Using Attribute Certificates to Implement Role-based Authorization and Access Controls

Rolf Oppliger [1)], Günther Pernul [2)], Christine Strauss [3)]

1) eSECURITY Technologies Rolf Oppliger,
Thunstrasse 57b, CH-3074 Muri, Switzerland
rolf.oppliger@esecurity.ch

2) Department of Information Systems, University of Essen
Universitätsstrasse 9, D-45141 Essen, Germany
Phone +49 (0)201 183 4041, Fax. +49 (0)201 183 4011
pernul@wi-inf.uni-essen.de

3) Department of Management Science, University of Vienna
Bruenner Strasse 72, A-1210 Vienna, Austria
Phone +43 (0)1 4277 38112, Fax. +43 (0)1 4277 38094
christine.strauss@univie.ac.at

1 Abstract

Users of electronic commerce applications often face the problem of how to judge the value of a document that is digitally signed by someone claiming to be an authorized agent of a particular organization, such as a company or a federal office. While the claimant might provide a personal certificate that can be used for authentication, the more general questions are related to the issue of authorization: how can a user be certain that the agent is truly authorized to act on behalf of the organization and that the agent is acting in a legally-binding manner? Similarly, how can the organization be held liable for the digital signatures its authorized agents provide? This paper elaborates on possible means of addressing these and similar questions. In particular, it addresses the utilization of attribute certificates for implementing role-based authorization and access controls. In addition, the paper also elaborates on a possible implementation for commercial registers that could be used to certify the attribute authorities that issue attribute certificates.

2 Introduction

The use of public key cryptography for encryption and digital signatures requires the existence of a public key infrastructure (PKI). A PKI typically consists of one or several certification authorities (CAs) that issue and revoke certificates for users or other CAs [FoBa97]. Since the term “certificate” was first introduced by Loren M. Kohnfelder to refer to a digitally-signed record that consists of a name and a public key, binding a public key to a globally unique name has been assumed to be the primary purpose of a certificate [Kohn78]. In fact, this assumption explains why several PKIs have been designed on the basis of existing naming schemes and corresponding directory services, such as provided by the ITU-T recommendations of the X.500 series.

Due mainly to the singular use of the term “certificate”, there is still considerable confusion on how a PKI should actually be established. For example, the Internet Engineering Task Force (IETF) has tasked a Public Key Infrastructure X.509 (PKIX) Working Group (WG) to profile and establish a PKI for the Internet community based on the ITU-T recommendation X.509 [ITU88], whereas the Simple Public Key Infrastructure (SPKI) WG has been tasked with designing and producing a certificate structure and operating procedure that meets the needs of the same community for trust management in as easy, simple, and extensible way as possible. The IETF’s motivation in tasking two WGs is explained in part by fears that the challenges faced in building an X.509-based PKI for the Internet community might be too vast. Note that the Privacy Enhanced Mail (PEM) WG failed several years ago to profile and deploy an X.509-based PKI for secure electronic messaging [Kent93]. Taking a closer look at the two approaches being followed by the IETF WGs mentioned above (namely, the IETF PKIX and SPKI WGs), one recognizes that the main difference between them lies in the fact that the IETF PKIX WG assumes the existence of a global name space, while the IETF SPKI WG does not make this assumption and instead uses linked local name spaces such as the ones proposed by Ron Rivest and Butler Lampson in their Simple Distributed Security Infrastructure (SDSI). See `http://theory.lcs.mit.edu/~cis/sdsi.html` for a comprehensive overview about the SDSI project. More recently, the proponents of the SDSI and the participants of the IETF SPKI WG have joined their efforts. The resulting certificates are often referred to as SDSI/SPKI certificates.

