PKI/PMI AND SMART TOKENS IN HEALTHCARE INFORMATION SYSTEMS

Hailong Liu, Wenhua Qi, Qishan Zhang, Jinpei Wu
School of Electronics and Information Engineering
Beihang University, P. R. China 100083

ABSTRACT

While healthcare industry is striving to achieve e-health systems for improvements in healthcare quality, cost, and access, privacy and security about medical records should be considered carefully. This paper makes a deep study of Public Key Infrastructures (PKIs) and Privilege Management Infrastructures (PMIs) and how they can secure e-health systems. To access resources, e.g. patient records, both authentication and authorization are needed, so public key certificates and attribute certificates are both required to protect healthcare information. From a typical medical scenario, we see not only static but also dynamic permissions are required. Dynamic authorization maybe the most complex problem in e-health systems.

KEYWORDS

security, PKI, PMI, public key certificate, attribute certificate, smart token, e-health, digital signature

INTRODUCTION

The Internet has become the most important, and potentially the most effective, communication medium the world has ever seen. More recently, the Internet has become a useful education and information tool for healthcare providers and healthcare consumers. E-health is defined as the application of Internet and other related technologies in the healthcare industry to improve the access, efficiency, effectiveness, and quality of clinical and business processes utilized by healthcare organizations, practitioners, patients, and consumers in an effort to improve the health status of patients[1].

While the e-health trend is going on, privacy and security of patient records should be considered carefully, for this is demanded by both ethic and legality. To protect patient records, security
requirements need to be defined, e.g. confidentiality, integrity, authenticity, and non-repudiation. These requirements can be met by means of cryptographic mechanisms as well as access control mechanisms. Substantially, the security depends on the authenticity of data like public keys as well as on the authenticity of special attributes of entities like roles and authorized rights. An authentic link between a public key and its owner can be provided by a public key certificate, while an authentic link between an entity and its attributes can be provided by a attribute certificate. So all the things involved in the management of these certificates can regarded as security infrastructures, that is PKIs/PMIs. PKIs manage public key certificates, and PMIs manage attribute certificates.

The rest of this paper is organized as follows. Section 2 outlines a common framework of PKIs and PMIs, especially a detail description about attribute certificates is given. Section 3 introduces the smart tokens and their usage. Section 4 describes how to use PKIs and PMIs and smart tokens in e-health systems, and a prototype is designed. Finally, in section 4 we conclude the paper and point out some future works.

**PKI/PMI FRAMEWORKS**

In order to control access to a resource, e.g. a patient record, both authentication and authorization are needed. Early version of the ITU-T X.509 standard\(^2\) have concentrated on standardizing strong authentication techniques, based on digital signatures, public key certificates, and Public Key Infrastructures (PKIs). The latest version of X.509\(^3\), published in 2002, is the first edition to standardize an authorization technique, and this is based on attribute certificates and Privilege Management Infrastructures (PMIs).

A PMI is to authorization what a PKI is to authentication. Consequently there are many similar concepts shared between PKIs and PMIs. These are summarized in Table 1.

<table>
<thead>
<tr>
<th>Concept</th>
<th>PKI entity</th>
<th>PMI entity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Certificate</td>
<td>Public Key Certificate (PKC)</td>
<td>Attribute Certificate (AC)</td>
</tr>
<tr>
<td>Certificate issuer</td>
<td>Certification Authority (CA)</td>
<td>Attribute Authority (AA)</td>
</tr>
<tr>
<td>Certificate user</td>
<td>Subject</td>
<td>Holder</td>
</tr>
<tr>
<td>Certificate binding</td>
<td>Subject’s Name to Public Key</td>
<td>Holder’s Name to Privilege Attribute(s)</td>
</tr>
<tr>
<td>Revocation</td>
<td>Certificate Revocation List (CRL)</td>
<td>Attribute Certificate Revocation List (ACRL)</td>
</tr>
<tr>
<td>Root of trust</td>
<td>Root Certification Authority</td>
<td>Source of Authority (SOA)</td>
</tr>
<tr>
<td>Subordinate authority</td>
<td>Subordinate Certification Authority</td>
<td>Attribute Authority (AA)</td>
</tr>
</tbody>
</table>

