

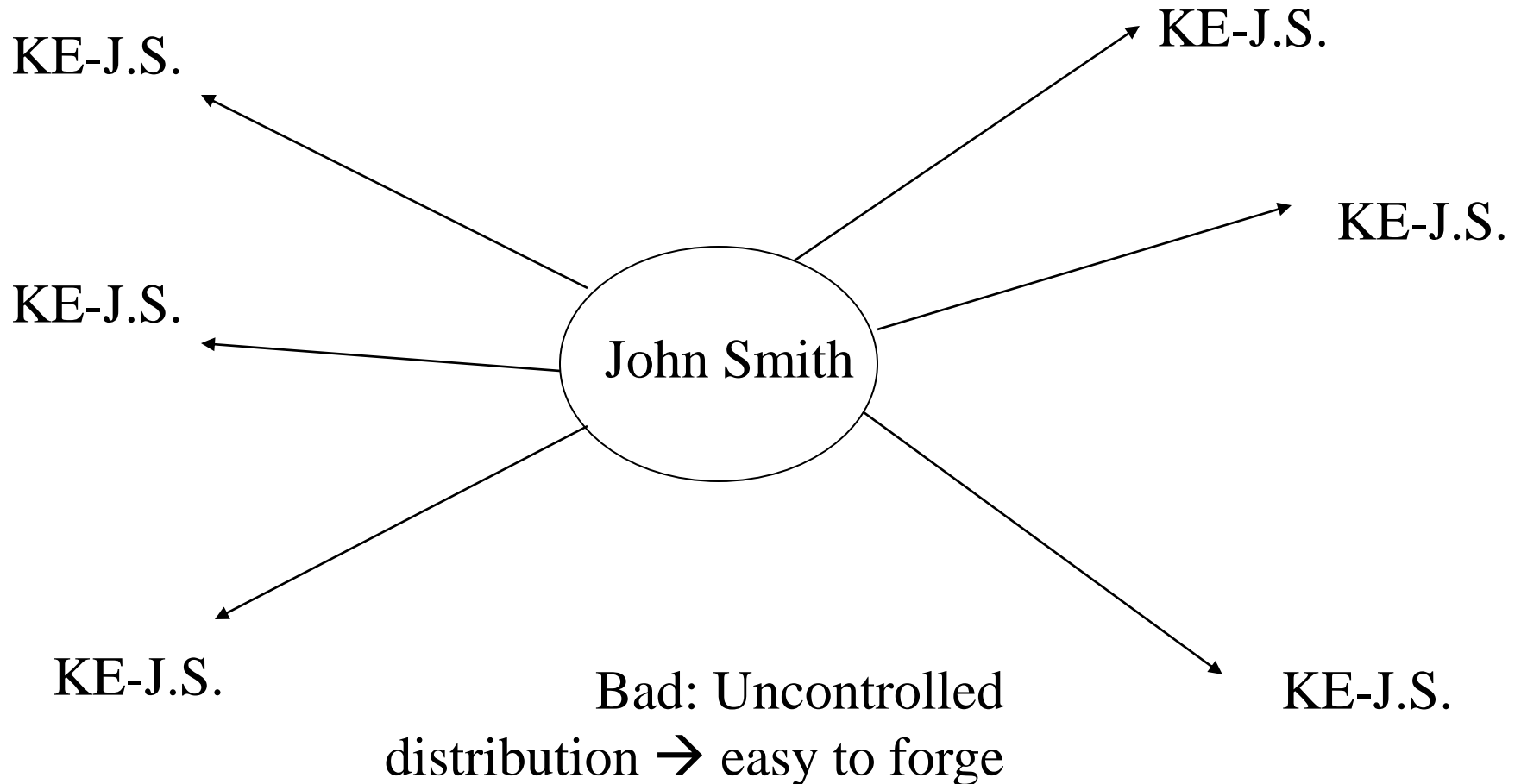
Symmetric-Key Distribution: Symmetric-Key Techniques