One may reasonably call into question whether the Internet community and the electronic commerce applications that are supposed to

be deployed on the Internet actually require the infrastructure provided by a X.509-based PKI. Besides, the situation is fundamentally different and simpler from a corporate point of view. Most corporate environments implement a name space in which each employee is not only identified with a unique name, but is often also assigned a unique employee number. As a consequence, the corporate environment does host a name space that can be used to have certificates bind public keys to unique names: uniqueness is guaranteed within the corporate environment. Moreover, electronic commerce applications can make use of attribute certificates that bind specific attributes to individuals acting in various specific roles.

The users of electronic commerce applications are often faced with the challenge of assessing the value of a document bearing the digital signature of someone who claims to be an authorized agent for a particular organization, such as a corporation or a federal agency. While the claimant may well provide a personal public key certificate that can be used for authentication, the more general questions concern the issue of authorization: how can the user be certain that the agent is actually authorized to act on behalf of the organization and that these actions are legally binding? Similarly, how can the organization be held liable for the digital signatures provided by its authorized agents? According to a position paper presented by Joan Feigenbaum at the Third USENIX Workshop on Electronic Commerce in 1998, a PKI that enables applications to decide who signed a request does not provide immediate utility; what is needed is an infrastructure that allows the verifier of a digitally-signed document to determine whether the signatory actually has the authority to carry out his intentions [Feig98]. According to this line of reasoning, a PKI should not be used primarily to enable authentication, but rather, to enable authorization.

Recently, the singular use of the term “certificate” has been challenged by the use and proliferation of SDSI/SPKI and attribute certificates within the Internet community. In a more general sense, the term “certificate” refers to a digitally-signed testimony addressed to “to whom it may concern” and stating some fact or granting some form of privilege. A certificate can bind a public key to a (globally unique) name. However, this is just one of several possibilities that a certificate has for stating a fact, and there are other facts that a certificate may state as well. For example, a certificate may grant some attributes to its owner; this is actually the aim of an attribute certificate (AC). Obviously, ACs are well suited for controlling access to system resources and for implementing role-based authorization and access controls accordingly. In this function, ACs are closely related to the

privilege attribute certificates (PACs) that are being used in the European SESAME (Secure European System for Applications in a Multivendor Environment) project [Oppl96] and have been incorporated into the Open Group's Distributed Computing Environment (DCE), as well as in Microsoft's Windows 2000 operating system in slightly modified forms.

Following this line of argumentation, this paper elaborates on various possibilities for addressing the authorization problem. In particular, it addresses the use of attribute certificates to implement role-based authorization and access controls. In addition, the paper briefly outlines a possible implementation for the commercial registers in Switzerland. The remainder of this paper is organized as follows: Role-based access controls and attribute certificates are introduced in Sections 2 and 3. Section 4 presents a possible model that shows how attribute certificates can be used to implement role-based authorization and corresponding access controls, while the above-mentioned implementation is briefly outlined in Section 5. Finally, the paper's conclusions are presented in Section 6.

3 Role-based Authorization and Access Controls

Access control is concerned with limiting the activities of a legitimate user within a system. It is enforced by a reference monitor which mediates every attempt by consulting an authorization base to determine if the user attempting to perform an activity is actually authorized to perform that activity. Usually, access control relies on and coexists with other security services in a system. However, it is important to make a clear distinction between authentication and access control: correctly establishing the identity of the user is the responsibility of the authentication service. Access control assumes that authentication of the user has been successfully verified prior to enforcement of access control. This is equally true for stand-alone systems, as well as for networked and distributed systems. In either case, an access control must be coupled with auditing that guarantees posteriori analysis of all the activities undertaken by users in the system. Although auditing might be essential for holding users accountable for their actions, it might also stand in conflict with privacy considerations at the same time.

Two different policies for authorization and access control are commonly performed. *Discretionary policies* govern the access of users to information on the basis of the users' identities and authorizations. Authorizations specify (for each individual user and each object in the system) the access modes that the user is allowed to perform on the

object. Each activity is checked against the authorization base. If an authorization stating the user can access the object in the specified mode exists, then access is granted - otherwise it is denied. Discretionary policies have a drawback in that they do not provide the control of information flow between users, which is the main focus of mandatory access control policies. *Mandatory policies* govern access on the basis of the classification of objects and users according to security levels: access to an object is granted if the security level of a particular user stands in accordance with the security level of the object.

