A new design of Privilege Management Infrastructure for organizations using outsourced PKI

Ed Dawson[1], Javier Lopez[2], Jose A. Montenegro[2], and Eiji Okamoto[3]

[1] Information Security Research Centre,
Queensland University of Technology, Australia
e.dawson@qut.edu.au

[2] Computer Science Department, E.T.S. Ingenieria Informatica
Universidad de Malaga, Spain
{jlm,monte}@lcc.uma.es

[3] Institute of Information Sciences and Electronics,
University of Tsukuba, Japan
okamoto@is.tsukuba.ac.jp

Abstract. Authentication services provided by Public Key Infrastructures (PKI) do not satisfy the needs of many e-commerce applications. These applications require additional use of authorization services in order for users to prove what they are allowed to do. Attribute certificates have changed the way in which the authorization problem has been considered until now, and Privilege Management Infrastructures (PMI) provide the necessary support for a wide use of those certificates. Although both types of infrastructures, PKIs and PMIs, keep some kind of relation, they can operate autonomously. This fact is specially interesting for companies who have taken or will take the decision to outsource PKI services. However, outsourcing PMI services is not a good option for many companies because sometimes information contained in attribute certificates is confidential. Therefore attribute certificates must be managed very carefully and, preferably, only inside the company. In this paper we present a new design of PMI that is specially suited for those companies that outsource PKI services but still need to manage the PMI internally. The scheme provides additional advantages that satisfy the needs of intra-company attribute certification, and eliminates some of the problems associated with the revocation procedures.

1 Introduction

It is well known that by using an *authentication service* you can prove who you are. Identity certificates (or public-key certificates) provide the best solution to integrate that basic service into most applications developed for the Internet that make use of digital signatures. However, new applications, particularly in the area of e-commerce, need an *authorization service* to describe what it is allowed for a user to do. In this case privileges to perform tasks should be considered.

For instance, when a company needs to establish distinctions among their employees regarding privileges over resources, the authorization service becomes important. Different sets of privileges over resources (either hardware or software) will be assigned to different categories of employees. In those distributed applications where company resources must be partially shared through the Internet with other associated companies, providers, or clients, the authorization service becomes an essential part.

Authorization is not a new problem, and different solutions have been used in the past. However, ”traditional” solutions are not very helpful for many of the Internet applications. Those solutions are not easy to use in application scenarios where the use of identity certificates, to attest the connection of public keys to identified subscribers, is a must. In such scenarios, types of independent data objects that can contain user privileges would be of great help. *Attribute certificates* proposed by the ITU-T (International Telecommunications Union) X.509 recommendation [10] provide an appropriate solution, as these data objects have been designed to be used in conjunction with identity certificates.

The use of a wide-ranging authentication service based on identity certificates is not practical unless it is complemented by an efficient and trustworthy mean to manage and distribute all certificates in the system. This is provided by a *Public-Key Infrastructure* (PKI), which at the same time supports encryption, integrity and non-repudiation services. Without its use, it is impractical and unrealistic to expect that large scale digital signature applications can become a reality [13],[1].

Similarly, the attribute certificates framework defined by ITU provides a foundation upon which a *Privilege Management Infrastructure* (PMI) can be built. PKI and PMI infrastructures are linked by information contained in the identity and attribute certificates of every user. The link is justified by the fact that authorization relies on authentication to prove who you are.

Although linked, both infrastructures can be autonomous, and managed independently. Creation and maintenance of identities can be separated from PMI, as authorities that issue certificates in each of both infrastructures are not necessarily the same ones. In fact, the entire PKI may be existing and operational prior to the establishment of the PMI.

From the company point of view this is a very important fact. The reason is that, on the one hand, an ”identity” tends to have a global meaning; thus, identity certificates can be issued by *Certification Authorities* (CAs) that are external to the organization. If this is the case, CAs can sometimes be under the control of private companies that offer specialized external services and facilities. In some other cases, CAs are under the control of national or regional governments, which is the most typical solution when applications run inside scenarios that are related to e-government services.

However, an ”attribute” tends to have a more local meaning. Privileges are used in a more closed environment, i.e, inside an organization, or among a group of them. Therefore, there are numerous occasions where an authority entitled to attest who someone is, is not the appropriate one to make statements about what that person is allowed to do. In the case of a private company, it seems more reasonable that someone from the senior staff in the company decides on privileges and, therefore, issues a certificate containing them.