- Symmetric-Key without Server
- Symmetric-Key with Server

Asymmetric-Key Exchange

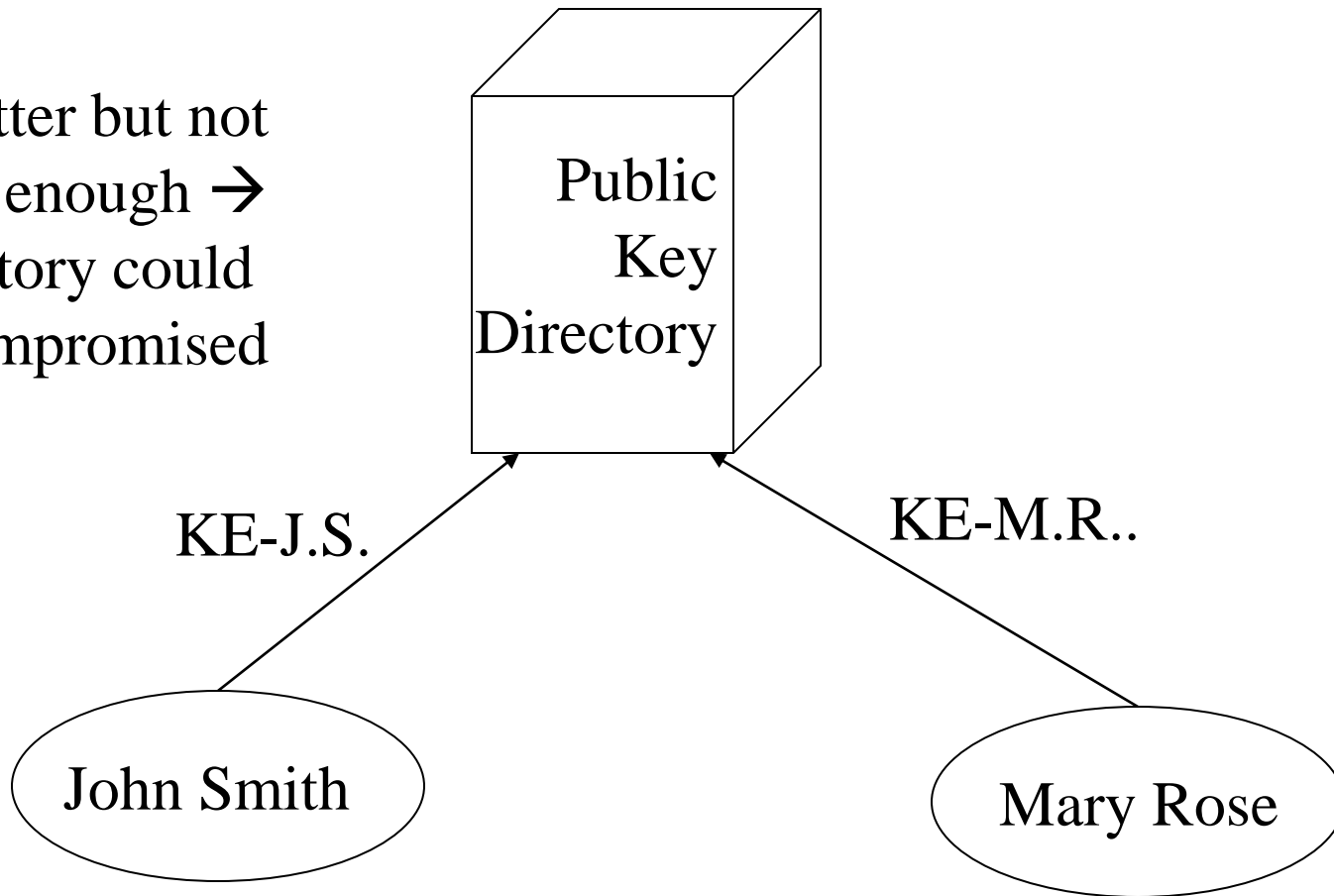
- Without server
 - Broadcasting
 - Publicly available directory
- With server
 - Public key distribution center
 - Certificates