Role-based policies have recently received increasing attention in the security community as an alternative to traditional discretionary and mandatory access control models [SCFY96; CMS97; FeKu92]. 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 (fulfill the responsibilities, duties and obligations of a person acting in a certain role). 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 assigned according to roles. Users are given authorizations to act in certain roles, with any individual user being capable of executing those activities for which his role is authorized. In general, a user can take on different roles on different occasions.

There are a number of characteristics of role-based models that make them suitable for our purposes:

- Authorizations are specified with reference to object classes rather than to single objects. For example, the fact that a department manager can sign documents of any type can be represented by an authorization for the role `Department Manager` on security object `document`, without needing to specify authorizations for each manager on each document. This characteristic makes it possible to develop acting patterns in terms of the access provided to object classes and thus avoids the need to look into individual objects.
- Authorizations to access objects are specified on the basis of activities users have to perform in order to fulfill their duties and to achieve their goals. A role is defined as a set of actions and responsibilities associated with a particular working activity. Requests by a user are then granted with respect to the role the user is currently acting in. As an example, consider users acting in

such roles as `clerk`, `teller`, or `branch manager`, with each role implying different obligations and duties. This characteristic makes it possible to compare roles for different systems and to analyze their similarities.

- Authorizations to access information can be specified for general transactions, thus avoiding elementary read and write operations. Transactions are procedures defined on object types and include more elementary operations, as well as calls to other procedures. For example, typical bank transactions such as `withdraw` or `deposit` can be defined for class `account` or `add-interest` and `sign-contract` for class `loan`. This capability allows one to evaluate the similarity between operations in a manner that can also be semantically rich.

- A natural hierarchy of roles exists in many applications and is often based on the common principles of generalization and specialization. Users assigned to specialization roles will inherit all authorizations assigned to more generic roles (super roles). Thus, hierarchical role structures can help to simplify authorization management.

- Roles allow a user to sign on at the lowest access level necessary to perform a specific activity. Users capable of signing on to powerful roles do not need to play these roles until the powerful privileges associated with them are actually required. This mechanism reduces the probability of successful intrusions performed by "masquerading" and thus limits potential security infractions.

- Separation of duty can be performed by specifying conditions on roles. For example, certain activities might require different users to *cooperate* in different roles by performing a task together, (i.e. the "Four-eyes-principle"), but might also exclude such cooperation by strictly *separating* certain activities (e.g., the granting of a credit and the booking of the order of payment must be carried out by two distinct users).

Thus, role-based models for attribute certificates have the advantage of establishing a set of structural, lexical, and transactional profiles. Such profiles may refer to similar roles in different organizations, but can imply the same set of authorizations, duties and obligations for users acting in these roles. Moreover, the task of administrating authorizations (i.e., specifying which transactions each role may exe-

cute on each object class) remains with one or more privileged users, such as an in-house authority or a national commercial register (the second example is further explored in Section 5, when we elaborate on the ZEFIX application).

4 Attribute Certificates

As already mentioned in the introduction, the binding association between the identity of the owner of a public key and that key must be documented in order to prove the ownership of the public key. This binding is usually called a “public key certificate” or “certificate” for short. Public key certificates are generated, distributed, and potentially revoked by CAs. In general, a X.509v3 public key certificate can also convey authorization information about its owner. This information can be encoded in one of the X.509v3 standard or extension fields. Note, however, that there are at least two important reasons why caution should be taken in using X.509v3 public key certificates for conveying authorization information [FoBa97]:

- First, the authority that is most appropriate for verifying the identity of a person associated with a public key may not be appropriate for certifying the corresponding authorization information. For example, in a company the corporate security or personnel department may be the appropriate authorities for verifying the identities of persons holding public keys, whereas the department of finance may be the appropriate authority for certifying permissions to sign on behalf of the company.
- Secondly, the two types of certificates may follow different dynamics. For example, the persons authorized to perform a particular function in a company may vary monthly, weekly, or even daily. Contrary to that, public key certificates are typically designed to be valid for a much longer period of time (e.g., one or two years). If it becomes necessary to revoke and reissue public key certificates frequently because of changing authorizations (that are encoded into the public key certificates), this may have a severe impact on the performance characteristics of the resulting certificate management system.

