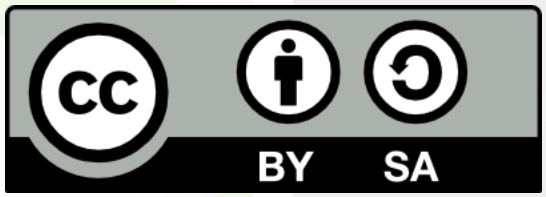
Computer Security

Authorization and Access Control

Prof. Jean-Noël Colin
jean-noel.colin@unamur.be
Office #306

University of Namur
Computer Science Faculty

www.unamur.be



Agenda

- Introduction
- DAC model
- MAC model
- RBAC model
- ABAC model
- A couple of others
- Policy Management and Enforcement: XACML

Introduction

- [permission] authorization to perform an operation on a resource
- [access rule] assignment of a permission to a subject, assorted with conditions
- [access policy] set of access rules
- [access control] enforcement of access policy; this requires
  - proper subject identification
  - secure access control system (incl. rules)
  - ownership/responsibility

General principles

- A Subject wants to perform an action on a resource
- All accesses must be mediated by a *guard* or *reference monitor*
- Based on its policy, the guard makes a decision to grant or deny the requested access
- Ensure that no access can by-pass the guard

General principles

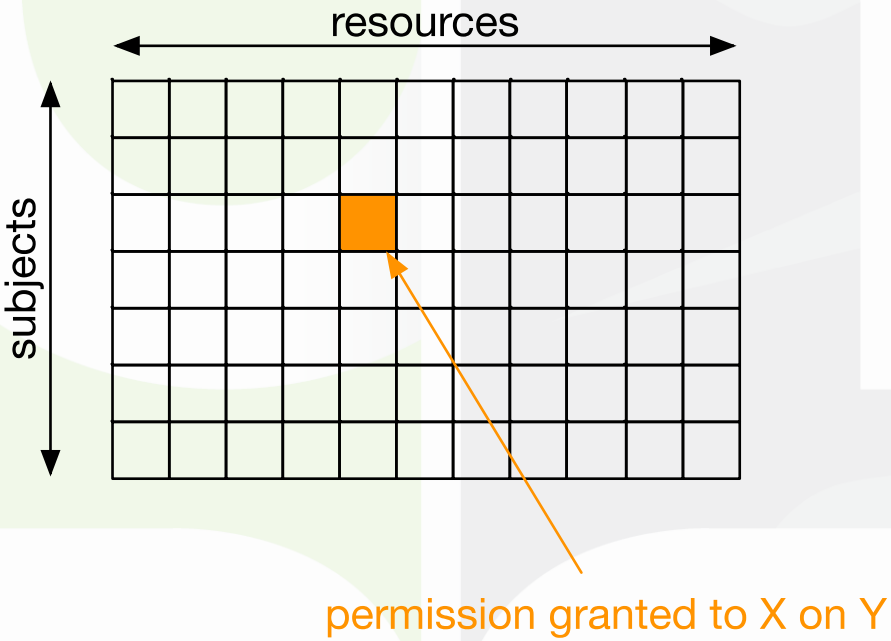
- Access Control Model
  - Meta-model of a policy
    - set of concepts, relationships, constraints that allow to express an access control policy in a formal way
  - a formalized policy can be automatically processed:
    - validation: consistency, rule conflict or redundancy detection
    - automatic enforcement
  - model = proven and validated mechanism
  - (many) different models exist
    - degree of control/involvement of the user, expressiveness, security objectives…

General principles

- Least privilege
  - grant the minimal set of permissions to allow proper execution of the assigned tasks
    - example: do not use ‘root’ account, use ‘sudo’
- Separation of duties
  - require separate roles/responsibilities to carry out a sensitive action
    - example: submit and approve a payment should not be in the hands of the same person

Discretionary Access Control (DAC)

- Relies on resource ownership
- Permissions defined by resource owner
- Permission matrix



Discretionary Access Control (DAC)

- Group
  - subject and resources
  - unique or multiple group assignment
- Capability or ACL (Access Control List)
  - capability: token describing the permissions granted to a user
    - ex: oAuth token
  - ACL: list of permissions linked to an object
    - frequently used in OS, LDAP server, DB. . .
- Other features
  - transfer of ownership
  - delegation of rights (access on behalf of)
- Limitations
  - complex management

Mandatory Access Control (MAC)

