Semantic Interoperability of Authorizations

Antonio Maña, Mariemma I. Yagüe, Francisco Sánchez

Computer Science Dept. Univ. of Malaga.
E.T.S.I.Informatica. Campus de Teatinos.
29071 Malaga, Spain
{amg, yague, cid}@crypto.lcc.uma.es

Abstract. The shift from paper documents to their respective electronic formats is producing important advantages in the functioning of businesses and Public Administrations. However, this shift is often limited to the internal operation of each entity because of the lack of security in the electronic communication mechanisms. Traditionally, these entities have managed their Local Area Networks (LANs) or even Virtual Private Networks (VPN) as isolated islands, where local identity-based authorization schemes were appropriate. But, the trend towards paperless procedures leads to the need for these entities to interoperate. As an advance, extranets were proposed to connect entities that share common goals in a way that automates their administrative interactions using Internet technology. However, the limited authorization and access control capabilities provided by extranets is a mayor drawback for their application in open and heterogeneous scenarios. Trust appears as the main issue to address in order to achieve secure interoperation of different independent entities. This paper presents a solution to this problem, based on the use of Privilege Management Infrastructures (PMIs) and the semantic description of the different authorization entities.

1 Introduction

Today, being able to procure and provide access to information is a defining characteristic of successful companies. And as companies open up their networks to partners and other third-party users to share information, security has become more important than ever. Companies require comprehensive security systems that allow controlled access.

Therefore, entities need to be able to limit access so that only permitted users have access to certain resources. This means that traditional encrypted tunnels such as VPNs, which require that everyone at both ends is trusted, are inadequate for third-party access. When it comes to sharing information with outsiders, companies need to provide one-way directed access to shared information.

An Extranet is a communication network connecting entities that share common goals in a way that automates their administrative interactions. When properly designed and implemented, extranet systems can be highly effective in improving cross-entity information flows. Extranet services use existing Internet infrastructure, which

makes extranets far more economical than proprietary networks. However, the limited authorization and access control capabilities provided by extranets is a mayor drawback for their application in open and heterogeneous scenarios.

Trust appears as the main issue to address in the design of a platform allowing secure interoperation of different independent entities. Many distributed application scenarios such as e-commerce, e-business, e-government, grid computing or web services can benefit from the services of such platform. Some important characteristics of these scenarios are:

- Independence of Authorities. The authorities, as well as the rules governing the functioning of each party are usually predefined and must be independent of others and under control of the legitimate authority.
- Attribute-based access. Usually, access is offered to previously unknown users (individual citizens and members of other entities). Knowledge of their identities, provided by a Public Key Infrastructure (PKI) is not sufficient in order to interact with them.
- Heterogeneity. In open distributed systems we deal with a large number of stakeholders or owners of resources with very different policies and interests, but also with a large number of previously unknown clients, with very different profiles and interests. Moreover, resources under control are intrinsically heterogeneous in type, format, origin, validity, etc. Consequently, the security requirements and access control criteria are also very disparate. As a result, it is impossible for administrators to foresee a fixed role-based structure of the users.
- Flexibility. A high degree of flexibility is required because of the heterogeneous nature of the resources (data and services), access criteria and users. In fact, flexibility appears as one of the most important goals to achieve. The model must be flexible enough to be applicable in different scenarios with few or no changes.
- Scalability. The scalability of the scheme is very important. Therefore, a fully distributed scheme is mandatory. Furthermore, due to the large amount of resources, it is important to be able to determine access conditions automatically, based on their associated semantic information.
- Interoperability. In these scenarios, it is not possible to predict the interactions with other parties. Typically, these interactions will take place only occasionally and parties will frequently be related by a few transactions in common. Because we are dealing with security-sensitive systems, it is essential to guarantee that the interoperation with other parties does not introduce any security weakness.
- Dynamism. This characteristic is essential in most of our targeted scenarios, where the existence of highly dynamic resources is frequent. The access control model must be capable of adapting to frequent changes in access control criteria, client attributes, environment conditions, resources available, etc. To avoid management overload due to the control of changes, the model must adapt in a transparent and automatic way to these changes.