Recognizing that X.509 public key certificates are not always the best vehicle for carrying authorization information, the U.S. American National Standards Institute (ANSI) X9 committee developed an alternative approach known as attribute certificates (ACs). Similar to public

key certificates, ACs bind the characteristics of an entity (called attributes) to that entity through the signature of a so-called Attribute Authority (AA) on a particular AC. Consequently, the major difference between a public key certificate and an attribute certificate is that the former includes a public key (with the key being certified), whereas the latter includes an attribute (with the attribute being certified). As such, an AC can be used for various purposes: for example, an AC may contain group membership, role, clearance, or any other form of authorization or access control-related information associated with the AC owner. In conjunction with authentication services, ACs may also provide the means to securely transport authorization information to decentralized applications. Consequently, ACs are particularly well suited for controlling access to system resources and implementing role-based authorization and access controls accordingly. Meanwhile, the use of ACs has also been incorporated into both the ANSI X9.57 standard and the X.509-related standards and recommendations of both ITU-T and ISO/IEC. More recently, the IETF Transport Layer Security (TLS) and Public Key Infrastructure X.509 (PKIX) WGs have also started to work on AC-based authorization as a possible extension to cryptographic security protocols, such as the TLS protocol (the TLS protocol is a slightly enhanced version of Netscape's Secure Sockets Layer (SSL) protocol) [Farr00].

Anybody can define and register attribute types and use them in attribute certificates. The certificate is digitally signed and issued by an AA. AAs, in turn, are assumed to be certified by CAs, such that a single point of trust - namely a trusted public key of a root CA - can be used to validate the certificates of AAs, other CAs, and other end users. Apart from differences in content, an attribute certificate is managed the same way as a public key certificate. For example, if an organization already runs a directory service to distribute public key certificates and certificate revocation lists (CRLs), this service can also be used to distribute attribute certificates. Note that - similar to public key certificates - ACs can be used in either the “push” or “pull” model:

- In the “push” model, the ACs are pushed from the client to the server.
- In the “pull” model, the server pulls the ACs from an online network service (either the attribute certificate issuer or a directory service that is fed by the attribute certificate issuer).

An attribute certificate infrastructure should support both models since some applications work best when a client pushes the AC to the

server, whereas for other applications it is more convenient for the client simply to authenticate to the server and for the server to request the client's AC from a corresponding network service or attribute certificate repository. Note that this is somehow contradictory to Proposition 2 of [Rive98], where it is claimed that “the signer can (and should) supply all evidence the acceptor needs, including recency information.” While this proposition holds in most situations, there are also some situations that require a server to handle specific tasks on the client's behalf (e.g., thin clients or, more generally, devices with small computing power).

According to the specifications of the ANSI X9 committee, an attribute certificate may consist of the following fields [FoBa97]:

- *Version:* This field indicates the version of the AC format in use (currently version 1).
- *Subject:* This field identifies the principal with which the attributes are being associated. Identification can be either by name or by reference to an X.509 public key certificate. Such a reference comprises a combination of an X.509 issuer name and a corresponding certificate serial number.
- *Issuer:* This field identifies the AA that issued the AC.
- *Signature:* This field indicates the digital signature algorithm used to sign the AC.
- *Serial Number:* This field contains a unique serial number for the AC. The number is assigned by the issuing AA and used in a CRL to identify the attribute certificate.
- *Validity:* This field may contain a set of possibly overlapping time periods during which the AC is assumed to be valid.
- *Attributes:* This field contains information concerning the owner of the AC (the owner is the principal that is referred to in the subject field). The information may be supplied by the subject, the AA, or a third party, depending on the particular attribute type in use.
- *Issuer Unique Identifier:* This field contains an optional bit string used to make the issuing AA name unambiguous in the case that the same name was reassigned to different principals through time.