Table 1. A Comparison of PKIs and PMIs\(^9\)

A public key certificate (PKC) is used for authentication and maintains a strong binding between a
user’s name and his public key, whilst an attribute certificate (AC) is used for authorization and maintains strong binding between a user’s name and one or more privilege attributes. The entity that digitally signs a public key certificate is called a Certification Authority (CA), whilst the entity that digitally signs an attribute certificate is called an Attribute Authority (AA). The root of trust of a PKI is sometimes called the root CA whilst the root of trust of the PMI is called the Source of Authority (SOA). CAs may have subordinate CAs that they trust, and to which they delegate the powers of authentication and certification. Similarly, SOAs may delegate their powers of authorization to subordinate AAs. If a user needs to have his signing key revoked, a CA will issue a certificate revocation list (CRL). Similarly if a user needs to have his authorization permissions revoked, an AA will issue an attribute certificate revocation list (ACRL).

While PKCs have been widely used, ACs are relatively new concept, so we give a more detail discussion about ACs. Typically, a AC contains attributes about a entity, these attributes tell what permissions the holder have, so what a AC looks like is tightly restrict to the authorization scheme. Various authorization schemes have been devised in the past, and X.509 supports each of them.

Traditionally, the most popular and well-know model is the Discretionary Access Control (DAC) scheme. In the DAC scheme, users are optionally given access rights to resources by the resource administrator, and the access rights are typically held as access control lists within each target resource. In an X.509 PMI, the access rights are held within the privilege attributes of attribute certificates issued to users. Each privilege attribute within an AC will describe one or more of the user’s access rights. A target resource will then read a user’s AC to see if he is allowed to perform the action that he is requesting.

Another authorization scheme is the Mandatory Access Control (MAC) scheme, which is popular within military systems. A MAC system is often a Multilevel Secure (MLS) system. Every target is given a security label, which includes a classification, and every subject is given a clearance, which includes a classification list. The classification list specifies which type of classified target the subject is allow to access. A typical hierarchical classification scheme used by the military is: unmarked, unclassified, restricted, confidential, secret, and top secret. A typical security policy, designed to stop information leakage, is “no read up and no write down”. This specifies that a subject cannot read targets with a higher classification than his clearance, and cannot write to targets with a lower classification. X.509 supports MLS, by allowing subjects to be given a clearance AC. The privilege attribute in the AC now holds the users clearance. Targets can be securely configured with their own security label and the security policy that is to direct them.

More recently, research has focused on Role Based Access Controls (RBAC), and a standard for RBAC has been proposed[5]. In the core RBAC model, a number of roles are defined. These roles typically represent organizational roles such as secretary, manager, employee etc. In the authorization policy, each role is given a set of permissions i.e. the ability to perform certain actions on certain targets. Each user is then assigned to one or more roles. When accessing a target, a user presents his
role(s), and the target reads the policy to see if this role is allowed to perform this action. X.509 supports core RBAC by defining role specification attribute certificates that hold the permissions granted to each role, and role assignment attribute certificates that assign various roles to the users. In the former case, the AC holder is the role, and the privilege attributes are permissions granted to the role. In the latter case the AC holder is the user, and the privilege attributes are the roles assigned to the user.

The hierarchical RBAC model is a more sophisticated version of the core RBAC model. With this model, the roles are organized hierarchically, and the senior roles inherit the privileges of the more junior roles. So for example we might have the following hierarchy:

   employee > programmer > manager > director.

If a privilege is given to an employee role e.g. can enter main building, then each of the superior roles can also enter the main building even though their role specification does not explicitly state this. If a programmer is given permission to enter the computer building, then managers and directors would also inherit this permission. Hierarchical roles mean that role specifications are more compact. X.509 supports hierarchical RBAC by allowing both roles and privileges to be inserted as attributes in a role specification attribute certificate, so that the latter role inherits the privileges of the encapsulated roles.