Public announcement



Publicly available directory

Better but not
Good enough →
Directory could
Be compromised



Digital Signature

Need the same effect as a real signature •

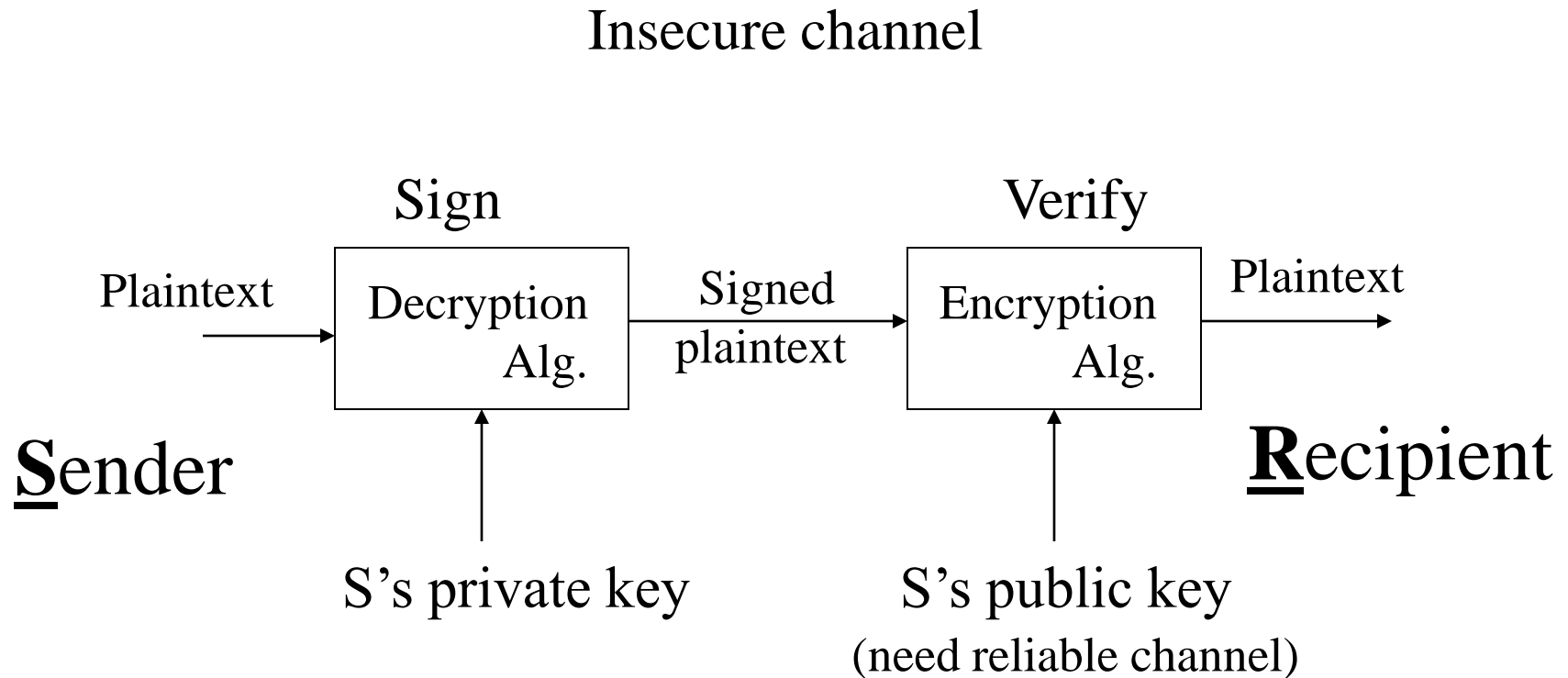
Un-forgeable –

Authentic –

Non-alterable –

Not reusable –

Digital Signatures in RSA



Attackers' Capabilities

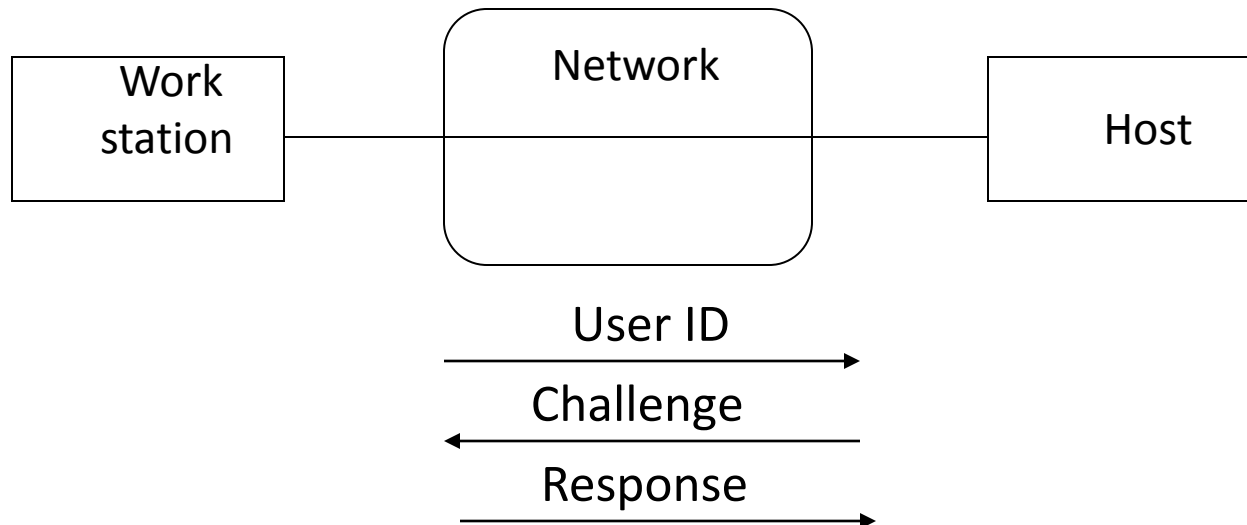
- Read traffic •
- Modify traffic •
- Delete traffic •
- Perform cryptographic operations •
- Control over network principals •

