published as: W. Zhou, V. H. Raja, Christoph Meinel, M. Ahmad: A Framework for Cross-Institutional Authentication and Authorisation;
Proceedings of the 6th Annual Conferences on eBusiness, eGovernment, eWork, eEurope 2005 (eChallenges 2005);
Ljubljana, Slovenia, October 2005, pp. 1259-1266.

# A Framework for Cross-Institutional Authentication and Authorisation

Wei ZHOU[1], Vinesh H. RAJA[2], Christoph MEINEL[3], Munir AHMAD[4]
[1,2]*School of Engineering, University of Warwick, Coventry CV4 7AL, UK*
[1]*Tel: +44 24 76575834, Fax: +44 24 76524307, Email: w.zhou.3@warwick.ac.uk*
[2]*Tel: +44 24 76523924, Fax: +44 24 76524307, Email: vinesh.raja@warwick.ac.uk*
[3]*Hasso-Plattner-Institute, University of Potsdam, Potsdam D-14482, Germany*
*Tel: +49 331 5509222, Fax: +49 331 5509325, Email: meinel@hpi.uni-potsdam.de*
[4]*B2B Manufacturing Centre, University of Teesside, Middlesbrough TS1 3BA, UK*
*Tel: +44 1642 342443, Email: m.m.ahmad@tees.ac.uk*

**Abstract:** To effectively participate in modern collaborations, member organizations must be able to share specific data and functionality with collaboration partners, while ensuring that their resources are safe from inappropriate access. This requires access control models, policies, and enforcement mechanisms for coalition resources. This paper specifically addresses how to exchange users' authentication and authorisation information between organizations, and how this mechanism can be used for virtual organization management. The basic principle is that a user is authenticated at his origin site, and the system creates a handle that is used to retrieve the user's attributes for the destination site. The destination site will use this handle to obtain the user's attributes that can be used for access control. A prototype called Cross Security Access Control Framework (CSACF) has been developed.

## 1. Introduction

With the advent of the information superhighway, businesses, governments and other organizations co-operate in innovative ways. To effectively participate in modern collaborations, member organizations must be able to share specific data and functionality with collaboration partners, while ensuring their resources are safe from inappropriate access. Such collaborations may dynamically change participants and trust relationships during the life cycle. This requires access control models, policies, and enforcement mechanisms for shared resources. However, current technologies do not comprehensively support such control for collaboration resource sharing. Though there are a variety of conceptual, technological and operational factors that contribute to this situation, we specifically address the following problem: how to exchange authentication and authorisation information in the inter-organizational collaborative computing environment, and the authorisation management in virtual organization.

In this paper, the proposed mechanism is that a user is authenticated locally at his origin site (identity provider), and the origin site creates a handle, from which the user's attributes (privileges) can be retrieved. This handle is stored in a XML document that will be signed and encrypted, and then sent to the destination site (resource provider). According to this handle, resource provider sends an attribute query message to the user's attribute authority for his authorisation information. The attribute authority then issues a X.509 attribute certificate that holds the user's privileges and sends it back to the attribute requester. The resource provider will provide some services to the user based on his privileges.

The rest of this paper is organized as follow. Section 2 introduces the basic concepts of privilege management infrastructure, XML Signature and XML Encryption. Section 3 introduces the Cross Security Access Control Framework (CSACF). Section 4 describes the Team and Task based RBAC (TT-RBAC) access control model. Section 5 describes how the CSACF and TT-RBAC support fine-grained and flexible virtual organization management. Section 6 generally describes our implementation. Section 7 compares our work to some related works. Finally, section 8 gives the conclusions and future works

## 2. Main related technologies introduction

### *2.1 – Privilege management infrastructure*

Privilege Management Infrastructure (PMI) is specified by the ITU-T and ISO/IEC [3]. The main function of PMI is providing a strong authorisation after the authentication has taken place. It has a number of similarities with Public Key Infrastructure (PKI) [4]. The basic data structure in PMI is X.509 Attribute Certificate (AC) [5]. Like Public Key Certificate (PKC) strongly binds a public key to its subject, AC strongly binds a set of attributes to its holder. In fact, attribute certificates have been designed to be used in conjunction with identity certificates, i.e. PMI and PKI are linked by information contained in the ACs and PKCs. For example the holder field in an AC contains the serial number and issuer of a PKC. The attribute certificate, identity certificate and their relation are depicted in Figure 1.

