**Patterns for Access Control in Distributed Systems**

Summary:

* The paper discusses security patterns for authentication and access control in distributed systems.
* Distributed systems face more security threats because of multiple users, partners, and environments.
* The three main patterns explained are:
  + Policy-based Access Control (PBAC)
    - All access rules are already stored in a central repository.
    - Components: Policy Enforcement Point (PEP), Policy Decision Point (PDP), Policy Information Point (PIP).
    - Ensures consistent policy enforcement across the system.
    - Advantage: flexible, supports multiple models (Access Matrix, RBAC).
    - Limitations: performance bottleneck, complexity.
  + Access Control List (ACL) Pattern
    - Each object**/**resource has a list of which users can access it and what operations they can do.
    - For example: file permissions in Windows/Unix.
    - Advantage: efficient to check access for a specific object.
    - Limitation: hard to manage when there are many users, revocation requires scanning lists.
  + Capability Pattern
    - Users are given tokens**/**credentials (capabilities) that prove their rights.
    - For example: a signed token showing permission for a user.
    - Advantage: fast verification, minimal trusted system part.
    - Limitation: hard to revoke, tokens may be stolen or misused
* These patterns can implement models like RBAC (Role-based access control) and access matrix.
* They can also be supported by standards like XACML (it is a policy language) and SAML (authorization assertions).