Precisely, this is the scope of the work presented here. This paper presents an attribute framework in which the PMI has been specifically designed for companies that have decided to outsource services provided by a PKI. In this case, the term "outsource" has the meaning of using an external authentication service,regardless of whether this is provided by a private organization (and hence, with some cost-per-service for the company), or by a governmental organization (a free service in many cases).

The new PMI scheme has several advantages in comparison with the scheme proposed by ITU. It makes use of a distributed architecture of authorities that satisfy the typical needs of attribute certification inside companies, avoids scalability problems associated to both extranet or company expansion, and eliminates problems associated with the revocation procedures.

The rest of the paper is structured as follows: Section 2 reviews the traditional solutions that have been used, and are actually used in many scenarios, for authorization management. Section 3 describes the initial approach of using attributes in the extensions fields of identity certificates, and why this solution is not suitable in most privilege applications. Also, this section shows how Privilege Management Infrastructures have been designed to provide a solution to those applications. In section 4 these infrastructures are studied and compared to Public Key Infrastructures, we argue about the mutual independence of the new infrastructures, and how this facilitates the outsourcing of services. Section 5 shows the new scheme we have designed that is specially suited for those companies that outsource authentication services but, because of the confidentiality of the information contained in the attribute certificates, still have to manage them internally. Finally, section 6 concludes the paper.

2 Previous solutions for authorization management

Traditional authorization schemes have mainly focused on access control, that is concerned with limiting the activities of a legitimate user within a system. Access control also assumes that authentication of the user (whatever method is used) has been successfully verified prior to enforcement of access control.

Two different schemes have been commonly used. The first one, *discretionary access control* governs the access of users to information on the basis of the users' identities and authorizations. Authorizations specify, for each individual user and each object (resource) in the system, the access rights of the user, that is, what the user is allowed to perform on the object. Each activity is checked against the access rights, which are held as access control lists within each target resource. If authorization stating the user can access the object in the specified mode exists, then access is granted, otherwise is denied.

The second one, *mandatory access control* governs access on the basis of the classification of resources and users according to security levels. Thus, access to a resource is granted if the security level of a particular user stands in accordance with the security level of that object. A classification list that is typically used in military applications is unmarked, unclassified, restricted, confidential, secret and top secret [2].

As can be seen, these schemes are suitable for authorization, but only when the access control of local resources is the problem to be solved. It is reasonable to think that management of access rights under both types of authorization policies must be done by system administrators.

A *role-based access control* scheme is an alternative solution to discretionary and mandatory schemes [6]. A role policy regulates the access of users to information on the basis of the activities that the users perform in the system in pursuit of their goals. A role can be defined as a set of actions and responsibilities associated with a particular working activity. Instead of specifying all the actions that any individual user is allowed to execute, actions are specified according to roles [15].

We can see that these solutions focus on the problem of access control. For a long time, access control has been used as synonymous to authorization. However, authorization involves many issues, for instance, group membership, role identification (collection of permissions or access rights, and aliases for the user's identity), limits on the value of transactions, access time for operations, security clearances, time limits, etc. In order to provide support to applications where authorization means something else than access control, attribute certificates become an excellent solution, as we explain in next section.

3 Authorization with Attribute Certificates: From PKI to PMI

Advantages of using attribute certificates to implement authorization have become clear. Even traditional access control solutions studied in previous section have evolved in this direction. A clear example is the integration of role-based schemes with attribute certificates. After the introduction of attribute certificates in the ITU-T X.509 recommendation [9], some proposals using them for role-based access control have been presented [8],[14].

As previously stated, one of the advantages of an attribute certificate is that it can be used for various purposes. It may contain group membership, role, clearance, or any other form of authorization. Yet another essential feature is that the attribute certificate provides the means to transport authorization information to decentralized applications. This is specially relevant because through attribute certificates, authorization information becomes "mobile", which is highly convenient for new e-commerce applications.

Actually, the mobility feature of attributes has been used in applications since ITU-T 1997 recommendation. However, it has been used in a very inefficient way. That recommendation introduced an ill-defined concept of attribute certificate.