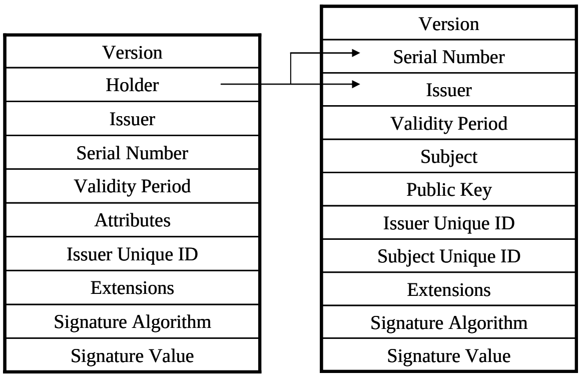

*Figure 1: Relation between attribute and identity certificate*

In PMI, the entity that digitally signs an AC is called an Attribute Authority (AA). The trusted root of a PMI is called Source of Authority (SOA). When a user's authorisation permissions need to be revoked, an AA will issue an Attribute Certificate Revocation List (ACRL) containing the list of ACs no longer to be trusted. There are two primary models for distribution of ACs: the "push" and "pull" model. In "push" model, the client supplies his AC to a server at the time of request. The "push" model is suitable when the client's rights should be assigned within the client's "home" domain. In the "pull" model, the server retrieves the client's AC from an AC repository. The "pull" model is suitable when the client's rights should be assigned within the server's domain.

ACs may be used with various security services, including access control, data origin authentication, and non-repudiation. In our work we use ACs to store the authorisation information, e.g. users' roles and access control policies.

### *2.2 – XML Signature and XML Encryption*

Both XML Signature [6] and XML Encryption [7] are W3C proposed recommendations. XML Signature provides syntax for representing signatures on digital content along with procedures for computing and verifying such signatures. XML Signatures can be applied to any kind of data in any format. XML Signature lets a user sign specific portions of the XML tree rather than the complete document. XML Encryption specifies a format and processing for encrypting data in XML. With XML Encryption, different nodes of an XML document can be encrypted with different keys, while some nodes are left in plain text.

## 3. Cross security access control framework

The Cross Security Access Control Framework (CSACF) is a framework that provides cross-organisational access control. Its architecture is depicted in Figure 2 and is composed of two independent parts: Access Control Engine (ACE) and Credential Service Centre (CSC). The ACE is responsible for access control. The basic components of ACE are Access control Enforcement Function (AEF) and Access control Decision Function (ADF). The CSC is responsible for authentication and obtaining users' attributes that are used for access control. The basic components of CSC are Authentication Service (AuthS) and Attribute Service (AttrS).

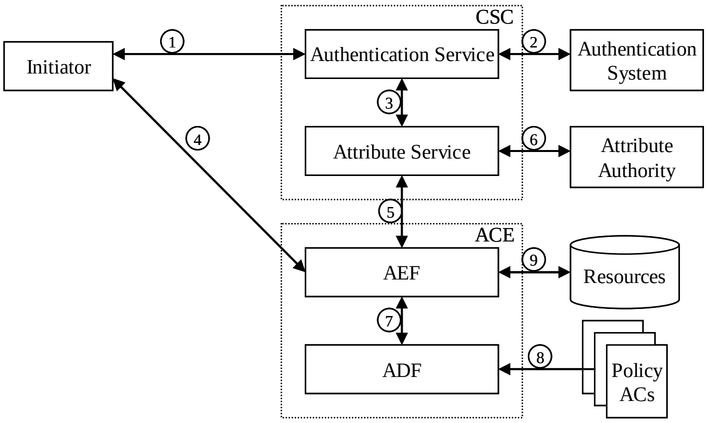

*Figure 2: Cross security access control framework*

### *3.1 – Access control engine*

Role-based access control (RBAC) emerged rapidly in the 1990s as a proven technology for managing and enforcing security in large-scale enterprise wide systems [1, 2]. In RBAC, access rights are associated with roles, and users are assigned appropriate roles thereby acquiring the corresponding permissions. It can provide more flexibility to security management over the traditional approach of using user and group identifiers.