Password Management Policy

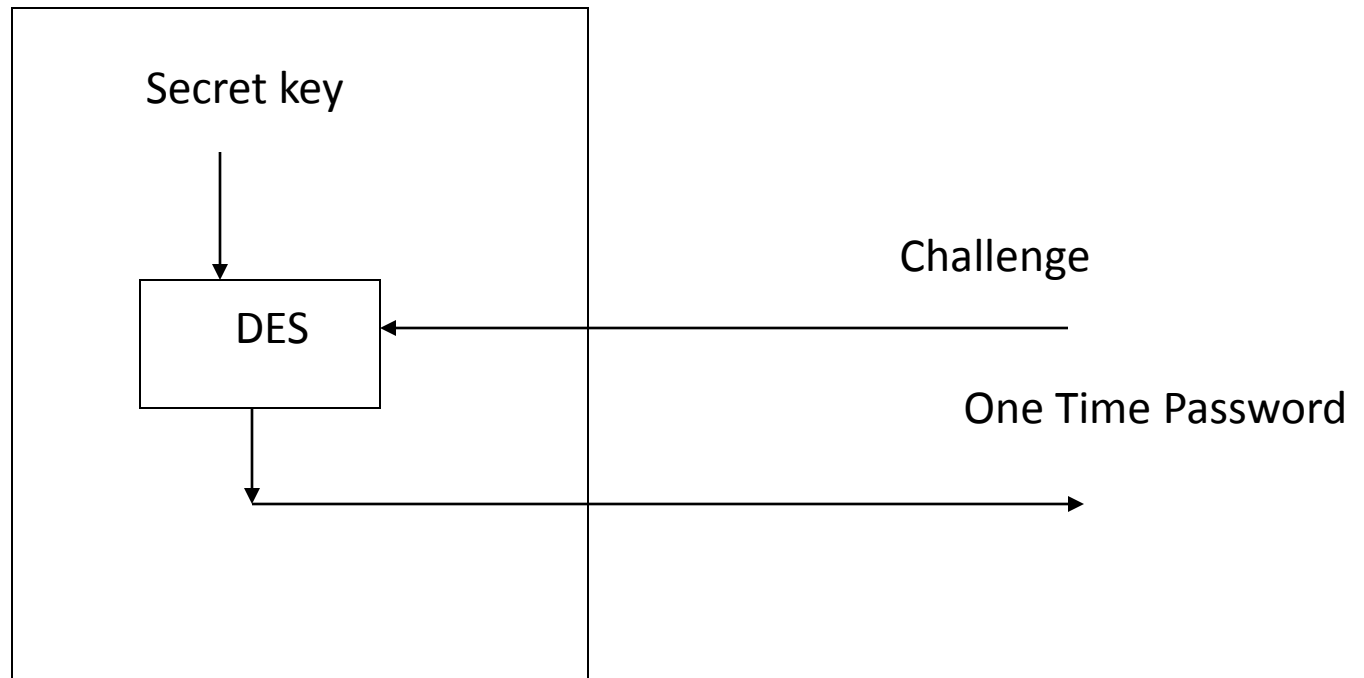
- Educate users to make better choices
 - Define rules for good password selection and ask users to follow them
 - Ask or force users to change their password periodically
 - Actively attempt to break user's passwords and force users to change broken ones
 - **One-time Password:** Screen password choices
- Use the password exactly once!

Challenge Response

- Non-repeating challenges from the host is used
- The device requires a keypad



Challenge Response



Devices with Personal Identification Number (PIN)

- Devices are subject to theft, some devices require PIN (something the user knows)
- PIN is used by the device to authenticate the user
- Problems with challenge/response schemes
 - Key database is extremely sensitive –
 - This can be avoided if public key algorithms are used

Biometrics

- Fingerprint •
- Retina scan •
- Voice pattern •
- Signature •
- Typing style •

Access Control

Protection objects: system resources for which protection is desirable •

Memory, file, directory, hardware resource, —
software resources, etc.