- *Extensions:* This field allows for the addition of new fields to the AC. It basically works the same way as the extensions field of an X.509 public key certificate.

Note that ACs constitute a general-purpose mechanism that potentially has many uses and that distribution of authorization information is just one use. Also, note that the above-mentioned format for an AC is just one proposal (that of the ANSI X9 committee) and that other competing formats have been and will probably continue to be proposed and submitted for standardization. For example, the World Wide Web Consortium (W3C) Digital Signature Working Group ("DSig") has proposed a standard format for making digitally-signed, machine-readable assertions about a particular information resource. In a first attempt to apply the standard format, the Working Group has focused on digital signatures for PICS labels. However, this is just one possible application, and there are many other applications that one can think of. More generally, it is the goal of the DSig project to provide a mechanism to make the statement:

signer believes *statement* about *information resource*

Obviously, attribute certificates also represent information resources and can be digitally signed according to the DSig syntax and semantics. Refer to the corresponding Web pages hosted at `http://www.w3.org` for further information about the DSig project. In this paper, we do not attempt to address all possible formats of ACs, but instead focus on their functionalities. From this point of view, it does not really matter whether ACs are implemented according to the formats proposed by the ANSI, ITU-T, ISO/IEC, IETF, or W3C. Regardless of which format is implemented, ACs represent an important technology for authorization in electronic commerce applications.

5 Using Attribute Certificates

In this section, it is assumed that a non-empty set TCA = {CA_1, CA_2, ... } of CAs that are commonly considered to be trustworthy exists. The trust associated with these CAs may be based on a general accreditation or certification scheme for CAs. For example, a state or country may publish criteria against which CAs must be evaluated and certified in order to be commonly considered as trustworthy. This is somewhat similar to the use of the Trusted Computer Security Evaluation Criteria (TCSEC) in the U.S., the Information Technology Security Evaluation Criteria (ITSEC) in Europe, or the Common Criteria (CC) on a global scale. The assumption that TCA is not empty is

essential for this paper; recent developments within the European Union have shown that this assumption may indeed be realistic [CEC97; CEC99].

Each member-CA of the TCA (and each CA that is commonly considered to be trustworthy) has its own public key published in authentic (and authenticated) form. These public keys are required to bootstrap trust related to public key certificates and ACs. As a result, the CAs of the TCA collectively represent a public key infrastructure (PKI) or certificate infrastructure for the state or country under consideration.

In addition to the TCA (and the corresponding PKI), it is further assumed that each organization O has at least one Attribute Authority AA(O) that is registered with an appropriate national body, such as a chamber of commerce. In general, there may also coexist several Attribute Authorities $AA_1(O)$, ..., $AA_i(O)$ for organization O. Again, each AA(O) holds a public key pair of which the public key is certified (and digitally signed) by a CA_i ($i > 0$) that is a member of the TCA. Usually, the AA(O) issues and revokes ACs for the authorized agents of O. The ACs, in turn, certify the bindings between a name (that is unique for the organization O) and a specific role within the corresponding organization. For example, if Mister X is an authorized agent for company Y, he will have a certificate for his public key and an AC for the role he plays within the company Y. The public key certificate is issued by a CA that is a member of the TCA, whereas the AC is issued by an AA(Y). Whenever X has to sign a document that must be legally binding in one way or another (such as a contract), he uses his private key to digitally sign the document and provides the intended recipient(s) with his public key certificate together with the AC certifying his role within the company Y. The recipient(s), in turn, use(s) X's public key certificate to verify the digital signature and apply the AC to actually verify that X enjoys the appropriate authorization within company Y.