We have developed a RBAC system with X.509 attribute certificates. The ACs used in the system can be classified into two categories; namely role ACs that store users' roles and policy ACs that store authorisation policies. The authorisation policies specify which roles have what rights on various targets. All the access control decisions are made based on the authorisation policies. Role and policy ACs are stored in LDAP servers. The heart of the system is an access control engine (ACE). The ACE executes the functions of authentication and authorisation, and then accesses the targets on behalf of the user.

Our access control framework conforms to the basic principle of ISO 10181-3 Access Control Framework that is defined by the Open Group [8]. This framework separates authentication from authorisation, and comprises of four components: Initiator (e.g. a user), Target (e.g. resources), Access control Enforcement Function (AEF) and Access control Decision Function (ADF). After passing authentication, the initiator submits access request that specifies an operation to be performed on a target. The AEF mediates access request, it submits decision requests to ADF. ADF decides whether access requests should be granted or denied based on the user's roles and access control policies. Finally, AEF enforces access control decisions made by ADF. More information about the ACE may refer to [9].

### *3.2 – Authentication service*

Authentication Service (AuthS) performs the functions related to user authentication. It can accomplish three tasks. The first is redirecting the user to his origin site for authentication. The second is connecting to local authentication system so that the user is authenticated at his origin site, and creating a handle that is used to retrieve attributes about the user. The third is forwarding the user's handle back to the destination site and performing impersonation check to the received handle. There may be multi-CSC between the destination site and the origin site. An institution must register to a CSC in order to get its service. All the exchanged messages are signed and then encrypted.

### *3.3 – Attribute service*

Attribute Service (AttrS) performs the functions related to retrieving users' attributes. It can accomplish three main tasks. The first is interacting with attribute authority to get attributes about a user, and then map one domain's attributes to another domain's attributes. The second is issuing an X.509 attribute certificate that holds the user's privileges. The third is forwarding the user's AC back to the requester through mutual authentication over SSL.

### *3.4 – Authentication and authorisation sequences*

There are four types of messages involved in the information exchanges. They are authentication request, handle response, attribute request and attribute response. In a generic application scenario, i.e. two sites involved, the CSACF acts as follows (step number relates to Figure 2):

- A user connects to an ACE-protected web site, and is redirected to the destination site AuthS for authentication. (step 1)
- The destination site AuthS finds the user's origin site and redirects him to his origin site AuthS with an authentication request message (not show in the Figure 2). (step 2)
- The origin site AuthS authenticates the user and creates a handle (gets from AttrS). This handle is encapsulated in a handle response message. (step 2 and step 3)
- This user is redirected back to the destination site AuthS with the handle response message. After impersonation check, he is redirected to the ACE. (step 4)
- The AEF sends an attribute request message to the destination site AttrS according to the user's handle. (step 5)
- The destination site AttrS contacts user origin site AttrS (not show in the Figure 2) for the user's attributes. The origin site AttrS gets user's attributes and encapsulates them in an X.509 AC, and sends it back to the requester via attribute response message. (step 6)
- Based on the user's access request and his attributes, the AEF submits a decision request to the ADF. (step 7)
- The ADF makes an access decision based on the access control policies. (step 8)
- The AEF enforces the decision made by ADF, either accesses to the target on behalf of the user or refuses this request. (step 9)

## 4. Team and task based RBAC model

Role-based mechanisms usually provide a sufficient way to establish access control in most information systems. However, passive permission assignment cannot efficiently support the dynamic aspects of many modern information systems. A variety of access control models have been developed in response to the requirements of active access control system [10, 11, 12, 13]. Motivated by the requirements of authorisation management in collaborative environments such as virtual organization authorisation management, we have developed a TT-RBAC access control model.

TT-RBAC as depicted in Figure 3 and consists of five sets of entities called users, roles, permissions, teams and tasks, as well as a collection of sessions. This approach is based on the integration of RBAC [1], team and task concepts. It has a layer structure with two layers, one is the RBAC layer; the other is the team-task layered that is used to group users and permissions. The team-task layer is placed on the top of RBAC, and can be integrated with current RBAC systems.

