle of parties involved in an exchange of identity information

Figure 4.13 Identity Information Exchange Approaches

Open Identity Trust Framework

OpenID

- An open standard that allows users to be authenticated by certain cooperating sites using a third party service

OIDF

- OpenID Foundation is an international nonprofit organization of individuals and companies committed to enabling, promoting, and protecting OpenID technologies

ICF

- Information Card Foundation is a nonprofit community of companies and individuals working together to evolve the Information Card ecosystem

OITF

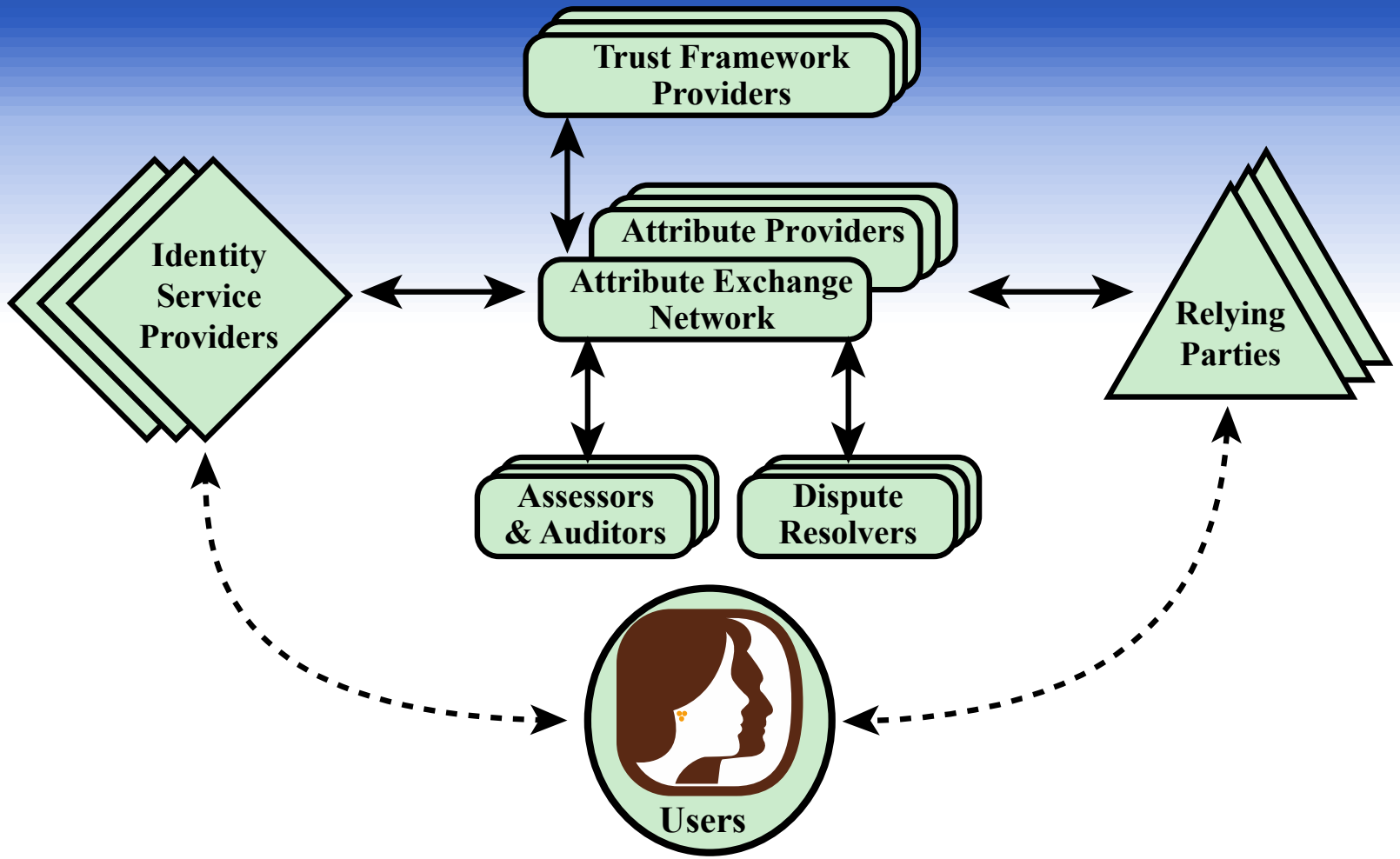
- Open Identity Trust Framework is a standardized, open specification of a trust framework for identity and attribute exchange, developed jointly by OIDF and ICF

