An E-Hospital Security Architecture

Fang Tian and Carlisle Adams

School of Information Technology and Engineering (SITE),
University of Ottawa, Ottawa, Ontario, Canada K1N6N5
{Ftian030,cadams}@uottawa.ca

Abstract. In this paper, we introduce how to use cryptography in network security and access control of an e-hospital. We first define the security goal of the e-hospital system, and then we analyze the current application system. Our idea is proposed on the system analysis and the related regulations of patients' privacy protection. The security of the whole application system is strengthened through layered security protection. Three security domains in the e-hospital system are defined according to their sensitivity level, and for each domain, we propose different security protections. We use identity based cryptography to establish secure communication channel in the backbone network and policy based cryptography to establish secure communication channel between end users and the backbone network. We also use policy based cryptography in the access control of the application system. We use a symmetric key cryptography to protect the real data in the database. The identity based and policy based cryptography are all based on elliptic curve cryptography—a public key cryptography.

Keywords: Availability, JADE, Layered Security Protection, Identity -Based Cryptography, Elliptic Curve Cryptography, Public Cryptography.

1 Introduction

Nowadays, hospitals are using electronic medical records and electronic applications to improve efficiency and thorough care for the patients. At the same time patients' privacy protection is becoming more serious compared with traditional paper-based hospitals. Patients' privacy might be exposed by malicious attacks of network or e-hospital application system or even by inappropriate access control rules. Some attacks such as denial of service can even cause e-hospitals not to be accessed at all. Therefore an appropriate security architecture can not only provide protection but also thwart the malicious attacks.

In this paper, we introduce the security architecture for an e-hospital. The whole security architecture of the system is a large project ranging from physical, premier, and network security to data security, etc. We reach our security goals of the e-hospital on the basis of the current system analysis and requirements of related patients' privacy protection regulations. We adopt layered security protection in the whole system, with the consideration of balancing security and availability. We use

J. Zheng et al. (Eds.): ADHOCNETS 2009, LNICST 28, pp. 639–651, 2010.
© ICST Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering 2010

identity-based cryptography in the secure communication channel establishment in the backbone network; we use policy based cryptography to provide a secure communication channel between user and backbone network; we also use policy based cryptography to provide access control and data confidentiality; and we use a symmetric key cryptography to protect real data in the database. The encryption and decryption of real data works at the page level when data are written to or read from a disk therefore is totally transparent to the application system.

The identity based and policy based cryptography we use are similar. They are all public key cryptography based on elliptic curve cryptography; they all use an elliptic curve defined over finite fields and suitable for some resource-constrained devices like the wireless end devices used in the current application system; and they all use public key cryptography to protect a symmetric session key which will be used in the protection of real communication between two parties. The difference between these two is that in the identity based cryptography, the identity is the public key and in the policy based cryptography the policy is the public key.

The paper is organized in the following way: In section 2, backgrounds and related work of this system have been introduced. This section includes the introduction of the current application system, identity-based encryption, policy-based encryption and elliptic curve cryptography; in section 3, we describe the security goal of the e-hospital system; in section 4, we propose the whole security architecture of the system. We also describe the algorithm of identity-based, policy based cryptography, and how to protect the real data in the database; we end the paper with our conclusion and future work in section 5.

2 Background and Related Work

In this section, we introduce one e-hospital, its workflow, its components, and its implementation environment. We introduce identity-based, policy-based, and elliptic curve cryptography.

2.1 E-Hospital System

2.1.1 E-Hospital Introduction

This e-hospital research is about creating a methodological framework for anytime-and anywhere-decision support (A3Support) for emergency department (ED) triage decision-making. The e-hospital system is a multi-purpose clinical decision support system and is currently used in pediatric asthma. It uses data mining technique to provide fast and possible solutions to the ED doctors and nurses. The system is designed to provide wired and wireless access for the ED staff. The end-user devices used to access the system can be PCs, laptops, or mobile phones [1].

In the function implementation, multi-agent architecture has been adopted in this e-hospital application system. This multi-agent is inspired by the Reusable Environment for Task-Structured Intelligent Networked Agents (RETSINA) by Carnegie Mellon University. The structure of the application system reflects the management workflow in the following way: Each user group has a corresponding interface agent; each

patient management task has a corresponding task agent; each hospital system has a corresponding information agent. The implementation environment is JADE, an agent-orientated language developed on Java. JADE is a middleware for the development of peer-to-peer intelligent-agent applications. It runs seamlessly in mobile and fixed environments, which enables multi-party applications, pro-activity, and the machine-to-machine paradigm.