For this reason, most actual applications do not use specific attribute certificates to carry authorization information. On the other hand, attributes of entities are carried inside identity certificates. The certificate field used for this purpose is the *subjectDirectoryAttributes* extension. This field conveys any desired Directory attribute values for the subject of the certificate, and is defined as follows:

```
subjectDirectoryAttributes EXTENSION ::= {
    SYNTAX   AttributesSyntax
    IDENTIFIED BY id-ce-subjectDirectoryAttributes }
AttributesSyntax ::= SEQUENCE SIZE (1..MAX) OF Attribute
```

This solution does not make entity attributes independent from identity, which can cause problems. Firstly, this is not convenient in situations where the authority issuing the identity certificate is not the authority for assigning privileges. This occurs very frequently, as we will discuss later. Secondly, even in the situations where the authority is the same one, we must consider that life of identity certificates is relatively long when compared to frequency of change of user privileges. This means that every time privileges change it would be necessary to revoke the identity certificate, and it is widely known that certificate revocation is a costly process.

Moreover, many applications deal with authorization issues like delegation (conveyance of privilege from one entity that holds a privilege to another entity) or substitution (one user is temporarily substituted by another user, and this one holds the privileges of the first one for a certain period of time).Identity certificates do not support delegation or substitution.

The ITU-T 2000 recommendation provides the solution to these problems. Attribute certificates are conveniently described, including an extensibility mechanism and a set of specific extensions are handled and a new type of authority for the assignment of privileges is defined, the *Attribute Authority* (AA).

The recommendation defines a framework that provides a foundation upon which a Privilege Management Infrastructure is built to contain a multiplicity of AAs and final users. Revocation procedures are also considered by defining the concept of *Attribute Certificate Revocation Lists* (ACRLs) which are handled in the same way as for CRLs published by CAs.

The identity and attribute certificates of one user are bound as shown in figure 1. We can see that the field *holder* in the attribute certificate contains the serial number of the identity certificate. Although linked, both certificates are independently managed. The important meaning of this is that a PKI and PMI are separate infrastructures in the sense that either structure can work on their own, or to be more precise, they can be established and managed independently. Next section describes in more detail this possibility.

4 Mutual independence of the Infrastructures

The mutual independence of the two infrastructures is also valid when considering other ways to describe the holder of the attribute certificate. In spite of using