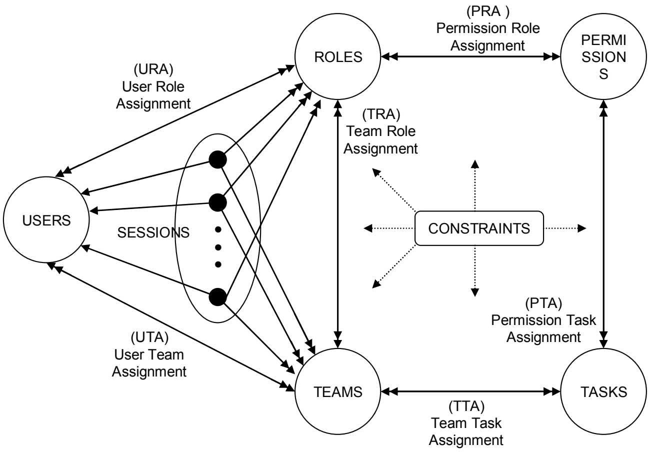

*Figure 3: Team and task based RBAC model*

In TT-RBAC model, the relations between the users, roles and permissions are the same as in the basic RBAC model. Team is a group of users in specific roles with the objective of accomplishing a specific task. A set of roles are assigned to a team. Task is a fundamental unit of business work or business activity. Each task is related to a set of permissions that are needed to finish this task. The assignment relation between team and task is many to many. In order to finish a task, a team must have enough privileges, i.e. roles. For a team member, only those roles that have been assigned to the team can be activated. His final permissions are also filtered by the permissions of the task assigned to the team. Context based permissions can also be added to a task, e.g. time.

The TT-RBAC model can give system administrators more flexibility in privilege management. Consider a collaboration scenario; several companies cooperate to design a product. One company may organize its partners into a team and assign the role of “designer” that has the permission of “view” and “edit”. This company also creates a task that specifies which parts can be operated by which actions, and assigns this task to the team. If only the permission of “view” is assigned to the task, then the team members, i.e. the partner companies can only view the design. If another task has the permission of “edit” assigned to the same team, then the partners have the “edit” right to that task. The local users through its RBAC system still have the full privileges of the role “designer”.

## 5. Support virtual organization management

Virtual Organization (VO) is a dynamic collection of resources and users unified by a common goal and potentially spanning over multiple administrative domains [15]. VO may apply some common policies about how its users can access the resources assigned to the VO, but each organisation will typically retain ultimate control over the policies that govern access to its resources. The dynamic and multi-institutional nature of these environments introduces challenging security issues that demand new technical approaches. Since VO resources and users are located within multiple organizations, a key problem associated with the formation and operation of distributed virtual organizations is that how to authenticate the users and enforce the common policies. We will describe how these issues are addressed by our CSACF and TT-RBAC through an example.

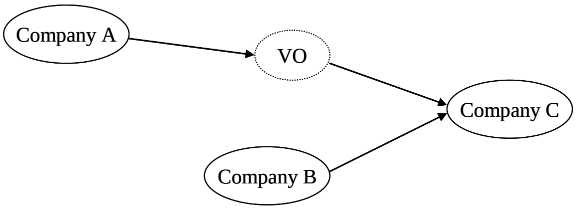

*Figure 4: An example of virtual organization*

As depicted in Figure 4, there are three physical organisations, i.e. company A, B and C, and one virtual organization, i.e. VO. Company C is a resources provider. Company A and B are identity providers. Company B and VO register in company C, and their users can be authenticated and authorised through the process described early, and then access the resources in company C. Company A does not register in company C, but it registers in VO. Because our framework supports transitivity, so company C can authenticate a user in company A and get his attributes through VO. VO can enforce some common policies to the user's attributes when it forwards them to the destination site.

The main purpose of CSACF is to securely transfer a user's attributes between the user's origin site and destination site that the user wants to access. But simply transferring attributes is not enough, generally one organisation cannot directly use the attributes (roles) defined in anther organisation. To fix this problem we introduce role contexts which are used for grouping and managing roles. Roles have functionalities only in a certain role context. One role context may cover several organizations, and one organization may contain several role contexts. Transferring privileges held by roles between different role contexts needs role mapping mechanism to do the privileges interpretation.