Subjects: active entities requesting accesses to resources •

User, owner, program, etc. —

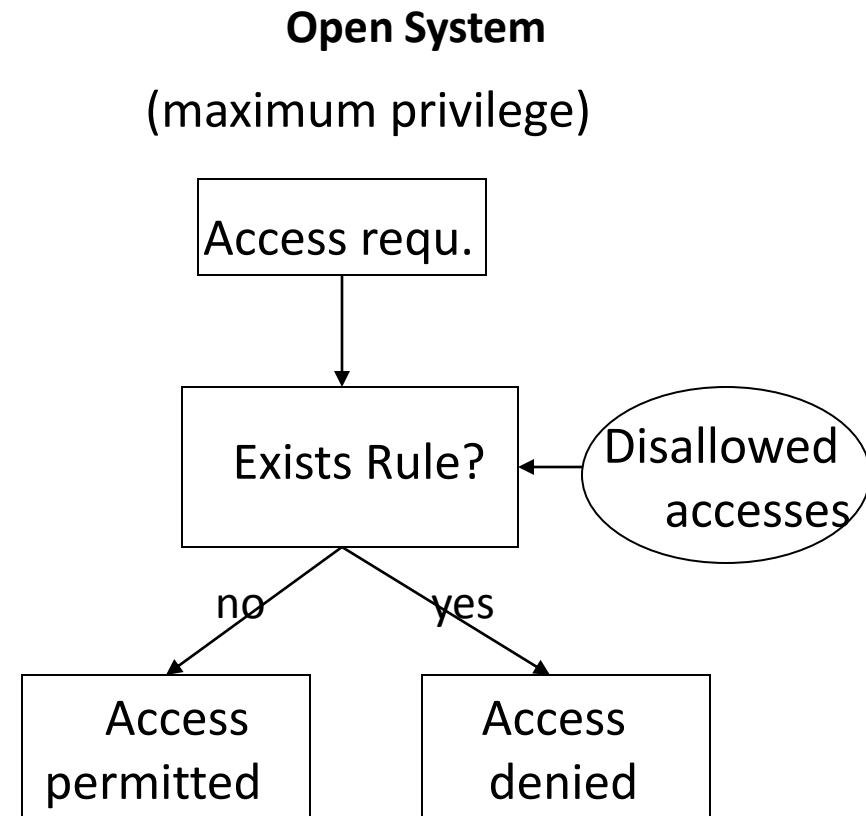
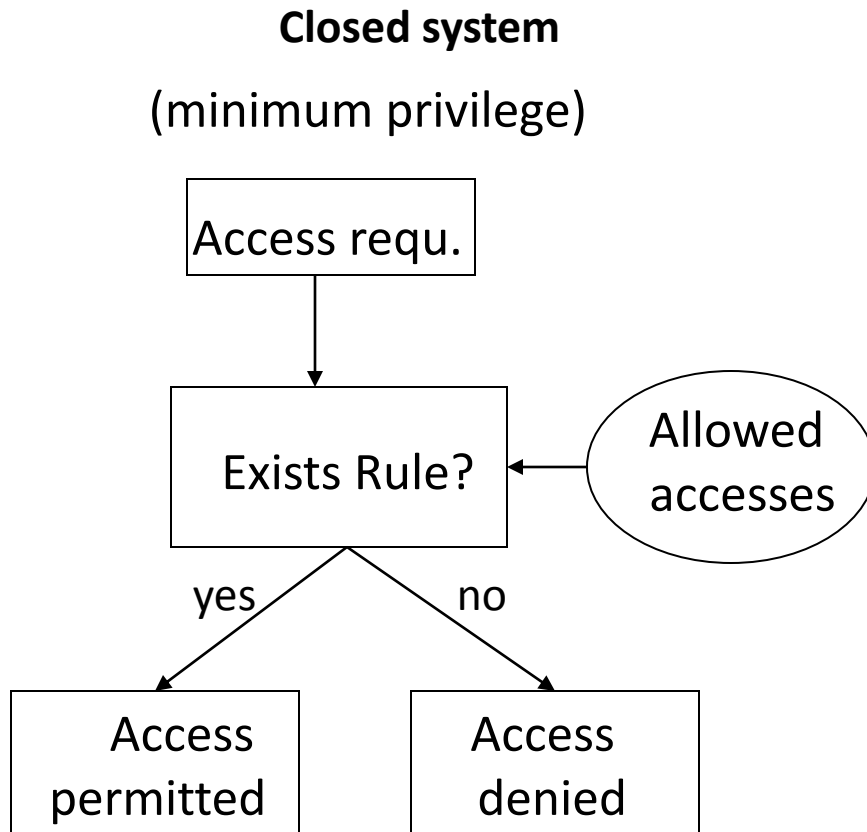
Access mode: type of access •