Another extension to core RBAC is constrained RBAC. Constrained RBAC adds Separation of Duty relations to the RBAC model. Separation of duty relations are used to enforce conflict of interest policies that organizations may employ to prevent users from exceeding a reasonable level of authority for their positions. This allows various constraints to be applied to the role and permission assignments. One common constraint is that certain roles are declared to be mutually exclusive, meaning that the same person cannot simultaneously hold more than one role from the mutually exclusive set. For example, the roles of student and examiner, or the roles of tenderer (one who submits a tender) and tender officer (one who opens submitted tenders) would both be examples of mutually exclusive sets. Another constraint might be placed on the number of roles a person can hold, or the number of people who can hold a particular role. X.509 only has a limited number of ways of supporting constrained RBAC. Time constraints can be placed on the validity period of a role assignment attribute certificate. Constraints can be placed on the targets at which a permission can be used, and on the policies under which an attribute certificate can confer privileges. Constraints can also be placed on the delegation of roles. However many of the constraints, such as the mutual exclusivity of roles, have to be enforced by mechanisms outside the attribute certificate construct e.g. within the privilege management policy enforcement function.

**SMART TOKENS**

Smart tokens are devices with a memory and processor which can generate and store keys. It also supports cryptographic functions such as encryption, digital signature, or key agreement. Smart tokens come in two forms. The most common takes the form of a rectangular piece of plastic with an
embedded microchip. The second is as a USB token. The two forms have no real difference. Some noticeable characteristics of smart tokens are portability, tamper-resistant storage, and isolation of computation activities (i.e. leveraging the features of cryptographic functions without revealing private keys to other system component). These features make smart tokens ideal devices for PKI use, for a person can store his private keys and corresponding public key certificates in a smart token, and log on a system very easily with high security with this token. We propose to use smart tokens in healthcare also.

PKI/PMI AND SMART TOKEN BASED E-HEALTH SYSTEMS
Similar to many other business sectors, healthcare and welfare are challenged by the new electronic media – Internet, and also a well-crafted and certificate-based security infrastructure is required. After a general discussion about PKIs/PMIs and smart tokens, we will describe how to use public key certificates and attribute certificates in e-health systems. Then, we will give an example for a case study.

The use of public key certificates
When we use public key cryptography to realize ciphers, digital signatures, and strong authentication, an authentic binding between the public key and its owner is needed. Such a binding is provided by public key certificate. Since each public key corresponds to particular private key, a binding of the private key to its owner is given indirectly. By the use of certificates, a lot of attacks fail like the man-in-the-middle attack, replay attack, masquerade attack, etc.

In the healthcare sector, there is a clear distinction between the identity-related part of security services and the professional part, both represented by certificates and related key pairs. So each health professional is issued PKCs, and these PKCs and their corresponding private keys should be carried by their owner conveniently. In several European countries, a smart card is used for this carrier, which is known as health professional card (HPC). In this paper, as mentioned above, a smart token is recommended for such use. Whether a card or a key is used is determined by a professional himself at his convenience.

A PKI in the healthcare sector mainly supports the following three security measures: confidentiality, integrity, and authenticity. Accordingly, each health professional needs at least three different key pairs, to be used for ciphers, digital signatures and strong authentication.

The use of attribute certificates
Attributes about an entity, such as authorization information, are undesirable to be with the entity’s PKC. For two reasons, first, authorization information often does not have the same lifetime as the binding of the identity and the public key. Second, the PKC issuer is not usually authoritative for the authorization information. This results in additional steps for the PKC issuer to obtain authorization information from the authoritative source.
For these reasons, it is often better to separate authorization information from the PKI. Yet, authorization information also needs to be bound to an identity. An AC provides this binding; it is simply a digitally signed (or certified) identity and set of attributes. In healthcare sector however, there is more information besides authorization, like qualification information. A general approach to illuminate several aspects of attribute certificates is show in the documents of the specification of a German electronic doctor’s license[10].