2.1.2 Related Work in Security Issues in a Mobile Agent System

The e-hospital system is a mobile multi-agent system. A mobile agent is a software program with mobility which can be sent from a computer into a network and roam among the computer nodes in the network. Nowadays, mobile agent paradigm helps form a large-scale loosely-coupled distributed system and because of its many salient merits it has attracted tremendous attention. While at the same time, mobile agent technology also brings significant new security threats because the mobile code generated by one party will transfer to and execute in an environment controlled by another party. Those security problems have become the bottleneck of the development and maintenance of mobile agent technology, especially in security sensitive applications such as electronic commerce and electronic hospital.

A lot of research has been dedicated to address the security problems in a mobile agent system. This research differs in its aim, emphasis, base and technique. Chess et. al. summarized the assumptions violated by mobile agent systems that underlie most existing computer security implementations such as authentication, reputation and trust [13]. Farmer, Jansen, Rothermel and etc. analyzed mobile agent system, overviewed the threats and security problems and identified security objects and requirements in [14].

Over the years, a number of mobile agent systems with security features have been developed and applied. Most of the secure mobile agent systems are trying to address the security issues like authentication, authorization, privacy and confidentiality. For example: Agent Tcl developed by Dartmouth College [15] is using public-key cryptography to authenticate the identity of the mobile agent but the authorization is based on access control lists and at a coarse granularity; Aglet [16] of IBM uses a database to implement a relatively fine grained access control but the authentication is implemented by Message Authentication Code; Ajanata [17] and Ara [18] use domain and place to provide protection for mobile agents; Condordia [19] created by Mistubishi Electronic Information Center America has a strong focus on security and reliability. Cryptography has been used in authentication and mobile agents migration.

The mobile agent systems mentioned above all use java in their implementation. Most of the security mechanisms mainly rely on Java features. The e-hospital uses java technology in its function implementation and therefore we will address most of the security issues using java security features as well. From all the mobile agent systems mentioned above, we can see a lot of them use cryptography in authentication, data confidentiality, data integrity, and network security. Our research is focused mainly on cryptography and how to use it in architecture of this e-hospital. We will describe the cryptography we use in this application system in the following sections.

2.2 Related Cryptography

2.2.1 Identity-Based Cryptography

Identity-based cryptography was first proposed by Shamir in 1984. The basic idea is to simplify the key management of some Internet applications using public key cryptography. It works in the following way: Suppose Bob wants to send some confidential information to Alice. First, Bob will encrypt the confidential information by a symmetric key and then he will encrypt this symmetric key using Alice's identity, such as her email address. The email address here is Alice's public key and, supposedly, only Alice has the corresponding private key. After receiving the encrypted message, Alice uses her private key to decrypt and get a symmetric key. She can then use this symmetric key to decrypt the encrypted message. The channel through which Alice gets her private key from a private key generator (PKG) is supposed to be a secure one. Identity-based cryptography provides a secure way to transmit a session key or a symmetric key over an insecure communication channel such as the Internet. [2]

Identity-based cryptography is a public key cryptography, but it is different from the traditional public cryptography in which both parties should have their private keys before communication occurs, and there is no key management like that in the traditional public key system. Identity-based cryptography uses elliptic curve cryptography to encrypt and decrypt. The elliptic curve used in the cryptography is based on the hardness of discrete logarithm problems of the elliptic curve over definite fields.

2.2.2 Policy-Based Cryptography

Policy-based cryptography is similar to identity-based. The difference is that in policy-based cryptography, specific policy or rules are used as the recipient's public key. Only those who are compliant with the policy can decrypt the message. Molva and Bagga propose a policy-based encryption system. [3] In their system, policies are formalized as monotonic logical expressions involving disjunctions (denoted $\vee$) and conjunctions(denoted $\wedge$) of conditions. Each condition is fulfilled by a specific credential issued by a certain trusted authority TA. An entity fulfills a policy if and only if it has access to a set of credentials associated to the logical combination of conditions defined by the policy.

Policy based cryptography can be used to provide data confidentiality and access control as well. Policy based cryptography can be used in the scenario where the access control rules are complicated and hard to be implemented by rules of network devices or application systems. For example if the access control policy of a certain type data "res" of company A is: "res" can be read if the user is from company B , he should be a member of organization C, the access time is between 9am and 5pm and he must be in place D. This policy is hard to be fulfilled by setting rules in the network devices or database and it may involve cross certificate of multi companies in traditional PKI system. Policy based cryptography provides a possible solution to the case above. We will explain in details how policy based cryptography works in section4.

2.2.3 Elliptic Curve Cryptography (ECC)

Identity-based cryptography and policy-based cryptography are founded on the hardness of discrete logarithm problems of elliptic curves defined over finite fields.

