1 Introduction

Removing the credentials of departing users is a relevant problem for large organizations. Removing all access rights from users who are leaving a company can be a challenging task due to the difficulty of understanding how the access rights are configured in an organization. The objective of this master thesis is to develop a novel approach based on authenticated data structures [1, 3].

Large organizations use access control to ensure that its employees are granted access to resources on a need-to-know basis only. The typical architecture of an access control system is the following. The policy administrator point (PAP) defines and loads the access control policy into the policy decision point (PDP), which becomes then in charge of deciding which employee can access which resources. When an employee wants to access a resource, she must present her credentials to a policy enforcement point (PEP) that will query the PDP whether the employee can access the resource. Upon confirmation from the PDP, the PEP grants access to the employee.

In large organizations, the PAP, PEP, and PDP may be supervised by different people, which do not necessarily trust each other. This poses a challenge for the PEP: if the PDP claims someone should be granted access, how can the PEP be confident that this is conformant with the policy defined by the PAP? This is important in scenarios where the PAP may change the policy on a regular basis and neither the PEP nor the PAP have sufficient resources to store and maintain copies of the access control policy.

In this project we want to address these challenges with a novel approach: authenticated data structures [1, 3]. An authenticated data structure is a structure managed by a prover who performs operations on it upon a verifier’s request. After performing an operation, the prover must generate and send to the verifier a proof (usually logarithmic on the size of the structure), which can be used by the verifier to efficiently check that the prover performed the operation correctly.

The objective in this project is to develop an authenticated RBAC model, where the PDP acts as prover and the PAP and the PEP act as verifiers. The PDP should be able to perform the usual RBAC operations on the model. For each operation, the PDP should in addition produce a succinct proof (e.g. logarithmic in the size of the RBAC model). The PAP or the PEP can use then this proof to verify that the PDP performed the operation correctly.

2 Tasks

The development of the authenticated RBAC model requires the following tasks to be completed:
**Specification:** Define two components. First, an authenticated data structure for a typical RBAC model [2] that will be stored in the PDP. Second, protocols that allow the PAP and the PEP to do the following operations on the RBAC model:

1. Query whether a user is authorized for a permission.
2. Insert a new user or a new permission to the model.
3. Assign a user to a role or a permission to a role in the model.
4. Revoke the assignment of a role to a user or the assignment of a permission to a role.
5. Remove a user or a permission from the model.

We consider the last two operations the most challenging.

**Correctness proofs:** State and prove properties that guarantee correctness and security for the operations above. By correctness we mean that if the PDP, the PAP, and the PEP behave correctly, then the PEP grants and denies access as the PAP intended. By security we mean that a malicious PDP cannot fool the PEP into granting or denying access in a way different as the PAP intended.

**Implementation:** Develop a prototype consisting of a PAP, a PDP, and a PEP with a sample RBAC model that allows end-users to interact and understand our approach with authenticated data structures.

**Experimental evaluation:** Measure the time required to perform the operations described in the specification task for an authenticated RBAC model and compare it with the time required to perform the same operations for a non-authenticated RBAC model.

3 Deliverables

- At the end of the second week, a detailed time schedule of the master thesis must be given and discussed with the supervisor.
- At the end of the master thesis a presentation of 30 minutes must be given during an Infsec group seminar. It should give an overview as well as the most important details of the work.
- Software and configuration scripts must be delivered to the supervisors.
- A final report consisting of an introduction, a discussion on the related work on authenticated data structures, a presentation of the authenticated RBAC model together with proofs and evaluations. Three copies of this report must be delivered to the supervisor.

**References**