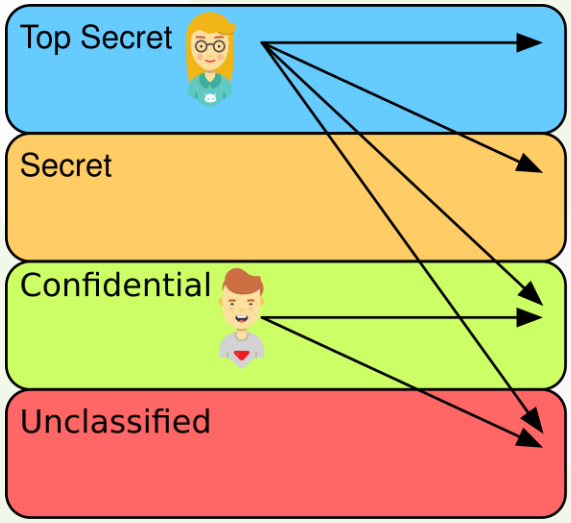
- Permissions defined a priori
  - at system design/configuration
  - user has no control over the access control policy
- Controls
  - information flow
  - information confidentiality and integrity
- Based on Multilevel Security – MLS
  - subject and resource are labeled with a level of security
    - clearance, classification
  - “Need-to-know” principle
    - sub-classes within a level

Mandatory Access Control (MAC)

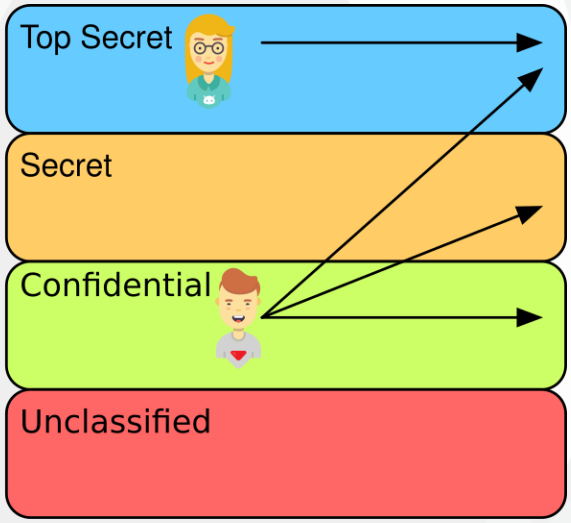
- Bell - La Padula Model (1973)
  - given
    - S, a set of subjects
    - O, a set of objects
    - A, a set of access operations
    - L, a set of security levels, with partial order
    - $f_S : S \rightarrow L$, defines the max security level of a subject
    - $f_C : S \rightarrow L$, defines the current security level of a subject, such that $f_C(s) \leq f_S(s), \forall s$
    - $f_O : O \rightarrow L$, defines the security level of an object
  - *ss-property*: $\forall s \in S, o \in O, a \in A : (f_O(o) \leq f_C(s)) \wedge$ (a of type *read*) $\Leftrightarrow$ permission(s, o, a)
  - **-property*: $\forall s \in S, o \in O, a \in A : (f_O(o) \geq f_C(s)) \wedge$ (a of type write) $\Leftrightarrow$ permission(s, o, a)

Mandatory Access Control (MAC)

- Bell - La Padula Model (1973)



No Read Up



No Write Down

Mandatory Access Control (MAC)

- Bell - La Padula Model (1973)
  - Need for exception mechanism
    - ex: send information downwards
  - Solutions
    - temporary lower subject's security level
    - concept of 'trusted' subject
  - BLP guarantees confidentiality
  - BLP does not guarantee integrity
  - often used in military context or environments where strong formal security is required

Mandatory Access Control (MAC)

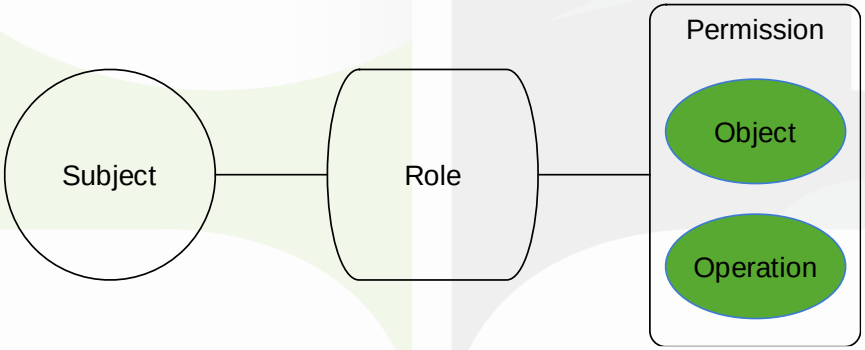
- Biba Model (1977)
  - Dual to Bell – La Padula
  - aims at protecting information integrity
  - Simple integrity principle: no write up
  - Integrity * principle: No read down
- MAC models are used!
  - SELinux
  - Solaris Trusted Extensions
  - Vista