The previous list of characteristics poses important challenges on the underlying security mechanisms and especially in authorization and access control systems. Paradoxically, it is frequent for access control and authorization mechanisms in distributed systems to rely on centralized security administration. In fact, existing solutions for distributed authorization and access control do not provide the flexibility and manageability required. Summarizing, it is clear that new solutions are required to address the

security needs of some of the new distributed applications, as it is the case of e-government, but also of web services, electronic commerce or grid computing.

The paper is organized as follows. Section 2 summarizes some background and related work. Section 3 describes the fundamentals of our proposal and outlines the system operation and implementation. Finally, section 4 summarizes the conclusions.

2 Background and Related Work

The problem of interoperation among autonomous applications has been extensively studied. For instance, it received significant attention during the late 1980s and early 1990s in the framework of the research in federated databases. The objective of this work was to resolve the structural differences among disparate database schemes. However, practical federated database systems failed because of the problem of semantic heterogeneity [1]. This problem appears when different applications mean different things by similar terms. Semantic heterogeneity is closely tied to the context-dependent interpretations of the concepts represented. Although interoperability of applications have been extensively studied (i.e., CORBA, DCom, Java, …), not much work has been done in semantic interoperation of applications. We have just to mention Web Services, providing interoperability among components with semantic heterogeneity. In this sense, a recent approach is to consider semantic aspects, applying concepts of the Semantic Web, such as ontology, to Web Services [2].

When considering the security requirements of different distributed applications, authorization often emerges as a central element in the design of the whole security system. The reason for this is that authorization is the source of the trust chain. Therefore, many security properties are determined by the flexibility, trustworthiness and expressiveness of the authorization scheme.

The problem of authorization is well known and has been studied for a long time. However, the advances in communication networks have fostered the evolution from centralized to distributed systems and applications. This situation requires the creation of new authorization models.

Currently, most authorization approaches are based on locally-issued credentials (containing attributes or privileges) that are linked to user identities. This type of credentials presents many drawbacks. Among them we highlight: (a) they are not interoperable; (b) the same credentials are issued many times for each user, what introduces management and inconsistency problems; (c) credentials are issued by the site administrator; however, in most cases, the administrator does not have enough information or resources to establish trustworthy credentials; and (d) they are tightly dependent on the user identity. But, in practice, it is frequent that the identity of the user is not relevant for the access decision. Sometimes it is even desirable that the identity is not considered or revealed. Furthermore, in systems based on identity, the lack of a global authentication infrastructure (PKI) forces the use of local authentication schemes. In these cases, subscription is required and users have to authenticate themselves to every accessed source.

Summarizing, when these local schemes are applied to distributed systems, especially to open ones, they result very limited and inconvenient. The most relevant prob-

lem when local schemes are applied to open distributed systems is the lack of interoperability. It is not reasonable to expect that heterogeneous systems for different purposes and under control of different stakeholders will be able to define a common homogeneous set of authorization criteria. Other problems are that (i) security administration is complex and error prone; (ii) allocation of policies to resources is explicit and static; (iii) access control criteria are defined either explicitly or on the basis of the location of the contents; (iv) schemes are based on user identity; and (v) access policies are dependent on the administrator of the server where the resource resides.

Based on asymmetric cryptography, digital certificates are used to bind a public key to some information. Identity certificates are the most common type of digital certificates in use today. These are used to bind identity information to keys. On the other hand attribute certificates bind attributes to keys. Therefore, attribute certificates provide means for the deployment of scalable access control systems in the scenarios that we have depicted.