6 Implementation

In Switzerland, the cantons maintain and are ultimately responsible for the commercial registers in their appropriate domains. Anyone can register a company with the commercial register of his canton. Moreover, anyone can request the current status of a specific company from the appropriate commercial register (that of the canton in which the company officially resides). In general, this status information is provided in paper form, is time-stamped, and is legally binding. In addition, the same information has also be made available online through a service named the "Central Business Names Index on

Internet"- or "Zentraler Firmenindex auf dem Internet" ("ZEFIX") in German. This service is publicly available and accessible at URL `http://zefix.admin.ch`. A slightly more advanced version of the service is also available for official use within the Intranet maintained by the Swiss federal administration (at URL `http://zefix.bj.admin.ch`); this service is for internal use only and is not available to the public.

The ZEFIX server maintains a database that includes the information provided by the cantons' commercial registers. For example, if Mister X meant to do business with Mister Y from company Z, he would request the legal status of Y (with regard to Z) by requesting the corresponding commercial register. If the commercial register acknowledged that Y is an authorized agent of Z, X would continue doing business with Y. However, if the commercial register did not acknowledge Y's authorization , X would not continue doing business with Y. In the paper world, the process of requesting the legal status of a person claiming to be an authorized agent takes some time (at least two postal deliveries); the ZEFIX server was established to shorten this period of time. It is now possible to request the same information that is available from the commercial registers in electronic form. The advantage is speed: the request can be answered with a database query in a relatively short period of time (several seconds). The disadvantage to this approach lies in the fact that the ZEFIX service cannot be made liable for any of the information that it provides. If the information provided by ZEFIX is to be used in some legally binding manner, it remains more prudent to request a paper extract from the commercial register; such an extract can be ordered online, but is delivered by means of the postal system.

Obviously, there are - at least - two possible means by which the ZEFIX service could be improved:

- First, it is possible to use SSL/TLS and HTTPS to secure the connection between the ZEFIX server and the requesting client (browser).
- Secondly, it is possible to use attribute certificates as outlined in the previous section of this paper.

This paper focuses on the second possibility. Note that the role of a commercial register is to make publicly available the information that is provided by those organizations that reside within its domain in some legally-binding way. An organization O is supposed to announce its authorized agents to the commercial register and the commercial register, in turn, is supposed to make this information

publicly available (mainly to the clients and trading partners of O). In essence, the commercial register does not provide new information; it simply notarizes information that is provided by the organization itself. As a consequence, it would be possible for the commercial register to publicly announce an entity within the organization, with the entity being capable of autonomously authorizing agents. These agent authorizations could then be made publicly available in repositories that would be managed by the commercial register(s).

The model introduced in the previous section can be used to implement this scheme. Assume that a commercial register is able to nominate an AA(O) for organization O, and to make publicly available a public key certificate for AA(O). Further, assume that AA(O) is able to issue and revoke ACs for the members of O. In this situation, if Mister X wanted to check the legal status of Mister Y (with regard to organization Z), he would verify the following things:

- The digital signature provided by Y;
- The corresponding public key certificate for Y;
- The attribute certificate of Y (issued by AA(Z));
- The nomination of AA(Z) by the appropriate commercial register.

If these issues were successfully verified, X could assume Y to be an authorized agent acting on behalf of Z in some legally-binding manner (depending on the legislation on electronic or digital signatures in the respective country). Obviously, this scheme also reduces the administrative overhead of the commercial registers, as most information is provided and maintained by the organizations themselves. Finally, it is important to note that the same mechanism can also be used from an organization's point of view to nominate trustees who file tax declarations on their behalves.

7 Conclusions

In electronic commerce applications, users are often faced with the challenge of establishing the value of a document that has been digitally signed by someone claiming to be an authorized agent of a particular organization, such as a corporation or government institution. This paper has elaborated on possibilities for addressing this problem. In particular, it has addressed the use of attribute certificates for implementing role-based authorization and access control models. In addition, it has also briefly outlined a possible implementation for the commercial registers in Switzerland (as part of the ZEFIX service).