The main reason for introducing role contexts is for defining virtual role contexts that are used as interfaces, through which the participating organizations can exchange privileges. Each virtual role context, like regular RBAC, contains specific roles and role hierarchy. Virtual role context participants contractually agree to its legal role structure, and define the mapping from their inner role structure to the role structure in virtual role context. With the help of the intermediary layer, two organizations' role structures indirectly interact with each other. The changes in the participating organizations do not affect the intermediary context security infrastructure. Instead, these changes are confined to modification of the mapping from one organization's role structure to virtual role context's role structure. Adding or removing an organization from the VO does not affect the VO's security infrastructure.

The RBAC of a VO runs in a virtual role context, similar to normal role contexts, the team-task layer can be added on the top of the RBAC of a VO. Through TT-RBAC the VO can provide access control at different levels of granularity on which it operates. At the VO participant side, the users and privileges related to collaborative activity can be organised into teams and tasks, and assigns teams to tasks. Through different kinds of assignment, the resource providers can flexibly manage their resources that are provided to the VO.

## 6. Implementation

We have developed a prototype based on our proposed CSACF framework and TT-RBAC access control model. The implementation of the prototype uses the following software: Apache Web Server [17], Jakarta Tomcat servlet container [18], IBM XML Security Suite [19], IAIK-JCE [20], MySQL [21] and OpenLDAP [22]. All the software either are open-source software or free download for evaluation. Currently, this prototype is used for demonstrating the system's feasibility.

## 7. Related works

There are three important related works. They are the Internet2 Shibboleth project [14], the Community Authorisation Service (CAS) [15] and the Virtual Organization Membership Service (VOMS) [16].

*7.1 – Shibboleth*

Shibboleth is a cross-institutional authentication and authorisation service for access control to Web-accessed resources. It provides a secure framework for one organization to transmit attributes about a web-browsing individual across security domains to another institution.

The major difference between Shibboleth and CSACF is that in Shibboleth the authentication and authorisation service only happen between two institutions. Whereas the CSACF supports transitivity, it can pass the authentication or authorisation service to next entity until a user is authenticated or authorised. The CSACF supports transitivity is because our primarily target is developing a mechanism used for exchanging authentication and authorisation information among big organizations that normally have hierarchical structures, e.g. governments. Through transitivity sub-organizations can be easily added or removed from the system. This issue is not addressed by Shibboleth.

*7.2 – CAS and VOMS*

The CAS is developed by the Globus project for Grid environments. Their authorisation model allows a resources site to grant a community access to its resources, and the authorisation server for that community to grant access to the community members. The VOMS is another solution to authentication in a GSI-enable Grid. The VOMS server is run by a virtual organization and supplies authorisation information about its own members. CAS and VOMS are similar architecturally in that both issue policy assertions to a user that the user then presents to a resource for the purpose of obtaining VO-issued rights. The primary difference between the two systems is the level of granularity at which they operate.

There are two major differences between CSACF and CAS or VOMS. In CAS and VOMS the users' authorisation information is managed in a central server. Whereas in CSACF, the users' authorisation information is managed in the users' origin sites, but through the transitivity the CSACF can perform common policies on the users. The second difference is that in CAS and VOMS users' authentication is done at resource provider sites; in CSACF users are authenticated at their origin sites. The major benefit of CSACF is the resource provider does not have to maintain partner's users. When the number of users gets large, the burden of managing identities for foreign users can be high.

## 8. Conclusions and future works

With the CSACF we can get two major benefits. The first is independent organizations can share their resources without the burden of managing the users who belong to other organizations. They only need to manage the trust relationships with other organizations. The second is the CSACF supports virtual organization management. Independent organizations can be freely added or removed from the VO, and these modifications do not influence the security infrastructure of the VO or other participating organizations. The TT-RBAC can provide flexible and different level granularity access control, its layer structure also makes it easy to integrate with exist RBAC systems.

Future works will be in two directions. One is continue to improve this framework, especially the virtual organization management. Some GUI management tools will also be developed. The other is about the application of the framework. We have developed a web portal that aims to help the Small and Medium-sized Enterprises (SMEs) improving their supply chain management. At the moment this system needs the users manually input the data. We are planning to use this framework to automatically connect enterprises' computer systems so that within a connected supply chain the production schedules, product data and status information can be synchronized and made available throughout the chain.