OIX

- Open Identity Exchange Corporation is an independent, neutral, international provider of certification trust frameworks conforming to the OITF model

AXN

- Attribute Exchange Network is an online Internet-scale gateway for identity service providers and relying parties to efficiently access user asserted, permissioned, and verified online identity attributes in high volumes at affordable costs



(B) Identity attribute exchange elements

Figure 4.13 Identity Information Exchange Approaches

Table 4.4
Functions and Roles for Banking Example

(a) Functions and Official Positions

Role	Function	Official Position
A	financial analyst	Clerk
B	financial analyst	Group Manager
C	financial analyst	Head of Division
D	financial analyst	Junior
E	financial analyst	Senior
F	financial analyst	Specialist
G	financial analyst	Assistant
• • •	• • •	• • •
X	share technician	Clerk
Y	support e-commerce	Junior
Z	office banking	Head of Division

Table 4.4
Functions and Roles for Banking Example

(b) Permission Assignments

Role	Application	Access Right
A	money market instruments	1, 2, 3, 4
	derivatives trading	1, 2, 3, 7, 10, 12
	interest instruments	1, 4, 8, 12, 14, 16
B	money market instruments	1, 2, 3, 4, 7
	derivatives trading	1, 2, 3, 7, 10, 12, 14
	interest instruments	1, 4, 8, 12, 14, 16
	private consumer instruments	1, 2, 4, 7
• • •	• • •	• • •

(c) PA with Inheritance

Role	Application	Access Right
A	money market instruments	1, 2, 3, 4
	derivatives trading	1, 2, 3, 7, 10, 12
	interest instruments	1, 4, 8, 12, 14, 16
B	money market instruments	7
	derivatives trading	14
	private consumer instruments	1, 2, 4, 7
• • •	• • •	• • •