The latest ITU-T X.509 recommendation [3] standardizes the concept of attribute certificate, and defines a framework that provides the basis upon which a Privilege Management Infrastructure (PMI) can be built. Precisely, the foundation of the PMI framework is the Public Key Infrastructure (PKI) framework defined by ITU [4]. This new recommendation defines a new type of authority for the assignment of privileges, the *Attribute Authority* (AA), while a special type of Authority, the *Source of Authority* (SOA), is settled as the root of delegation chains. One important point is that PKI and PMI are separate infrastructures in the sense that either structure can work on its own, or to be more precise, they can be established and managed independently.

3 Semantic Integration of PMIs in the Access Control System

The aforementioned problems related to the use of local schemes, lead us to consider a fully distributed approach. Accordingly, the inclusion of external authorization entities in the access control scheme facilitates the separation of responsibilities, enhances the security levels, and makes credentials interoperable among different access control systems.

By considering attributes to be the basis of the access control model we can develop a very flexible and open model that fits most scenarios. In fact, MAC [5], DAC [6] and RBAC [7] schemes can be specified using the attribute-based approach. In [8] we proposed a modular and dynamic approach based on the separate specification of the access control criteria and the rules of allocation of policies to resources. Additionally, the use of attributes as the central element of the model is complemented with the use of metadata to represent the semantics of the different elements in an access control system.

This new model is called Semantic Access Control (SAC) [9] because it is based on the semantic properties of the resources to be controlled, properties of the clients that request access to them, semantics about the context and finally, semantics about the attribute certificates trusted by the access control system. In SAC, access policies are expressed in terms of sets of attributes instead of users or groups. For interoperability and security reasons, client attributes must be digitally signed by a trusted certification

entity external to the access control management system. Therefore, attribute certificates are used to prove that users meet the required attributes. This scheme scales well in the number of users and also in the number of different attributes used by the access control system.

In the development of the SAC model, we have considered the operation of several independent access control systems and authorization entities. In SAC, the access control to resources is independent of their location. The identification of the user or client is not mandatory. The independence of the authorization function is the key to the interoperability because it allows attributes to be safely communicated avoiding the necessity of being locally emitted by each system administrator.

Additionally, this approach avoids the registration phase of the client, and the evaluation and issuance of a client attribute repeatedly for each access control system. Finally, this scheme promotes the operation of specialized authorization entities with deep knowledge of the domain of the attribute to attest, enhancing the practical security and privacy levels of the system.

In access control schemes based on attribute certificates, the semantics of the policies depend heavily on the semantics of the attribute certificates. For this approach to be secure, a mechanism to establish the trust between these access control systems and the authorization entities is required. We have addressed this problem using semantic information about the certifications issued by each authorization entity. This mechanism is the core of the semantic integration of the PMI, which is essential in order to achieve interoperability in these scenarios. Furthermore, this integration solves the problems of separation of duties, scalability and interoperability. The main reason for this is the necessity of understanding the specific security requirements, as well as the semantics of the attribute certificates managed. As we will show, a new metadata model, called *Source Of Authorization Description* (SOAD), has been created for this purpose. The SOAD metadata model conveys the semantics of the attribute certificates providing semantic information that will be essential in the process of access decision.

The semantic information about the attribute certificates issued by each SOA also assist the security administrator through the process of specification of the access control policies, as it conveys the meaning of each attribute. Additionally, the semantic information represented by the SOAD model enables the automatic detection of inconsistent policies, through a *Semantic Policy Validator* (SPV) tool developed with this objective. The SPV makes inference processes using the rules defined in the SOAD documents.

The ability to perform a semantic validation of access control policies is an essential design goal of the SAC model. Both the *Semantic Policy Language* (SPL) defined in SAC and the semantic descriptions of the certificates issued by each SOA (conveyed by SOAD documents) are designed to serve this objective. The semantic validation ensures that the policies written by the security administrator produce the desired effects. The SPV can perform three types of validations:

1. Test Case Validation: Given a request to access a resource and a set of attribute certificates, this algorithm outputs the sets of attribute certificates needed for accessing that resource. Most of times, this feature will be used to check that a set of attribute certificates is incompatible with the access criteria for that resource. For instance, the administrator of our university can use this validation to guarantee that

it is not possible for a student to access a given resource (i.e., documents containing marks). During the validation process, the SPV generates the sets of attribute certificates that are not excluded by the input set, and checks the generated ones against all possible combinations of attribute certificates that grant access to the resource.
2. Access Validation: Given a request to access a resource, this algorithm outputs the sets of certificates that grant access to that resource. For this validation process, the SPV generates the policy for the resource and all sets of attribute certificates equivalent to those required by the policy.
3. Full Validation: The goal of this process is to check which resources can be accessed given a set of attribute certificates. Therefore, SPV generates the policy for each resource and, afterwards, all attribute certificates that can be derived from the input set of attribute certificates. Finally, it informs of every resource that can be accessed using the input attribute certificate set.

3.1 The SOAD Metadata Model

The set of SOADs represents the semantic description of the PMI. SOAD documents are digitally signed [10] XML-Schema instances expressing the different attributes certified by each SOA, including names, descriptions and relations. Such descriptions are the basis for building a mechanism to provide client applications (i.e., access control system) with knowledge about the meaning of the attributes issued by each SOA.
SOAD documents include a reference to the SOA described (`SOA_ID`), the declaration of the attribute certificates issued by that SOA and the relations between these attribute certificates.
The attribute declaration section consists in a set of `SOA_Attribute` elements. Each one of these elements defines an attribute certificate issued by the SOA referenced by `SOA_ID`, described by a name (`AttributeName`), a value (`AttributeValue`) and the signer of the certificate (`SOA_ID`).
Relations between attributes are expressed using `SOARule` elements. Each relationship is represented by a logical rule where both, the premise and the conclusion are set of attribute certificates, combined by a logical operator indicating the relation among these certificates. Each premise comprises certificates (`SOA_Attribute` elements) issued by the SOA being described or external ones. The conclusion comprises `AttributeSet` elements composed by attribute certificates issued by the SOA being described. In this way, the SOA can declare any kind of relationship among the certificates it issues and the certificates issued by other SOAs. Additionally, the client applications (i.e. access control systems) can control which relationships they accept and under which conditions.

3.2 Implementation

The system is implemented using three different applications: SOAD Manager, SOAD Server and SOAD Client.

The SOAD Manager is a Java™ application that allows SOAs to create SOAD documents. It has advanced edition capabilities that facilitate the definition of SOADs in an intuitive and easy way.

The SOAD Server is responsible for the publication and distribution of SOADs. This application implements an interface to allow SOAs to upload their SOADs and another one to allow clients to locate and retrieve the SOADs they need.

The principal purpose of the SOAD Client is to allow client systems to locate and retrieve SOADs from SOAD Servers. Additionally, it offers a subset of the SOAD edition capabilities available in the SOAD Manager. This application is also used to automate and tailor the process of refreshing the SOAD.

Figure 1 shows screenshots of the SOAD Manager and the SOAD Client Applications.