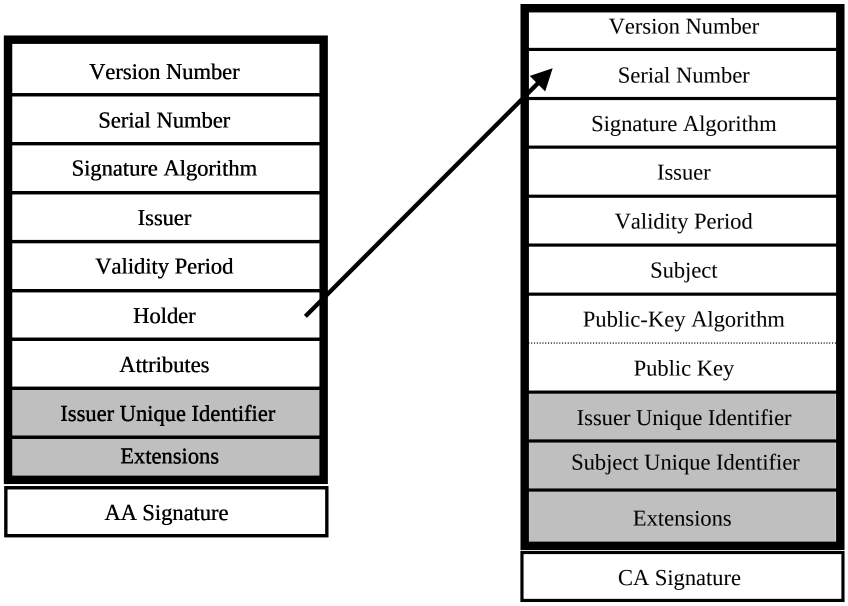


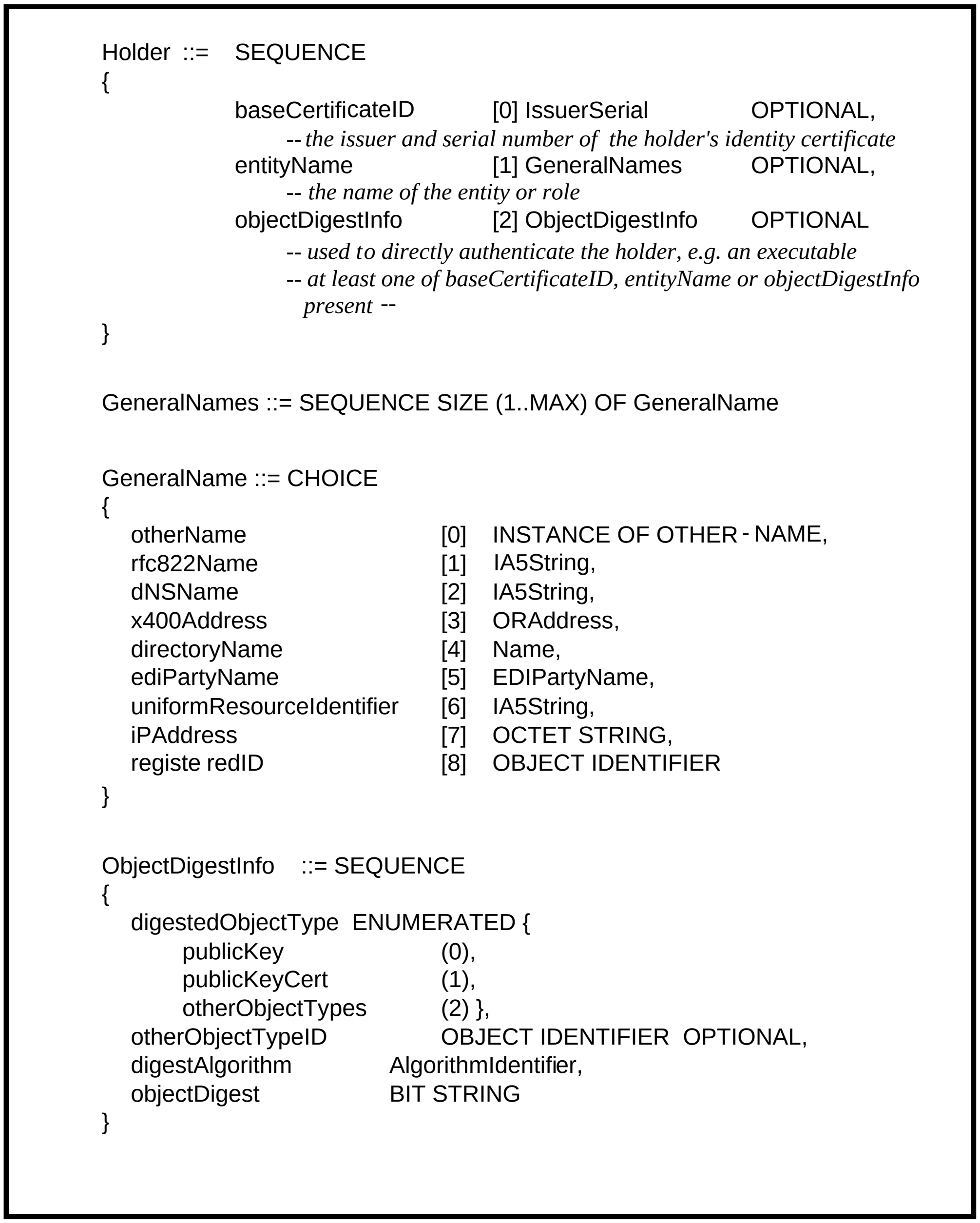
Fig. 1. Relation between identity and attribute certificates

the serial number for the identity certificate it is possible to bind the attribute certificate to any object by using the hash value of that object. For instance, the hash value of the public key, or the hash value of the identity certificate itself can be used. All possibilities for binding can be concluded from the ASN.1 [11] specification for the field *holder* shown in figure 2, where other related data structures are also specified. As we will see in next section, the content of this specification is essential for the scheme that we have developed.

The infrastructures are absolutely separated when considering the situation in which some other authentication method different from that one based on identity certificates is used. In these cases, a PKI is not even used, and the name of the subject is a good option to describe the holder of the attribute certificate.

The discussion about the separation of functions between PKIs and PMIs is a very relevant issue for this paper. From the point of view of the theory, the separation is possible as we have argued in previous paragraphs. From the point of view of real application scenarios separation is not only possible but, in our opinion, very convenient. We previously argued that in most cases the authority issuing the identity certificate is not the authority for assigning privileges. That is, the entity having the role of Certification Authority is not the same one as that one having the role of Attribute Certificate.