There are several open issues related to the large-scale implementation of ACs according to one of the procedures mentioned earlier. For example, ACs and AC services must be standardized in some way or another; at the time of this writing, this has not yet come to be the case. Similarly, the use of ACs must be supported in custom client (and server) software. In particular, a client must be able to send the appropriate ACs together with public key certificates. Eventually, the HTTP and SSL/TLS protocol specifications (or the specifications of other cryptographic security protocols) must be adapted to make use of ACs. This work is currently under way within the IETF TLS WG. Finally, there are also some legal issues that must be addressed. Note that global markets and global commerce are not (yet) based on a uniform legal framework; consequently, there are currently no binding regulations for electronic global markets. Global executable law that regulates electronic commerce will - like in most other analogous technology-related fields, such as hacking and data protection - lag behind the actual technological and business developments. It might be left to the initiative of concerned groups having common interests to establish prototypical services that are based on common efforts and mutual benefits. The wide variability that is found even in the basics of conventional commercial law makes it particularly difficult to harmonize the distinct national versions. No practicable solution should be expected in the near future. For example, each individual state within the United States of America has its own Business Corporation Act and doing business in the US involves federal, state and local laws. Non-US companies dealing with US companies are well advised to get a certified copy of the board of directors' resolution authorizing the transaction that is to be performed. Similarly, it is not clear what (national or international) bodies register the AAs of the organizations that want to make use of ACs. The commercial registers addressed in this paper are just one possibility. In any event, prototype implementations similar to the one proposed in this paper will help to study and (hopefully) improve general understanding about the problems that surround the large-scale deployment of attribute certificates.

8 References

[CMS97] Castano, S., Martella, G., Samarati, P. (1997), Analysis, comparison and design of role-based security speicifications. Data and Knowledge Engineering 21, pp. 31-55.

[CEC97] Communication from the Commission of the European Communities, Ensuring Security and Trust in Electronic Communi-

cation - Towards a European Framework for Digital Signatures and Encryption, COM(97) 503 final, Brussels, 8.10.97.

[CEC99] Communication from the Commission of the European Communities, European Parliament Council Directive on a common framework for electronic signatures, Nov. 1999.

[Farr00] Farrell, S. An Internet AttributeCertificate Profile for Authorization, Internet Draft <draft-ietf-pkix-ac509prof-*.txt>, work in progress.

[Feig98] Feigenbaum, J. (1998), Towards an Infrastructure for Authorization, Position Paper presented at the 3rd USENIX Workshop on Electronic Commerce, August 31 - September 3, Boston, Massachusetts, MA (USA).

[FeKu92] Ferraiolo, D., Kuhn, R. (1992), Role-based access controls. Proc. 15th NIST-NCSC National Computer Security Conference, pp. 554-563.

[FoBa97] Ford, W., Baum, M.S. (1997), Secure Electronic Commerce: Building the Infrastructure for Digital Signatures & Encryption, Prentice Hall PTR, Upper Saddle River, NJ, (USA).

[ITU88] ITU-T (former CCITT) Recommendation X.509 (1988), The Directory - Authentication Framework.

[Kent93] Kent, S.T. (1993), Internet Privacy Enhanced Mail, Communications of the ACM, Vol. 36, No. 8, pp. 48-60.

[Kohn78] Kohnfelder, L.M. (1978), Towards a Practical Public-key Cryptosystem, MIT bachelor's thesis.

[Oppl96] Oppliger, R. (1996), Authentication Systems for Secure Networks, Artech House Publishers, Norwood, MA.

[Rive98] Rivest, R.L. (1998), Can We Eliminate Certificate Revocation Lists? Proceedings of Financial Cryptography.

[SCFY96] Sandhu, R.S., Coyne, E. J., Feinstein, H., Youman, C. E. (1996), Role-based access control models. IEEE Computer Vol. 29, No. 2, pp. 38-47.