Role-Based Access Control (RBAC)

- Proposed by Ferraiolo and Kuhn in 1992
- Approved as ANSI standard in 2004
- ‘Remedy’ to MAC and DAC limitations
  - rigidity of MAC
  - management of DAC
- Higher level of abstraction
  - closer to business concepts

Role-Based Access Control (RBAC)

- Model concepts
  - [Object] resource subject to access control
  - [Operation] action over an object
    - in the standard, linked to the notion of program
  - [Permission] authorization to perform operation on object
  - [Role] function defined in the context of an organization
    - defines authority and associated responsibilities User subject who whishes to perform an operation on object
    - considered as human in the standard, extended to machine, process. . .



Role-Based Access Control (RBAC)

- Permissions derived from the roles assigned to subjects, which allows to enforce policies like
  - least privilege
  - static separation of roles
    - ex: conflict of interest
  - cardinality of roles
    - ex: max 1 person assigned with director role
- Session
  - at session opening, activation of a subset of authorized roles
  - can be subject to further constraints
    - dynamic separation of roles
- Hierarchical roles
  - roles can be organized in hierarchies, allowing for permission inheritance

Role-Based Access Control (RBAC)

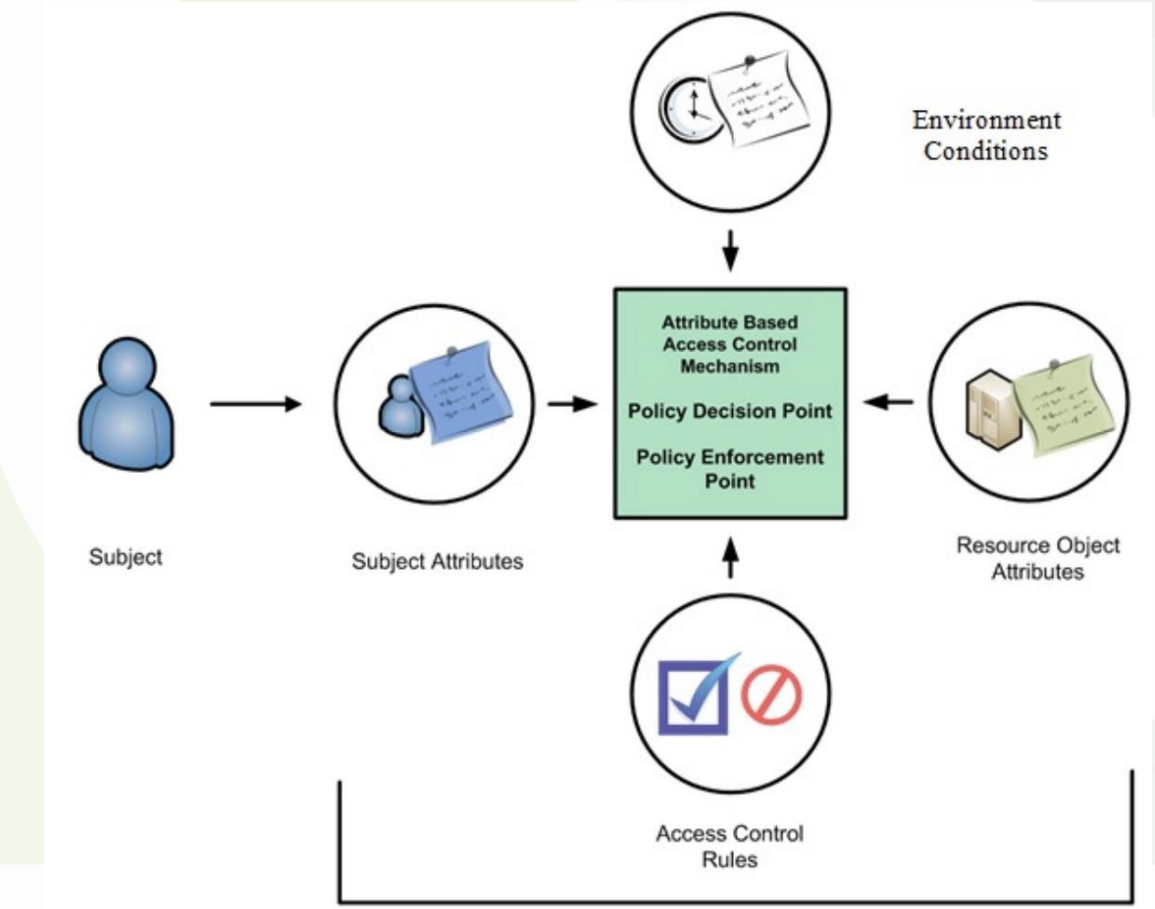
- Model management
  - 3 steps
    - Definition of permissions
    - Permission-to-Role assignment
    - Role-to-User assignment
  - Each role must have an owner that manages
    - role definition
    - role assignment rules
    - incidents related to the role

