

Part 1: Initial state

(a) Sets of Subjects, Objects, and Actions

- **Subjects S** = { Alice (Knight), Bob (Wizard), Carol (Rogue), Shun }
- **Objects O** = { Manifesto, Spellbook, Sword, TreasureChest, CastleDoor, InventoryList }
- **Actions A**
 - Standard: read, write
 - Domain specific rights modelled as distinct permissions:
 - attack(Shun)
 - cast_on(CastleDoor)
 - take(Sword)
 - steal(TreasureChest)
 - burn(TreasureChest), burn(CastleDoor)

(b) Access Control Matrix (ACM)

Subject Object	Manifesto	Spellbook	Sword	TreasureChest	CastleDoor	InventoryList
Alice (Knight)	read, attack(Shun)	—	—	—	—	read
Bob (Wizard)	—	read, write	—	—	cast_on	—
Carol (Rogue)	read	—	take	steal	—	—
Shun	—	—	—	burn	burn	—

(c) Access Control Lists (ACLs)

- **Manifesto:** { Alice: read, Carol: read, Alice: attack(Shun) }
- **Spellbook:** { Bob: read, write }
- **Sword:** { Carol: take }
- **TreasureChest:** { Carol: steal, Shun: burn }
- **CastleDoor:** { Bob: cast_on, Shun: burn }
- **InventoryList:** { Alice: read }

(d) Capability Lists (CLs)

- **Alice:** { Manifesto: read, attack(Shun); InventoryList: read }
- **Bob:** { Spellbook: read, write; CastleDoor: cast_on }
- **Carol:** { Manifesto: read; Sword: take; TreasureChest: steal }
- **Shun:** { TreasureChest: burn; CastleDoor: burn }

Part 2: Dynamic System and the Principle of Least Privilege

New decree

1. Bob is granted enchant on Sword
2. Alice must be able to use the enchanted Sword on Shun.

(a) Updated Access Control Matrix (ACM)

Subject \ Object	Manifesto	Spellbook	Sword	TreasureChest	CastleDoor	InventoryList	Shun
Alice (Knight)	read, attack(Shun)	—	—	—	—	read	use

Bob (Wizard)	—	read, write	enchant	—	cast_on	—	—
Carol (Rogue)	read	—	take	steal	—	—	—
Shun	—	—	—	burn	burn	—	—

(b) Updated Capability Lists (CLs)

- **Alice:** { Manifesto: read, attack(Shun); InventoryList: read; Shun: use }
- **Bob:** { Spellbook: read, write; CastleDoor: cast_on; Sword: enchant }
- **Carol:** { Manifesto: read; Sword: take; TreasureChest: steal }
- **Shun:** { TreasureChest: burn; CastleDoor: burn }

(c) Analysis and discussion

- **For administration:** Access Control List are easier. To give Bob the new privilege you edit the Sword object and add enchant for Bob. One object touched, change is local.
- **For auditing a user:** Capability lists are easier. To see everything Alice can do, read Alice's CL once and you will see her new use right on Shun without scanning all objects.

Summary: Use ACLs when you manage access on an object. Use capability views when you review or explain a subject's total privileges. This keeps least privilege tight and makes audits straightforward.