Read, write, execute —

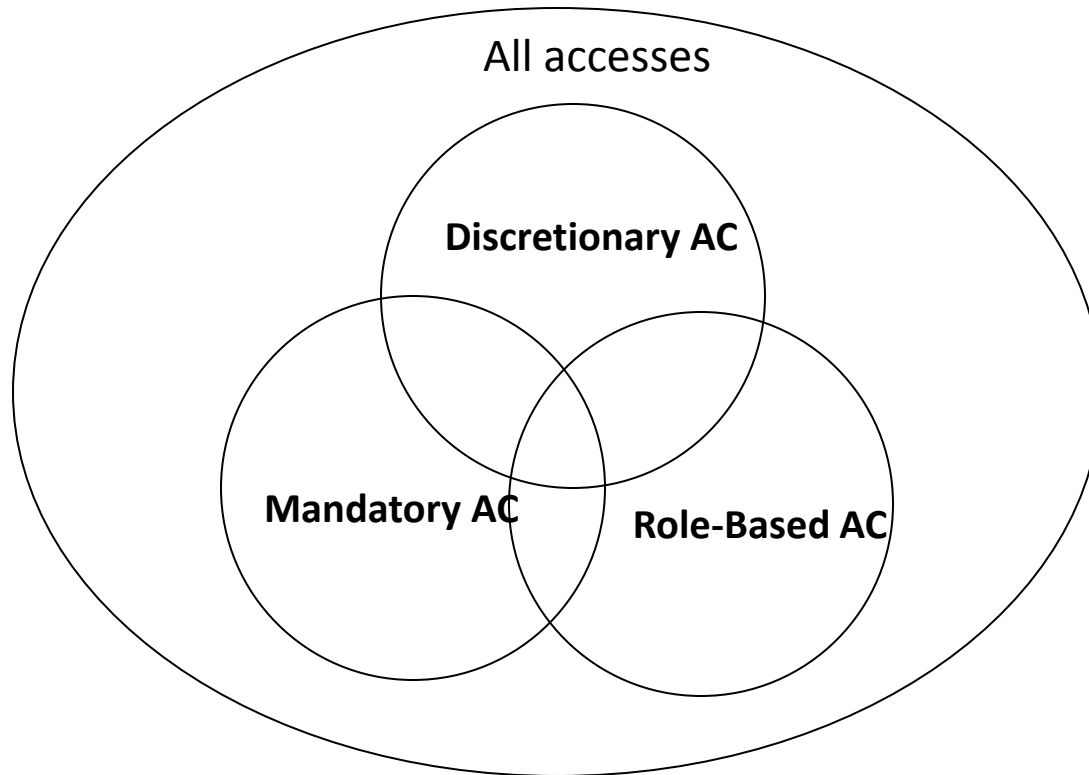
Access Control Requirement

- Cannot be bypassed
- Enforce least-privilege and need-to-know restrictions
- Enforce organizational policy

Closed v.s. Open Systems



Access Control Models



Access Matrix Model

		OBJECTS AND SUBJECTS →	
SUBJECTS ↓		File 1	File 2
	Joe	Read Write Own	Read
	Sam		Read Write Own

Implementation

Access Control List (column) (ACL)	File 2	File 1
	Joe:Read	Joe:Read
	Sam:Read	Joe:Write
	Sam:Write	Joe:Own
Capability List (row)	Sam:Own	

Joe: File 1/Read, File 1/Write, File 1/Own, File 2/Read

Sam: File 2/Read, File 2/Write, File 2/Own

	<u>Object</u>	<u>Access</u>	<u>Subject</u>
Access Control Triples	File 1	Read	Joe
	File 1	Write	Joe
	File 1	Own	Joe
	File 2	Read	Joe
	File 2	Read	Sam
	File 2	Write	Sam
	File 2	Own	Sam