Attributed-Based Access Control (ABAC)

"an access control method where subject requests to perform operations on objects are granted or denied based on assigned attributes of the subject, assigned attributes of the object, environmental conditions, and a set of policies that are specified in terms of those attributes and conditions"

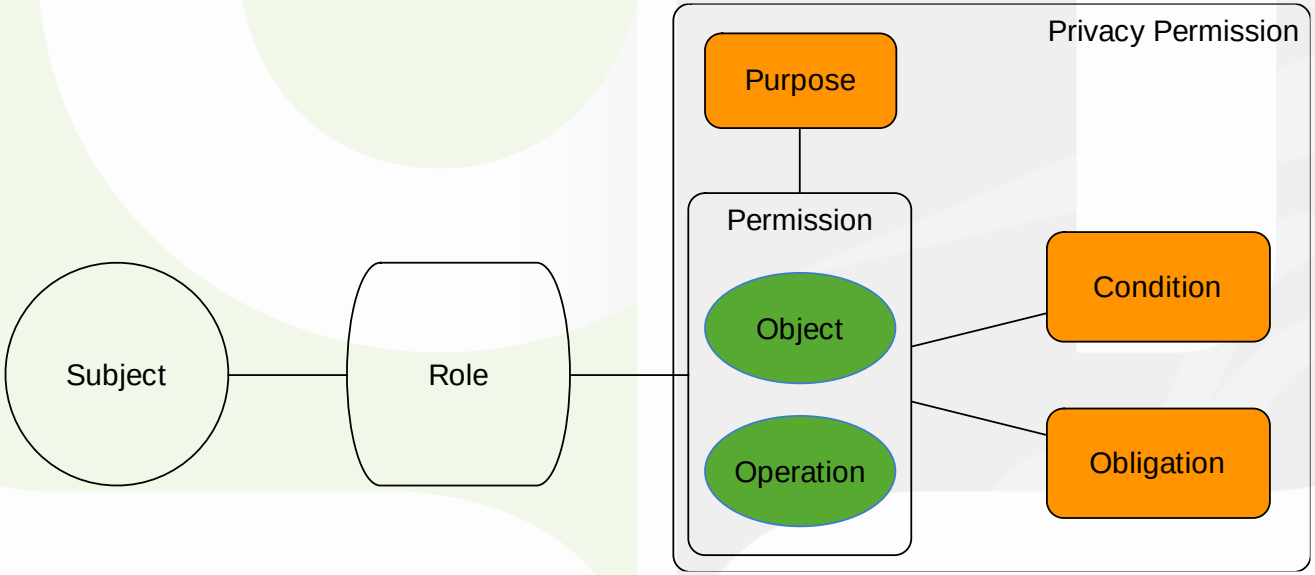
Source: NIST 800-162: Guide to Attribute Based Access Control (ABAC) Definition and Considerations

Attributed-Based Access Control (ABAC)