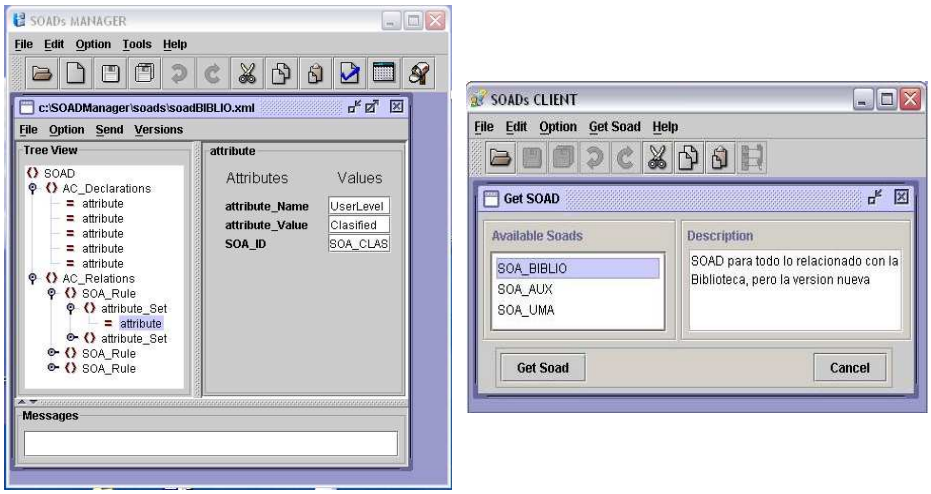

Fig. 1. SOAD Manager and SOAD Client Applications.

3.3 System Operation

Figure 2 depicts the flow of SOADs from originating SOAs to client access control systems. Each SOA creates SOADs to describe the attribute certificates it issues. These SOADs are then made available to client systems in one or more SOAD Servers. When necessary, clients retrieve the SOADs of the SOAs they trust from SOAD Servers. Clients are then able to process the received SOADs locally in order to limit the attributes and relations they accept from each particular SOA. These local SOADs are then used in the computation of the access control decisions. Associated to each local SOAD, clients can set different parameters to control when they must be refreshed, where to refresh it from, etc.

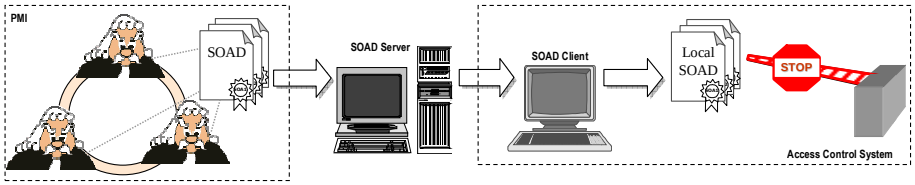

Fig. 2. Flow of SOADs

3.4 A Case Study in E-Government

We use *e-government* as the scenario to illustrate our proposal, because it is one of the most relevant and interesting of the aforementioned applications. The term *e-government* is often defined as the use of information and communication technologies to support and improve the activities of public administrations. This definition means that, to some degree, e-government is not a new issue. But the real potential of e-government lies on the possibility of substituting traditional paper-based procedures by their electronic versions, achieving what have been called “paperless systems”, implementing the necessary mechanisms to achieve trustful and transparent interoperation between the different parties involved (government agencies, citizens, private businesses and organizations, other arms of government and even foreign governments).

Figure 3 shows a typical e-government scenario. In particular, we use a representative example, involving the interaction of several government agencies, some private business and organizations as well as individual citizens. Consider the case of a tax collection agency starting a judicial process against a citizen, due to unpaid taxes. This process implies different exchanges of sensitive information among different parties. The judge from the corresponding court can request information to the Town Hall cadastral agency about the cadastral value of the buildings belonging to the citizen accused, to the bank about the drawing account of the citizen, to the police department about the criminal records of the accused citizen, etc. On the other hand, the accused citizen and his lawyer may request information about the stage of the judicial process.

Interoperability is an essential requirement in this scenario. The existence of different government agencies that need to cooperate, and the special security requirements inherent to these transactions, makes this problem very complex. In order to securely perform this information exchange, each party has to be recognized as an official entity with jurisdiction to do the intended task. Identity-based schemes are not always the best option. Every single piece of information in the different sites has different requirements making it impossible in practice to assign privileges to identities. In this case, the authorization of the other party (i.e., the examining judge) is based on some specific properties or attributes (to be the judge assigned to the process), not on identity (to be Mr. Jones). These properties represent the conditions that the user (or the client agency requesting the service) has to fulfil in order to access the information, that is, the access control policy.

In this scenario, an access control model based on attributes is very appropriate and can provide simple solutions to such problems. But, the real advantage comes when attributes become interoperable. To achieve such interoperability, we must satisfy two

important conditions. First, attributes must be come from trusted sources, and second, we must be able to understand what those attributes mean.