Elliptic curves used in cryptography are typically defined over two types of finite fields: fields of odd characteristic and fields of characteristic two. We use ECC defined over the first one. Elliptic curve over odd characteristic fields Fp, where p>3 and is a large prime, can be written as $y^2=x^3+ax+b$, where $a\in$ Fp, $b\in$ Fp, and $4a^3+27b^2 \neq 0$. The elliptic curve can obtain the same security level with a much shorter key size compared with RSA or DSA. This attribute makes it well suited for systems with constraints on processor speed, security, heat production, power consumption, bandwidth, and memory. Cell phones, PDAs, wireless devices, laptops, and smartcards are applications that benefit from elliptic curve. The following table is a comparison between ECC and other encryption schemes.

Table 1. Comparable key size of different encryption scheme [7]

Security level (in bits)	Symmetric scheme (key size in bits)	ECC-based scheme (size of *n* in bits)	DSA/RSA (modulus size in bits)
56	56	112	512
80	80	160	1024
112	112	224	2048
128	128	256	3072
192	192	384	7680
256	256	512	15360

3 Security Goal of E-Hospital

In this section, we describe how we reach the security goal of the e-hospital system. We look at the related standards and regulations of patients' privacy protection; we consider the users' requirements and the expectations of the e-hospital system; and we refer to the security goal of the original security design.

3.1 Related Privacy Acts and Regulations

The security goals of this project come from the privacy protection act and regulation, namely the Personal Health Information Protection Act (PHIPA); end users' requirements of the e-hospital application system; and health level seven (HL7) standards.

PHIPA is privacy protection legislation built on the Personal Information Protection and Electronic Document Act (PIPEDA) guidelines for the protection of personal electronic information. PHIPA is informed by 10 principals to protect personal information, namely accountability; identifying purpose; consent; limiting collection; limiting use; disclosure and retention; accuracy; safeguards; openness; access; and challenging compliance.

The HL7 standard is another guideline to which we refer. HL7 specifies the security framework for medical data exchange and addresses the following levels or layers of communication: link, network, transport, session, and application. For each of these layers of communication, HL7 addresses the following security services: authentication; authorization and access control; system and data integrity; confidentiality; accountability; availability; and non-repudiation.

3.2 User Requirements

We investigated the e-hospital system, and the following are major concerns in reaching our security goals: 1. The "circle of care," which defines members of the health care team who are involved in providing care and treatment to a particular patient. Members of a circle of care can collect and use the patient's personal health information for that care, unless they know that the patient has expressly withheld or withdrawn consent. 2. "Lock boxes," which specify that though some medical staff has privilege to patients' documents, patients have rights to have their health information withheld from the members of a circle of care

The above two concerns can help us to understand who has what privilege to which patient's medical records, and a picture of "need to know" is clear. These concerns are not defined in the act but are very important conceptions and should be considered in the practice.

Another concern is the data mining technique used in the e-hospital application system. Since data mining is used in the whole triage, all of the data from the database to data mining tool must be in clear text or else the data mining tools cannot work properly. This is very important in designing our security architecture. We need to guarantee the whole system security and be sure of the availability of the application system. Availability means the data-mining tool must run properly and those with authorization can access the system as they used to.

3.3 Security Goal of E-Hospital

The security goals of the original design are as follows: 1. Sensitive information should be protected with a higher level of security. 2. Methods of protection should include technological methods such as the use of passwords and encryption 3. Access should be limited to a need-to-know basis. 4. Per PHIPA and PIPEDA specifications, medical records should be protected from loss, theft, unauthorized access, disclosure, copying, use, or modification, regardless of their format.

PHIPA also specifies the following: 1.From the point of availability, the data should be accurate, up-to-date, and accessible when needed. 2. Auditing is a big part of medical practice, and the accountability goal is to make sure access is based on a legitimate need to know, and no collision exists. 3. PHIPA also specifies the integrity and non-repudiation of medical data.

In order to reach the above security goals, we should be clear about information labeling. How many data categories are there in the current application system? In the current e-hospital system, one patient has one hospital document, which consists of four medical records. All four types of medical records are confidential, according to PHIPA. Therefore, we must adopt the proper mechanism to protect all of them. The

protection mechanisms we plan to use include but are not limited to the following: physical security protection, such as a specific server room with access control or CCTV; network security protection, such as an end-to-end secure communication channel; network perimeter protection, such as a firewall or intrusion detection system; backbone network security protection, such as a secure domain or network separation; two-factors authentication; database encryption; operation security, such as a disaster recovery procedure; and staff security, such as duty separation and job rotation. In the following section, we will describe in detail what we will do to meet the security goals, with the consideration of security and availability balance.

4 Security Architecture of E-Hospital

In this section, we describe the security architecture of this e-hospital. We propose our security architecture on the basis of current system analysis and related regulations of patients' privacy protection.