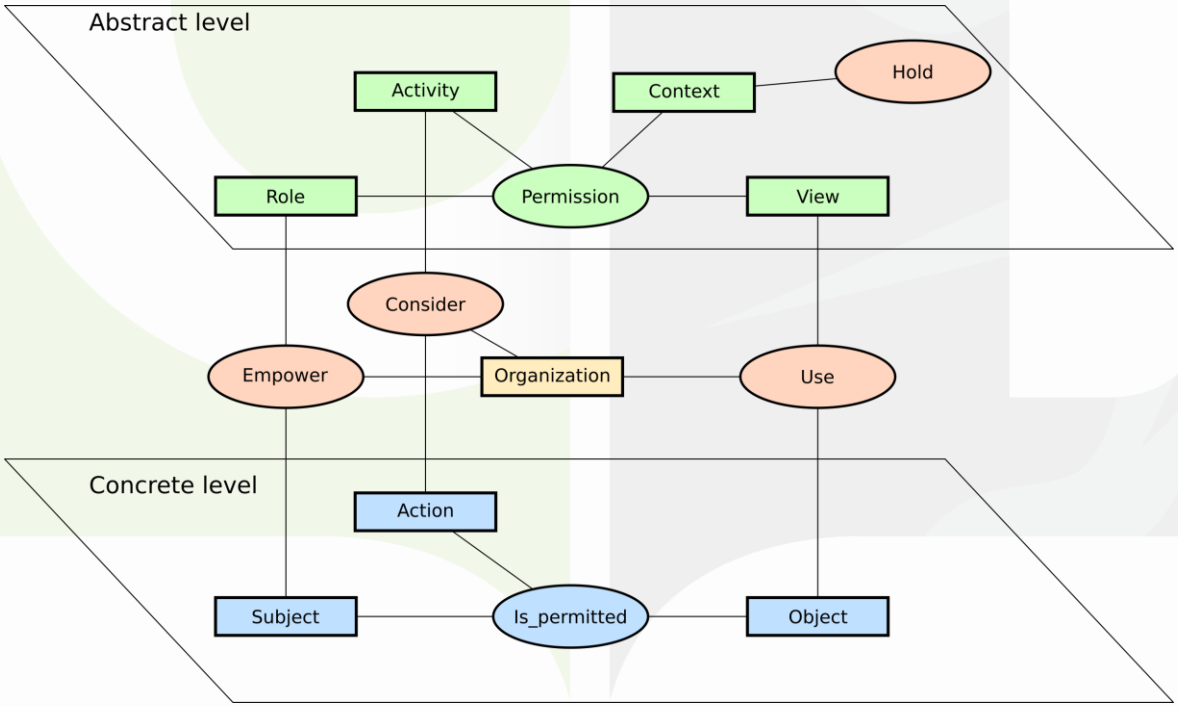
Privacy Aware RBAC (P-RBAC)

- Extends RBAC to include privacy-related mechanisms
  - Purpose, condition, obligation



Organization-based Access Control (OrBAC)

- Focus on the concept of organization
- Two levels: abstract and concrete
- Permission can be positive or negative



Policy Management and Enforcement

- The situation
  - Access control policies are complex in their content and structure
  - Policies are enforced in multiple places: firewall, applications…
  - Regulatory pressure: trace accesses and decisions…
- The consequences
  - High management and maintenance cost
  - High risk of inconsistent policies and decisions
  - Hard to trace