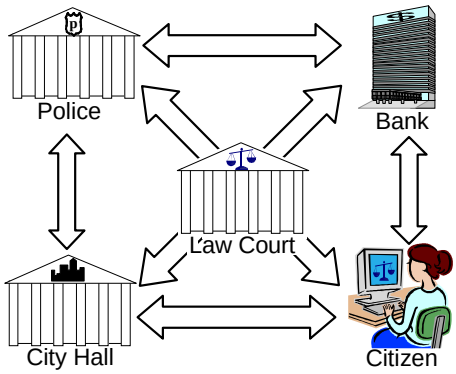

Fig. 3. An e-government scenario

Our proposal fulfils the requirements of this kind of transactions, providing fine-grained access control, enabling the secure communication among government agencies and assigning the attestation of attributes to trusted entities with an in-depth knowledge of the properties to attest (SOA of the Policy Department, SOA of the Law Court, etc.).

4 Conclusions

The possibility of automating the processing of semantic information is a big challenge for the resolution of many relevant problems, as is the case of semantic interoperability. The objective of this work is to reach semantic interoperability through semantic integration in distributed environments, where remote and heterogeneous parties must exchange information in a controlled manner. We think the development of mechanisms for the semantic integration in distributed environments where heterogeneity is common, implies the development of semantic models supported by metadata infrastructures. In the case we are concerned with, the kind of information to be described is essential to maintain the secure, trustful and transparent interoperation of the different parties involved in electronic transactions.

We have presented a solution for the interoperability of authorizations (attribute certificates) based on the description of their semantics. This solution provides a foundation to build interoperable access control systems with external and independent authorization services. Additionally, the semantic modelling of the authorizations enables interesting possibilities, such as the semantic validation of access control policies.

References

1. Sheth, A., Larson, J.: Federated Database Systems for Managing Distributed, Heterogeneous and Autonomous Databases. ACM Computing Surveys, 22(3) (1990) 183 - 236
2. World Wide Web Consortium: Semantic Web Services Interest Group. Retrieved January 2003 from http://www.w3.org/2002/ws/swsig/
3. International Telecommunication Union (2000). ITU-T Recommendation X.509. Information technology – Open systems interconnection – The Directory: Public-key and attribute certificate frameworks. Technical Cor. 3 (02/03) [Electronic version] http://www.itu.int/rec/recommendation.asp?type=items&lang=e&parent=T-REC-X.509-200302-P!Cor3
4. International Telecommunication Union (1997). ITU-T Recommendation X.509, Information Technology – Open systems interconnection – The Directory: Authentication Framework. 1997. [Electronic version] http://www.itu.int/rec/recommendation.asp?type=items&lang=e&parent=T-REC-X.509-200302-T!Cor5
5. Qian, X., Lunt, T.F.: A MAC policy framework for multilevel relational databases. IEEE Transactions on Knowledge and Data Engineering, 8(1) (1996) 1-14
6. Baraani, A., Pieprzyk, J., Safavi-Naini, R.: Security In Databases: A Survey Study. Retrieved September 2003 from [http://citeseer.nj.nec.com/baraani-dastjerdi96security.html. (1996)
7. Sandhu, R., Ferraiolo, D., Kuhn, R.: The Nist model for role-based access control: Towards a unified standard. In Proceedings of 5th ACM Workshop on Role-Based Access Control. Berlin, Germany (2000)
8. López, J., Maña, A. and Yagüe, M.I: XML-based Distributed Access Control System. Lecture Notes in Computer Science, Vol. 2455. Springer-Verlag (2002)
9. Yagüe, M.I., Maña, A., López, J., Pimentel, E., Troya, J.M.: A Secure Solution for Commercial Digital Libraries. Online Information Review Journal, 27(3): 147-159. Emerald Publishers (2003)
10. World Wide Web Consortium: XML-Signature Syntax and Processing (2002) [Electronic version] http://www.w3.org/TR/xmldsig-core/.