The main reason for this argument is that the identity of a user has a global meaning in most certification schemes (although a few schemes do not support



```
Holder  ::=   SEQUENCE
{
              baseCertificateID          [0] IssuerSerial            OPTIONAL,
                     -- the issuer and serial number of  the holder's identity certificate
              entityName                 [1] GeneralNames            OPTIONAL,
                     -- the name of the entity or role
              objectDigestInfo           [2] ObjectDigestInfo        OPTIONAL
                     -- used to directly authenticate the holder, e.g. an executable
                     -- at least one of baseCertificateID, entityName or objectDigestInfo
                        present --
}

GeneralNames ::= SEQUENCE SIZE (1..MAX) OF GeneralName


GeneralName ::= CHOICE
{
    otherName                        [0]   INSTANCE OF OTHER - NAME,
    rfc822Name                       [1]   IA5String,
    dNSName                          [2]   IA5String,
    x400Address                      [3]   ORAddress,
    directoryName                    [4]   Name,
    ediPartyName                     [5]   EDIPartyName,
    uniformResourceIdentifier        [6]   IA5String,
    iPAddress                        [7]   OCTET STRING,
    registe redID                    [8]   OBJECT IDENTIFIER
}

ObjectDigestInfo    ::= SEQUENCE
{
    digestedObjectType  ENUMERATED {
           publicKey                 (0),
           publicKeyCert             (1),
           otherObjectTypes          (2) },
    otherObjectTypeID                OBJECT IDENTIFIER  OPTIONAL,
    digestAlgorithm              AlgorithmIdentifier,
    objectDigest                 BIT STRING
}
```

Fig. 2. ASN.1 specification of Holder and related data structures

this idea, for instance, SPKI [4]). Thus, the CA does not necessarily belong to the organization where the user belongs to. The identity certificate can be issued by a CA managed by a governmental organization, a public organization, or even by a private company specialized in providing services and facilities related to certification of identities.

On the contrary, we believe that a user attribute has non-global meaning. An attribute certificate contains some kind of user authorization, and an authorization is merely valid for an specific application, scenario or environment. Hence, it can rarely be considered to have a global meaning. In fact, it is reasonable to think that the same user will have several attribute certificates, for different applications, scenarios or environments, while using only one identity certificate for all cases. Moreover, because of the restricted scope of an attribute certificate, it is convenient that this certificate is issued by an authority that is more or less local to the scope of the user.

This argument is even more valid if user attributes are considered as confidential information. The certificate may contain some sensitive information and then attribute encryption may be needed, as proposed by PKIX [5]. That kind of confidential information should be solely managed by people belonging to the organization.These reasons, but also the fact that user privileges can change frequently, suggest that, in order to preserve some level of efficiency, authorities external to the user organization are not the most appropriate to issue attribute certificates.

Non-globality of attribute certificates is not in contradiction with the ”mobility” feature argued at the beginning of this section. As we stated, this type of certificates are extremely useful for new distributed Internet applications because they facilitate that user authorization is not limited to a local computer system or to local resources. On the contrary, the mobility feature allows that authorization comprises a set of computer systems and resources geographically distributed over the network. An application running in this way, although distributed, can not be considered global.

Once we have conveniently argued that many applications need that the PKI and the PMI are established and managed independently, we consider in next subsection the idea of outsourcing the PKI while managing the PMI inside the organization.

4.1 Outsourcing the PKI

For many years big organizations have designed, deployed and managed their own security solutions. However, many of those organizations are not considering this model anymore. They have realized that they do not have the skills to evaluate the multitude of security vendor products, deploy, integrate and manage these products into their existing network infrastructure. Also, difficulties with recruiting highly skilled, costly security specialists add to the list of problems for most organizations. This is specially true for security solutions that include firewalls, virtual private networks, URL filtering and, certainly, identity certificates [12].

Therefore, many organizations try to remain focused on their own business and outsource as many security services and technologies as possible. Outsourcing is done to those security companies, typically known as *Managed Security Services Providers*, that have the resources to continuously update security-related products.

As for authentication and PKI, managed security service provide customers with third-party infrastructure to guarantee the authenticity of their clients, devices and content for a variety of applications, including remote access, IPSEC, server applications, work-flow messaging systems and e-commerce solutions. As PKI attracts a growing number of companies and organizations, the case of outsourcing PKI becomes favourable.

The principal argument against outsourced PKI is loss of control at the customer site. Therefore, outsourced PKI products are transferring control almost entirely to the customer. However, and even with this transference of control, the argument is entirely valid for PMIs. In our opinion, a PMI and their related services should not be outsourced because in many cases attribute certificates contain information that is of special relevance for the company. That information describes the privileges of their employees, and in some cases it is sensitive enough as to be totally or partially encrypted. Some clear examples are standard attribute types, like "access identity", "group", "role", "clearance", "audit identity", "administrators group", etc., that may put in high risk information considered as confidential inside the organization. When encryption of attributes is involved, the *Cryptographic Message Syntax* is used to carry the ciphertext and associated per-recipient keying information [7].