## 9. Acknowledgement

The authors would like to thank the One North East RDA for their financial support for B2B Manufacturing Centre. This centre is in collaboration between the Warwick Manufacturing Group at the University of Warwick and the University of Teesside. http://www.b2b-mc.co.uk.

University of Warwick is partner of the EU-funded FP6 Innovative Production Machines and Systems (I*PROMS) Network of Excellence. http://www.iproms.org.

## References

[1] R.S. Sandhu, E.J. Coyne, H.L. Feinstein, C.E. Youman, Role based access control models, IEEE Computer, 29 February 1996.

[2] David F. Ferraiolo, R.S. Sandhu, Serban Gavrila, D. Richard Kuhn and Ramaswamy Chandramouli, Proposed NIST Standard for Role-Based Access Control, ACM Transactions on Information and Systems Security (TISSEC), Volume 4, Number 3, August 2001.

[3] ITU-T Rec. X.509 ISO/IEC 9594-8, The Directory: Public-key and Attribute Certificate Frameworks, May 3, 2001.

[4] R. Housley, W. Ford, W. Polk, D. Solo, Internet X.509 Public Key Infrastructure Certificate and CRL Profile, January 1999 , http://www.ietf.org/rfc/rfc2459.txt.

[5] S. Farrell, R. Housley, An Internet Attribute Certificate Profile for Authorization, April 2002, http://www.ietf.org/rfc/rfc3281.txt.

[6] The World Wide Web Consortium (W3C), XML-Signature Syntax and Processing W3C Recommendation, 12 February 2002, http://www.w3.org/TR/xmldsig-core/.

[7] The World Wide Web Consortium (W3C), XML Encryption Syntax and Processing W3C Recommendation, 10 December 2002, http://www.w3.org/TR/xmlenc-core/.

[8] ITU-T Rec. X.812(1995)|ISO/IEC 10181-3:1996, Security frameworks in open systems: Access control framework.

[9] W. Zhou, C. Meinel, Implement role based access control with attribute certificates, The 6th International Conference on Advanced Communication Technology (ICACT2004), Volume 1, P. 536-541, Korea, February 2004.

[10] R. K. Thomas, Team-based access control (TMAC): A primitive for applying role-based access controls in collaborative environments, Proceedings of the Second ACM Workshop on Role-based Access Control, P. 13-19, Fairfax, Virginia, November 1997.

[11] R. K. Thomas, R. Sandhu, Task-based authorization controls (TBAC): A family of models for active and enterprise-oriented authorization management, Proceedings of the IFIP WG 11.3 Workshop on Database Security , P. 166-181, Lake Tahoe, California, August 1997.

[12] C. K. Georgiadis, I. Mavridis, G. Pangalos, R. K. Thomas, Flexible Team-Based Access Control Using Contexts, Proceedings of the ACM Symposium on Access Control Models and Technologies, May 2001.

[13] E. Cohen, R. K. Thomas, W. Winsborough, D. Shands, Models for Coalition-based Access Control (CBAC), Proceedings of the ACM Symposium on Access Control Models and Technologies, June 2002.

[14] M. Erdos, S. Cantor, Shibboleth-Architecture DRAFT v05, http://shibboleth.internet2.edu/.

[15] L. Pearlman, C. Kesselman, V. Welch, I. Foster, S. Tuecke, The Community Authorization Service: Status and Future, http://www.globus.org/security/CAS/GT3/.

[16] R. Alfieri, R. Cecchini, V. Ciaschini, L. dell'Agnello, A. Gianoli, F. Spataro, VOMS: an Authorisation System for Virtual Organizations, Presented at the 1st European Across Grids Conference, Santiago de Compostela, February 13-14, 2003.

[17] Apache Web Server, http://www.apache.org/.

[18] Jakarta Tomcat servlet container, http://jakarta.apache.org/tomcat/.

[19] IBM alphaworks, XML Security Suite, http://www.alphaworks.ibm.com/tech/xmlsecuritysuite.

[20] IAIK-JCE, IAIK Java Cryptography Extension (IAIK-JCE), http://jce.iaik.tugraz.at/index.php.

[21] MySQL, the open source SQL database server, http://www.mysql.com/.

[22] OpenLDAP, the open source Lightweight Directory Access Protocol (LDAP), http://www.openldap.org/.