ACL v.s. Capabilities

ACL: •

Per object based –

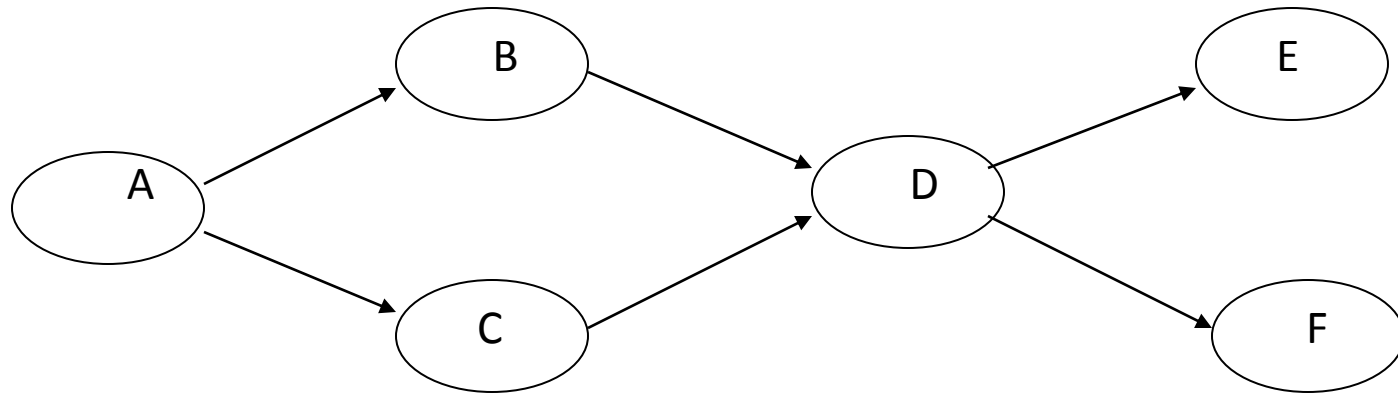
Good for file systems –

Capabilities: •

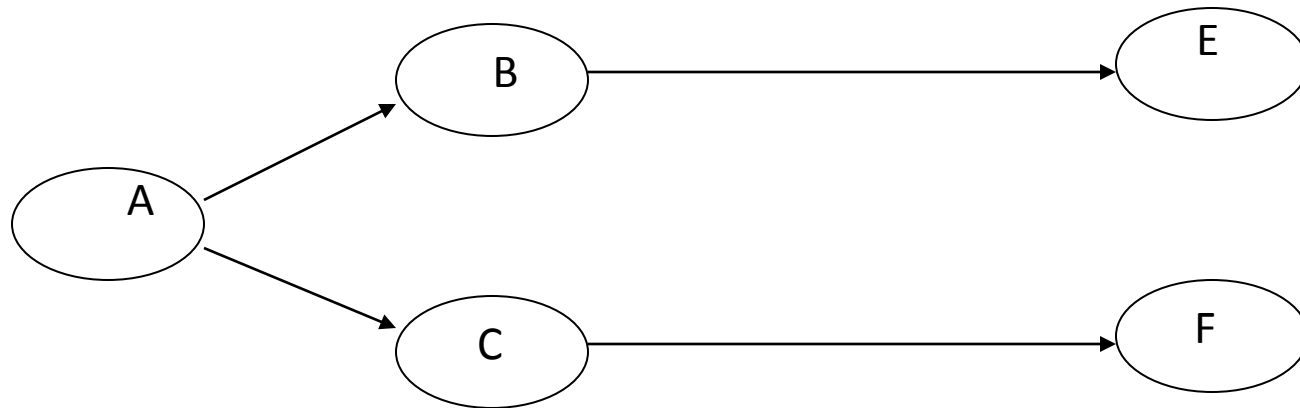
Per subject based –

Good for environment with dynamic, short-lived –
subjects

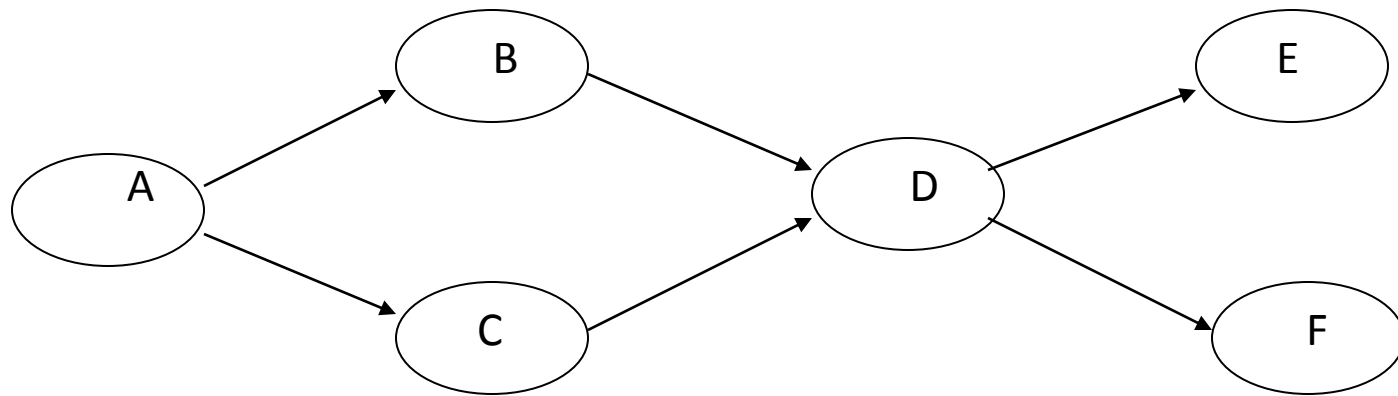
Non-cascading Revoke



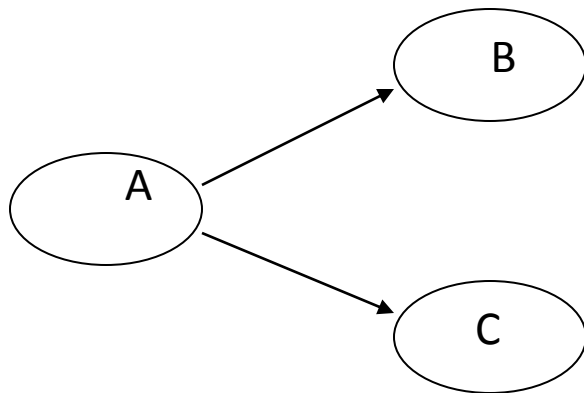
A revokes D's privileges



Cascading Revoke



A revokes D's privileges



DAC Overview

Advantages: •

Intuitive –

Easy to implement –

Disadvantages: •

Inherent vulnerability (look TH example) –

Maintenance of ACL or Capability lists –

Maintenance of Grant/Revoke –

Limited power of negative authorization –

Access Control

MAC

Mandatory Access Control

Objects: security classification #

e.g., grades=(confidential, {student-info})