Therefore, outsourcing the PKI but not the PMI becomes a real working environment for many organizations. Of course, this scenario does not give additional difficulties to the organization. It is quite clear that when both infrastructures exist then identity certificates must be generated in the first place.

Afterwards, the organization will create attribute certificates binding each of them to the corresponding identity certificate issued by the outsourced PKI. As previously stated, binding can be done by using serial numbers of identity certificates or, whatever alternative information, as was shown in figure 2. It is important to point out this last idea, as our solution makes use of one of those alternatives, as we will explain in next section.

5 A new scheme of PMI

A new scheme of PMI has been designed considering some basic goals which we summarize as: (a) use of a distributed architecture of authorities that satisfy the needs of intra-company departmental certification; (b) avoid scalability problems associated to both extranet or company expansion; and (c) eliminate problems associated with the revocation procedures, specially those introduced by the use of ACRLs.

Regarding the distributed architecture of authorities, our scheme is based on the fact that the typical structure of many companies is hierarchical. In fact, companies tend to have their own structure of divisions, departments, sections, etc., as the example in figure 3 shows.

We also take into consideration that, in most of cases, it is desirable that the distributed authorization infrastructure mimics or fits the company structure. Therefore, we propose a scheme with various managers acting as Attribute Authorities and operating independently over different *domain of users* (group of employees). The location of those authorities matches with the nodes of the hierarchy in the organization, that is, each node corresponds to a division, department, etc. This facilitates that every Attribute Authority issues attribute certificates for those employees over which it has a direct control.

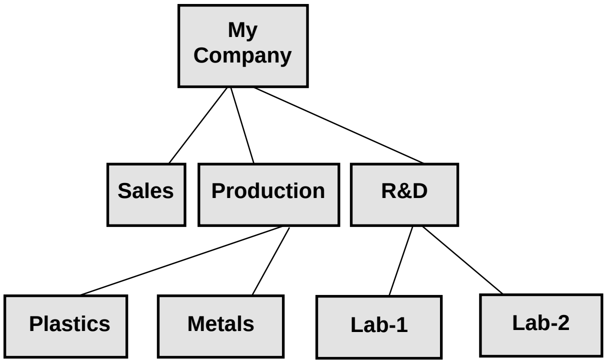


Fig. 3. A typical hierarchical company infrastructure

The main elements in our PMI design are the *Attribute Certificate Service Units* (ACSUs), which integrate attributes certification and management functions. Figure 4 shows all the components of a ACSU, which core is an Attribute Authority. More precisely, every ACSU is managed by an Attribute Authority who may be the manager in the division, department, etc.

Additionally, the ACSU contains a database to store the attribute certificates of the users local to the ACSU's domain. Each user's certificate is stored exclusively in the database of his/her ACSU, and that database is solely managed by the corresponding authority. Therefore, updating and revocation of certificates are local operations that do not affect the rest of the system. Revocation is possible, although ACRLs are not used in the system. When an attribute certificate is revoked because the user privileges change, it is deleted from a database and a new certificate is issued by the authority and stored.

The third component of the ACSU is the *Attribute Server*. Whenever a user (or a resource, an application, etc.) needs to know the privileges of certain user A in a domain X, the Attribute Server of that domain delivers the attribute certificate requested. A more detailed description of the request and deliver procedures are explained in the next Subsection.

5.1 System operation

The operation of the system is related to the very natural way of identifying users inside the organization. For instance, according to figure 3, the employee *Alice* may belong to the department of *Metals*, which is included in the division of *Production*. In the operation of our system, this employee is identified as *Alice@metals.production.mycompany* inside the organization. This is clearly neither considered nor used as an e-mail address by our system, although we use the same format [3] because it allows to link the attribute certificate to the identity certificate. Occasionally, it may happen that in some organizations the structure of divisions, departments, etc. may coincide with the hierarchy of Internet domains, but this does not have an effect in the system.

Attribute certificates issued by authorities in our new design follow the format established by ITU and, at the same time, extend the composition of *Holder* field. Of course, this extension is not a particular one of our creation. In contrast, it strictly follows one of the alternatives under the specification of the standard as shown in figure 2.