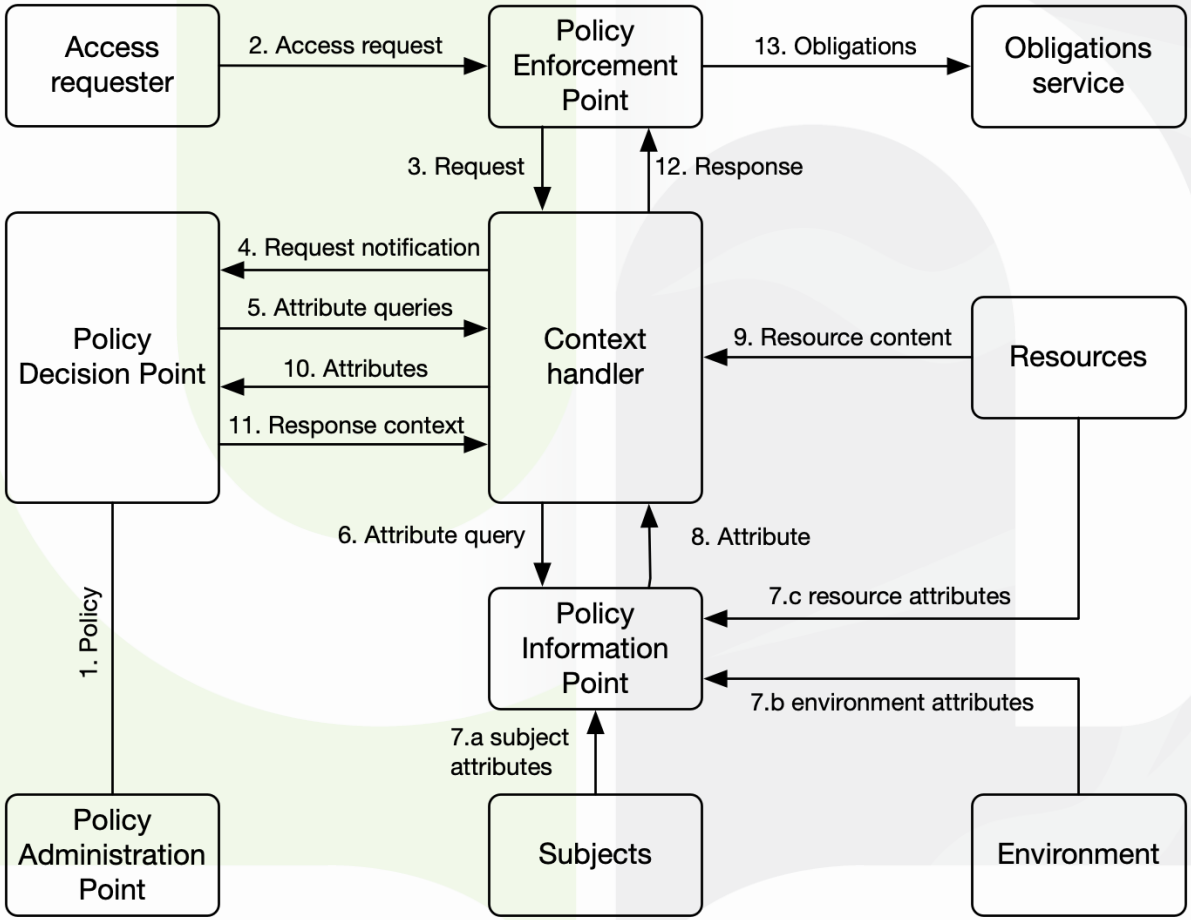
Policy Management and Enforcement

- Towards a distributed approach
  - Centralized definition and management of policies
  - Adoption of a common model and language for policy expression
  - Managing policies: write, validate, update, approve, combine, enforce…
  - Separate decision flow from application flow
    - Policies defined outside of application
    - Policy decisions are taken outside of application
    - Application only executes the decision

Policy Management and Enforcement

- XACML Model
  - OASIS Standard: v2.0 2005, **v3.0 2013**
  - Generic, extensible
  - Distributed model
  - Access control model agnostic
  - Defines
    - a policy language (policy set, policy, rule, combination rules)
    - A communication protocol
  - Several components
    - [PAP] Policy Administration Point
    - [PEP] Policy Enforcement Point
    - [PDP] Policy Decision Point
    - [PIP] Policy Information Point

Policy Management and Enforcement



Policy Management and Enforcement

1. PAPs write **policies** and **policy sets** and make them available to the PDP. These policies or policy sets represent the complete policy for a specified **target**.
2. The access requester sends a request for access to the PEP.
3. The PEP sends the request for access to the context handler in its native request format, optionally including attributes of the subjects, resource, action, environment and other categories.
4. The context handler constructs an XACML request context, optionally adds attributes, and sends it to the PDP.
5. The PDP requests any additional subject, resource, action, environment and other categories (not shown) attributes from the context handler.
6. The context handler requests the attributes from a PIP.
7. The PIP obtains the requested attributes (7.a, 7.b, 7.c)
8. The PIP returns the requested attributes to the context handler.
9. Optionally, the context handler includes the resource in the context.
10. The context handler sends the requested attributes and (optionally) the resource to the PDP. The PDP evaluates the policy.
11. The PDP returns the response context (including the authorization decision) to the context handler.
12. The context handler translates the response context to the native response format of the PEP. The context handler returns the response to the PEP.
13. The PEP fulfills the obligations.
14. (Not shown) If access is permitted, then the PEP permits access to the resource; otherwise, it denies access.

Conclusion

- Access Control policies are central to define **who** can do **what** under which **conditions**
- Part of the requirements engineering
- Various models are available to model, express, validate and enforce the policy: choose the one that best fits your needs
- When implementing a decentralized approach
  - Think ‘interoperability’
  - Be patient