For qualification, attribute certificates firstly point out the profession of a health professional. Examples for some Europe-wide types of professions have been defined by CEN/TC 251 WG7 N45/46, namely: physician, dentist, pharmacist, midwife, nurse, physiotherapist, psychologist, psychotherapist, speech therapist, chiropractor, optician, dental nurse, dispensing pharmacist, administrator, and dental hygienist. Additionally, attribute certificates hold general information about a health professional which means to have a qualification to practice as a specialist with a certain experience on behalf of, and accepted by, regional or national health bodies. Further information might also reflect any kind of knowledge in terms of specific surgeries or investigation methods using devices as, e.g., tomographical devices, ultra sound devices, X-ray devices, etc.

For authorization, attribute certificates hold permission information. As we have mentioned in section 2, permissions are different heavenly at different authorization scheme, so attribute certificates are application-related. Because there are a wide range of processes and procedures deriving from different application area within healthcare sector, and this leads to a large amount of specific and different roles, rules, and access rights for each health professional, so RBAC is more suited to healthcare than other schemes. To adopt RBAC in healthcare, a good model must be designed carefully, which includes role hierarchies and permissions allocated to each role. When implement a PMI, there are two kinds of attribute certificates: role specification attribute certificates and role assignment attribute certificates. Each health professional is issued role assignment attribute certificates, and access resource as the roles declared in the certificates.

A medical scenario
In healthcare sector, different from other business sector, apart from economic and organizational aspects, security and privacy requirements have to be taken more and more into account. As health professionals at all mostly work with highly sensitive medical data, appropriate administrative, technical, ethical, and legal solutions have to be developed and implemented. In the following we intend to explain the basic statements of this paper by introducing a typical example within the healthcare sector.

Contrary to the banking sector where a clerk is very often allowed to get access to client-related personal data, the healthcare sector is managed in a rather different way. Even doctors within the same department of a clinic or hospital are by law not allowed to get access to the same amount of personal data even for the same patient. It depends on the roles they are engaged to play within a
very specific care and treatment process, and the related rules and policies.

From access point of view, this situation should be seen like two triangles (Figure 1). The top represents a lot of people sharing only a few rather general data items about a patient. In the middle section, a diagnostic or therapeutic team has to share much more information about the specific patient in order to treat him well. And at the bottom of this triangle, we find the one and only responsible doctor with access to all medical and administrative patient-related information. So this access aspect could be seen as the major problem of providing

- the right amount of patient-related medical and administrative information
- to the right person
- at the right time

in order to achieve the highest possible quality in the care and treatment process.

This may seem to be rather simple – but it isn’t. Even when it comes to the example explained before (doctors in the same department of a clinic) it very much depends on their roles within the care process. If one of them became the caring doctor of a specific patient, he has of course access to all data. This is both a legal and ethical requirement. If the other doctor is “just” another doctor of this department he may have access to data which are relevant for the services he provide for the department. But as soon as he might become the deputy chair of the department (even for a couple of hours if the chair is absent for a surgery or a meeting) he has to have access to the medical data. In this case, he (as the deputy chair) is responsible for all activities provided by his team.

This example might have illuminated that the rights of professional in healthcare consist both static and dynamic permissions and admissions. Due to that, the content of attribute certificates varies from case to case. Basically, static permissions may be issued by health authorities, dynamic permissions may be issued even by employers (hospitals, clinics). Static permissions can be connected with dynamic permissions by a supervisor or administrator within the IT environment of a hospital. This kind of realizing application security functions is even possible only identity-based.
CONCLUSIONS

Secure access to patient-related medical and administrative data guaranteeing both security and privacy is one of the main challenges in e-health systems. Security infrastructures should be built to meet these security and privacy requirements. We have given a general discussion about PKI/PMI and a case study of how to use public key certificates and attribute certificates in e-health systems. It’s just a beginning, for in China, many hospitals just have Hospital Management Information Systems (HMIS), and only a few have Electronic Medical Records (EMR). Furthermore, we have no legal acts to protect electronic patient records. We still have a lot of works to do.

To build a security infrastructure for healthcare, national or regional health authorities play an important role for their knowledge of health professionals. A formal act to protect electronic patient records is also needed. From technique point of view, a well-crafted authorization model is critical for protecting patients’ records. This needs further study.

REFERENCES