To be more precise, in our implementation the field *Holder* of every attribute certificate generated inside the organization contains two concatenated data objects. Types of those data objects are *IssuerSerial* and *rfc822*, respectively; that is, the result of the sequence *baseCertificateID* and *entityName*. Such a sequence is totally valid according to the mentioned specification.

The first data object, an identity certificate serial number, binds the attribute certificate issued internally with the identity certificate issued by an outsourced company or organization. The second data object, which is a user identification similar to the one of the example (*Alice@metals.production.mycompany*), is used for the search of *Alice*'s privileges following the procedure that is explained next for a general scenario.

The scheme defines a special user called AA@<domain> ($AA@x.y.z$ in the example shown in Figure 5), that denotes the corresponding AA in every ACSU. The certificate of any AA is stored in the database of its parent ACSU ($y.z$), except for the top-level domain ($.z$), that is the source of authorization.

The attribute certificate request process will be started by *Bob* as soon as he receives a request from *Alice*. This request has the following simplified information structure: *[Alice@x.y.z, operation]*S_{Alice}. The meaning of such structure is that *Alice* requests to *Bob* the permission to perform the operation, and digitally signs his request in order to avoid impersonation. At this moment, *Bob* needs *Alice*'s identity certificate to verify the request, and *Alice*'s attribute certificate to check if she is allowed to perform the operation.

Then, *Bob* firstly initiates the procedure to obtain *Alice*'s attribute certificate, that has been generated inside the organization. Figure 5 shows the procedure to obtain and verify it. We can see that *Bob* requests *Alice*'s certificate from his own ACSU (step 1) and this one directs the request to the ACSU located

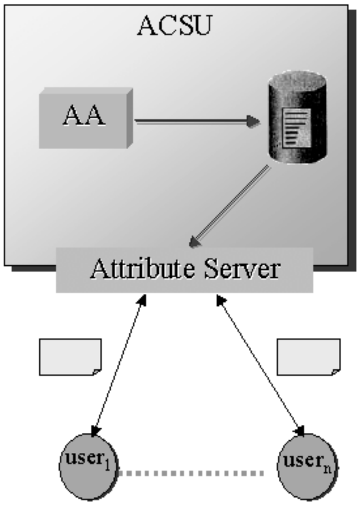


Fig. 4. Components of the ACSU

at the *x.y.z* node (step 2). The response from the addressee's ACSU (step 3) is then forwarded to *Bob* (step 4).

Afterwards, and in the case *Bob* needs to be more confident in the certificate he has received, he can request the certificate of *AA@x.y.z* from the ACSU located at *y.z*, obtaining a new certificate (Figure 5b). This procedure guarantees *Bob* that *Alice*'s authority has not been impersonated. This is a process of verifying the chain of attribute certificates up to the source of authority (at the top of the organization), but *Bob* can decide when he wants to stop going up.

According to the figure *Bob* must request the attribute certificate from his ACSU. This is so because access restrictions to ACSUs are set in the system, in such a way that a user can not access other ACSUs but the one in the domain where he is included.

It some circumstances it could happen that no ACSU is present at a certain node, say *y.z*. If this is the case, the certificate of *AA@x.y.z* would be automatically requested from the parent node, that is, *z*. This allows for a company to use an incomplete structure without loss of functionality.

6 Conclusions

By using an authentication service you can prove who you are, but it is clear that this not enough for many of the applications in e-commerce scenarios. Additionally, it is necessary to prove what you are allowed to do or, in other words, it is necessary to use an authorization service. Although authorization is not a

