Lecture 9

Tuesday, 4 October 2022 10:44

access control

- DAC
- MAC
- RBAC
- ABAC

trust management

- RT

DRM

access control is identity-based authorization

rights pre-defined and granted to subjects

repeated access until explicit revocation of rights

access decision at request time

enforcement at server side

trust management → (RT) similar, but based on credentials

DRM
- → limit copying, printing, sharing
- → protect intellectual property
- → provide evidence of misuse (watermarks)
- → prevent unauthorized use of proprietary documents

access control, due to its static nature, is not suitable for situation with changing access (e.g. pay-per-use)
→ we want e.g. consumable rights

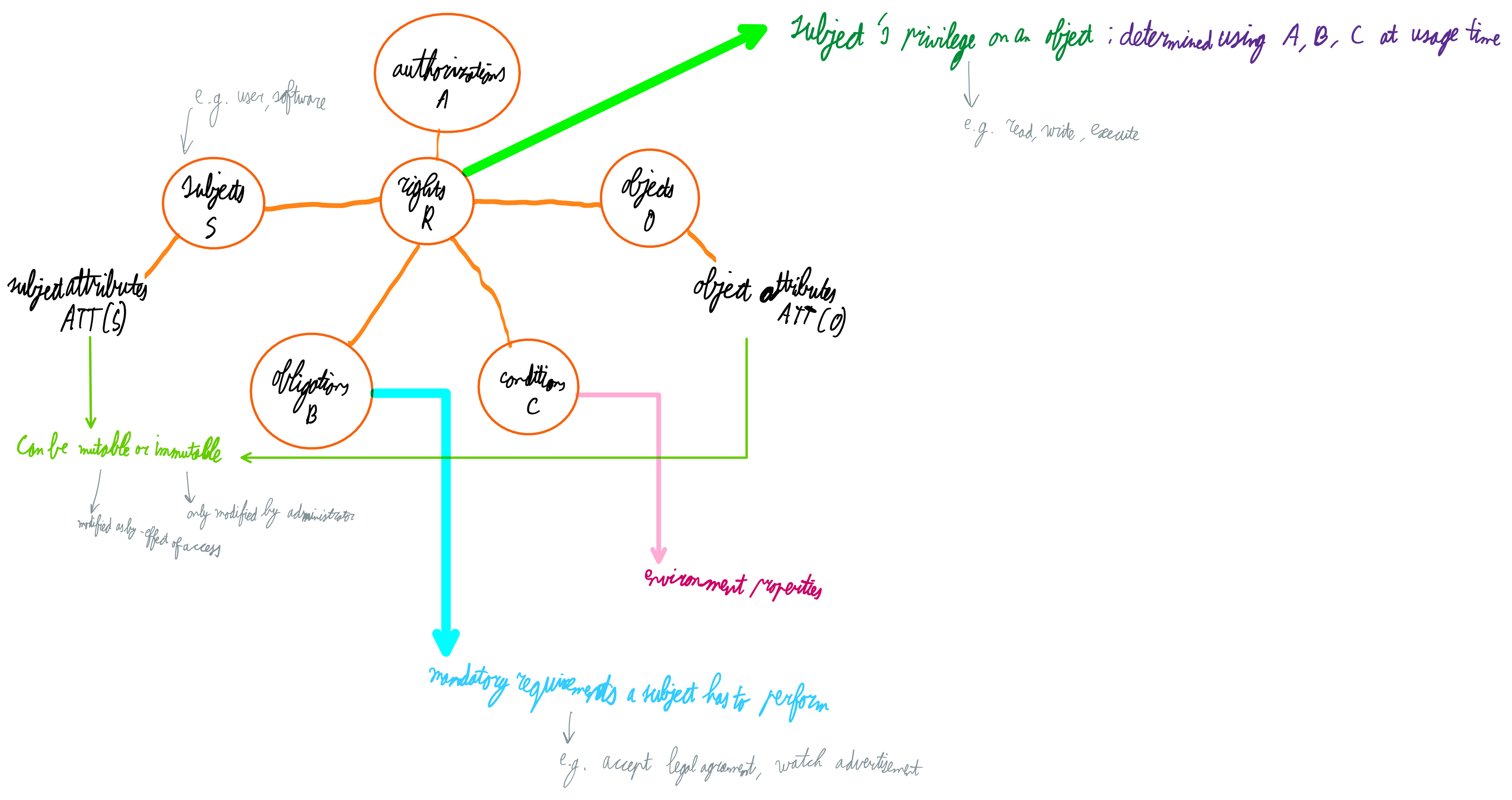
in addition, access control does not provide control over disclosed objects

---

In usage control, attributes can change as a result of access

decision factors
- authorizations
- obligations
- conditions

decision properties
- continuity of decisions → pre-decision, ongoing-decision
- mutability of attributes → pre-update, ongoing-update, post-update



Let $\geq$ be the dominance relation in the BLP model

Let L be a lattice of security labels with that dominance relation

clearance : $S \to L$

classification : $O \to L$

BLP model

$ATT(s) = \{clearance\}$

$ATT(o) = \{classification\}$

$allowed(s, o, read) \Rightarrow clearance(s) \geq classification(o)$

$allowed(s, o, write) \Rightarrow classification(o) \geq clearance(s)$

$allowed(s, o, r) \Rightarrow preA(ATT(s), ATT(o), r)$

is equivalent to

$\neg preA(ATT(s), ATT(o), r) \Rightarrow denied(s, o, r)$

---

ongoing authorization

$allowed(s, o, r) \Rightarrow true$

$stopped(s, o, r) \Leftarrow \neg onA(ATT(s), ATT(o), r)$

Exercise: limited number of simultaneous usages, revocation using total time

$allowed(s, o, r) \Rightarrow true$

$stopped(s, o, r) \Leftarrow usageNum(o) > 10 \wedge (id(s), tt) \in totalT(o)$ (10: limit) with tt time of the active user with greatest total time

$preupdate(usageNum(o)) : usageNum(o) = usageNum(o) + 1$

$postupdate(usageNum(o)) : usageNum(o) = usageNum(o) - 1$

some more rules...

---

pre-obligation

obligation subjects OBS

obligation objects OBO

obligations OB

$getPreOBL(s, o, r) =$