Subjects: security clearances #

e.g., Joe=(confidential, {student-info})

Access rules: defined by comparing the security #
classification of the requested objects with the
security clearance of the subject

e.g., subject can read object only if label(subject)
dominates label(object)

Mandatory Access Control

If *access control rules* are satisfied, access is **permitted**

e.g., Joe wants to read grades.

label(Joe)=(confidential,{ student-info })

label(grades)=(confidential,{ student-info })

Joe is permitted to read grades

Granularity of access rights! **permitted**

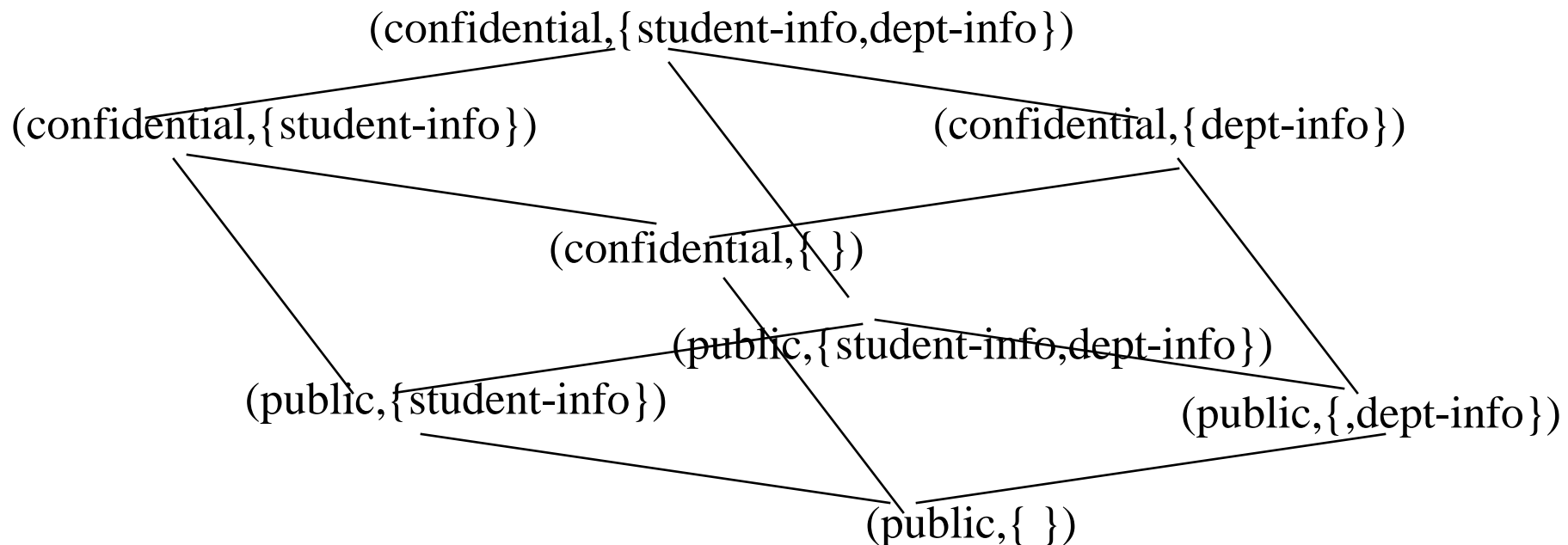
Mandatory Access Control

Security Classes (labels): (A,C)

A – total order authority level

C – set of categories

A = confidential > public , C = {student-info, dept-info} e.g.,



Mandatory Access Control

Dominance (\geq): label $l=(A,C)$ dominates $\#$
 $l'=(A',C')$ iff $A \geq A'$ and $C \supseteq C'$

e.g., (confidential, { student-info }) \geq (public, { student-info })

BUT

(confidential, { student-info }) $\not\geq$ (public, { student-info,
department-info })

BLP Axioms 1.

Simple-security property: a subject s is allowed to read an object o *only if* the security label of s dominates the security label of o

No read up ☐

Applies to *all subjects* ☐

BLP Axioms 2.

**-property*: a subject s is allowed to write an object o *only if* the security label of o dominates the security label of s

No write down ■

Applies to *un-trusted subjects* only ■