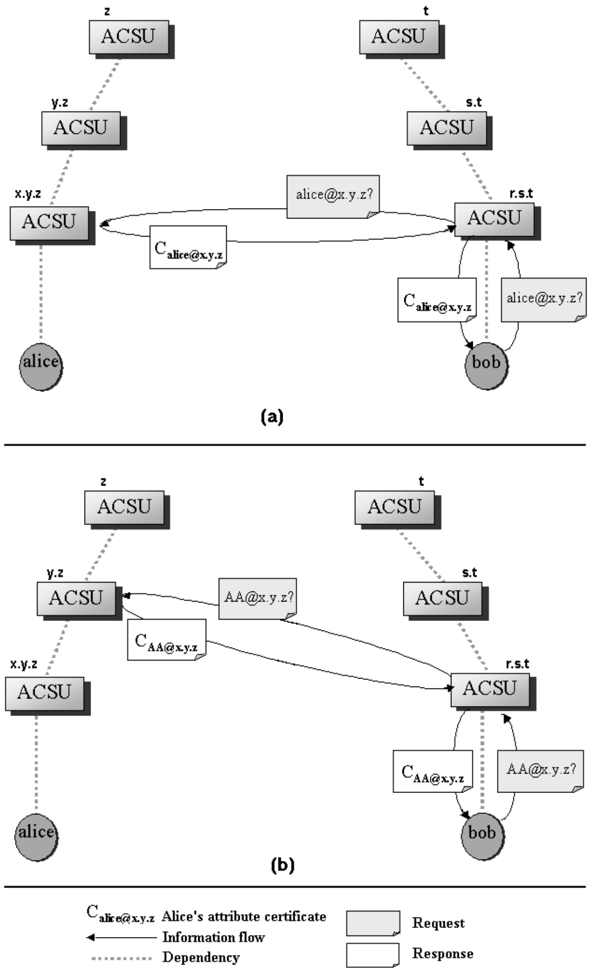


Fig. 5. a)Certificate Request b)Certificate Verification

new problem, traditional solutions have been used for central applications, and not distributed applications. Therefore, they are not valid for the scenarios we are considering.

ITU-T has created the concept of attribute certificate in order to solve these problems. The attribute certificates framework provides a foundation upon which a Privilege Management Infrastructure (with a multitude of Attribute Authorities) can be built. In fact, the ideas followed by ITU-T when designing PMIs are very likely to that one used to create PKIs. Both type of infrastructures are similar from the functional point of view, and they are linked, as the identity certificate and the attribute certificate of one user have fields with the same content.

Although linked, both types of infrastructures can operate independently. In the case of PKI services, these can be outsourced, but the case with PMI services is not the same, because much of the information contained in attribute certificates may be confidential. Therefore attribute certificates must be carefully managed and, preferably, only inside the company.

In this paper we have presented a new design of PMI that is specially suited for those companies that outsource PKI services but still need to manage the PMI internally. The new scheme has been designed considering some basic goals: (a) use of a distributed architecture of authorities that satisfy the needs of attribute certification that most companies have; (b) avoid scalability problems associated to both extranet or company expansion; and (c) eliminate problems associated with the revocation procedures, specially those introduced by the use of ACRLs. Therefore, although revocation procedures are allowed, there is no need for using ACRLs.

References

1. C. Adams, S. Lloyd, "Understanding Public-Key Infrastructure: Concepts, Standards and Deployment Considerations", New Riders, 1999
2. D. Chadwick, "An X.509 Role-based Privilege Management Infrastructure", *Business Briefing: Global Infosecurity*, 2002
3. D. Crocker, "Standard for the format of Arpa Internet Text Messages", Request for Comments 822, August 1982
4. C. Ellison et al. "SPKI Certificate Theory", Request for Comments 2693, IETF SPKI Working Group, September 1999
5. S. Farrell, R. Housley, "An Internet Attribute Certificate Profile for Authorization", Request for Comments 3281, IETF PKIX Working Group, April 2002
6. D. Ferraiolo, R. Jun, "Role-based access control", *Proc. 15th NIST-NCSC National Computer Security Conference*, 1992, pp. 554-563
7. R. Housley, "Cryptographic Message Syntax", Request for Comments 2630, IETF PKIX Working Group, June 1999
8. J. Hwang, K. Wu, D. Liu, "Access Control with Role Attribute Certificates", *Computer Standards and Interfaces*, vol. 22, March 2000, pp. 43-53
9. ITU-T Recommendation X.509, "Information Technology - Open systems interconnection - The Directory: Authentication Framework", June 1997

10. ITU-T Recommendation X.509, "Information Technology - Open systems interconnection - The Directory: Public-key and attribute certificate frameworks", March 2000
11. B. Kaliski "A Layman's Guide to a Subset of ASN.1, BER, and DER", November 1993
12. M. Lira, "Outsourcing your security to a Global Provider", *Business Briefing: Global Infosecurity*, 2002
13. A. Nash, W. Duane, C. Joseph, D. Brink, "PKI: Implementing and Managing E-Security", McGraw-Hill, 2001
14. R. Oppliger, G. Pernul, and Ch. Strauss. "Using Attribute Certificates to Implement Role-based Authorization and Access Control", *Proceedings of the 4. Fachtagung Sicherheit in Informationssystemen (SIS 2000)*, Zürich, October 2000, pp. 169-184
15. R.S. Sandhu, E.J. Coyne, H. Feinstein, C.E. Youman, "Role-based access control models", *IEEE Computer* Vol. 29, No. 2, 1996, pp.38-47