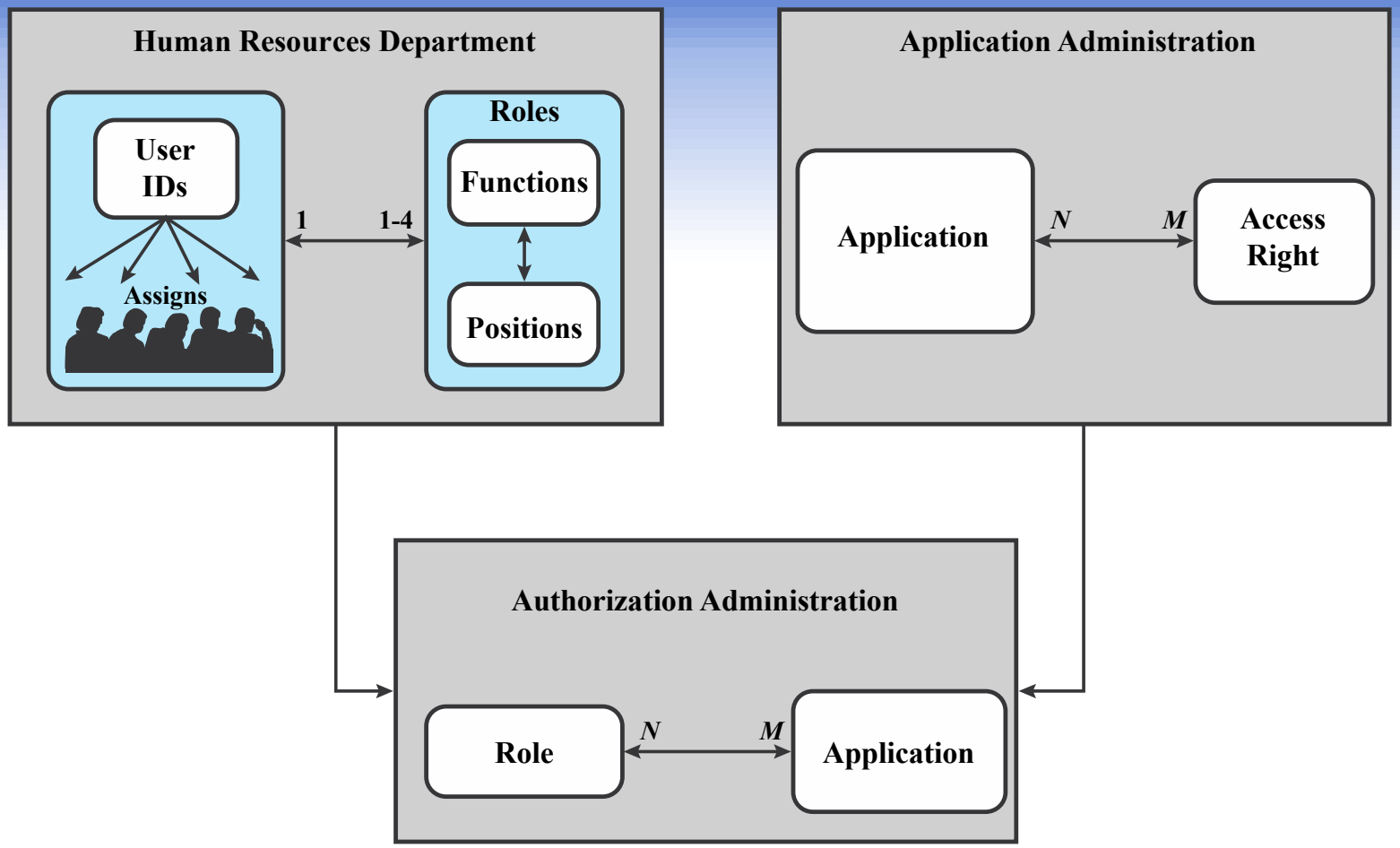


Figure 4.14 Example of Access Control Administration

Summary

- Access control principles
  - Access control context
  - Access control policies
- Subjects, objects, and access rights
- Discretionary access control
  - Access control model
  - Protection domains
- UNIX file access control
  - Traditional UNIX file access control
  - Access control lists in UNIX
- Role-based access control
  - RBAC reference models

- Attribute-based access control
  - Attributes
  - ABAC logical architecture
  - ABAC policies
- Identity, credential, and access management
  - Identity management
  - Credential management
  - Access management
  - Identity federation
- Trust frameworks
  - Traditional identity exchange approach
  - Open identity trust framework
- Bank RBAC system

In class exercises revisited

Bob's Pizza Shack

In groups consider this scenario

Bob is a small business owner of Bob's Pizza Shack and wants to create a website to allow online credit card delivery orders

- Identify the languages needed for implementing the case project.
  - What are the known vulnerabilities of these languages and where does the system provide opportunity for misuse?

Alice's Online Bank

In groups consider this scenario

- Alice opens Alice's Online Bank (AOB)
  - Internal use bank employees
  - Interoperates with another bank
  - Interacts with customers
    - Web
    - Mobile app
- Identify the languages needed for implementing the case project.
  - What are the known vulnerabilities of these languages and where does the system provide opportunity for misuse?

Location based social media app

In groups consider this scenario Open source group wants to create a mobile app to allow groups to communicate/find each other in public demonstrations/protests

- Communication internet, as well as, P2P (WiFi/Bluetooth) in case internet cut off – so if person you want to contact is on other side of crowd and no internet as long as P2P network can be established with app can reach somebody outside your immediate vicinity via the P2P network
- Should be able to communicate securely messages and images to people you identify within your group (group as whole or direct)
- Should be able to share GPS location with people in group (group as whole or direct)