When we approach the security architecture for a system, we notice there are different protection mechanisms working at different levels. For example, a security guard can scare away evildoers, but what if an intruder attacks the system from a network? Here, the intrusion detection system can provide some level of security. In order to strengthen the security features of a system, we cannot rely on only one protection method; we should use different layers of security strategy to protect the whole system. We will describe in detail how we design a layered protection system.

4.1 Layered Design to Meet Security Goals

We use a layered security design to provide different levels of protection. We define different security domains, and each security domain has a different trust level. After defining different security domains, parts of the access control rules in the system are clear. We define three security domains in the system, namely, domain1, domain2, and domain3. Domain1 has the highest security level and the most trustworthy. If domain1 has been successfully intruded upon, the whole system security will be jeopardized. Domain3 has the lowest trust level, and domain2 is in the middle. The different domains include the following :

Domain1: The database and synchronize agent are in domain1. The original medical records are in domain1, and according to the current policy, all data in this domain are read-only.

Domain2: Task agents are in domain2, and they are between domain3 and domain1. All of the access requests from domain3 are handled by domain2. Domain3 cannot access domain1 directly. Only domain2 can access domain1 on behalf of domain3, and it is read-only.

Domain3: User interface agents are in domain3. It can only access domain2 and in a read-only mode.

The access control workflow is illustrated in figure 1.

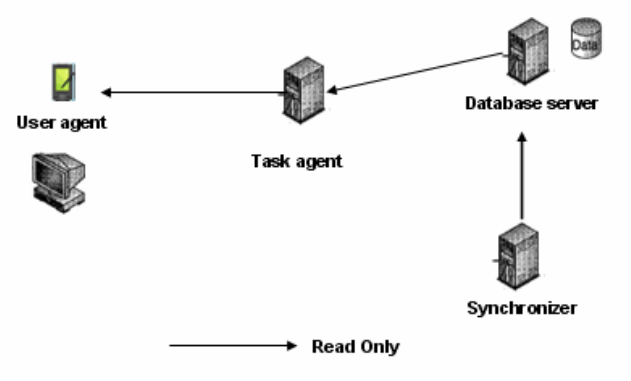

Fig. 1. Access Control Flow

Now, we have a clear picture of the whole system architecture. According to different domains, we provide different protection mechanisms. Our layered protection is illustrated below.

For domain3, we will provide the following protection for end users. We will use antivirus and a personal firewall to provide protection against malicious codes. We will keep the original authentication in the application system. In the original system they use two-factor authentication: usename/pwd and RFID card.

Domain2 is in the backbone system and in the middle of domain3 and domain1. Protection for domain2 falls into perimeter security protection. We will use a firewall, intrusion detection (network sensor to detect network attacks, and host sensor to detect possible attacks to server), and access control rules from network devices such as gateway routers.

For domain1, which is the most secure in the whole system, we will use another firewall to separate it from domain2. This firewall can guarantee domain1 and domain2 are in different networks. Even if the attacker breaks domain2, he/she still needs to break the firewall and other protection in domain1 to crack the whole system. Since original data are in domain1, we should adopt some mechanism to protect it from malicious attackers and users without the need-to-know privilege. We will use a symmetric key cryptography to encrypt all the data in the database. This symmetric key cryptography is totally transparent to the application system. All of the encryption or decryption occurs when the data are written to or read from the disk. The key used to protect the data will be protected under master key of the database and this master key is maintained by another security administrator not the database administrator. The reason we need a security administrator is that the database administrator can only have least privilege and need to know.

Now, the only thing left in the system is the transmission channel protection. The e-hospital system provides wired and wireless access for the end users. A WEP protocol has been adopted. WEP works in the data-link layer and is not a secure protocol. We must provide some protection from the network layer or transport layer before the application begins to exchange data. In real world practice, SSL is used to provide network security and end users' access control is fulfilled by setting rules in

application system or database. In this e-hospital we have noticed the access policy for end users is hard to meet by rules from database or application system.

The access policy is composed by users' group information, the access time information, the access location information and some other constraints such as if patients have withhold their health information from some members of the group. As mentioned before policy based cryptography provides a possible solution to access control with complex policies. And since policy based cryptography is a public key cryptography, it can be used in providing network security as well. We can use policy based cryptography to protect the communication session key between the user and the backbone network. Only those who are compliant with the policy can decrypt and get the communication session key.

For the backbone network, since the access policy is relatively simple, all the communication in the application system is done through agents and each agent has its unique ID, we use identity based cryptography in a key agreement protocol and thus get a session key between two communication parties. The whole system architecture is illustrated in figure 2.

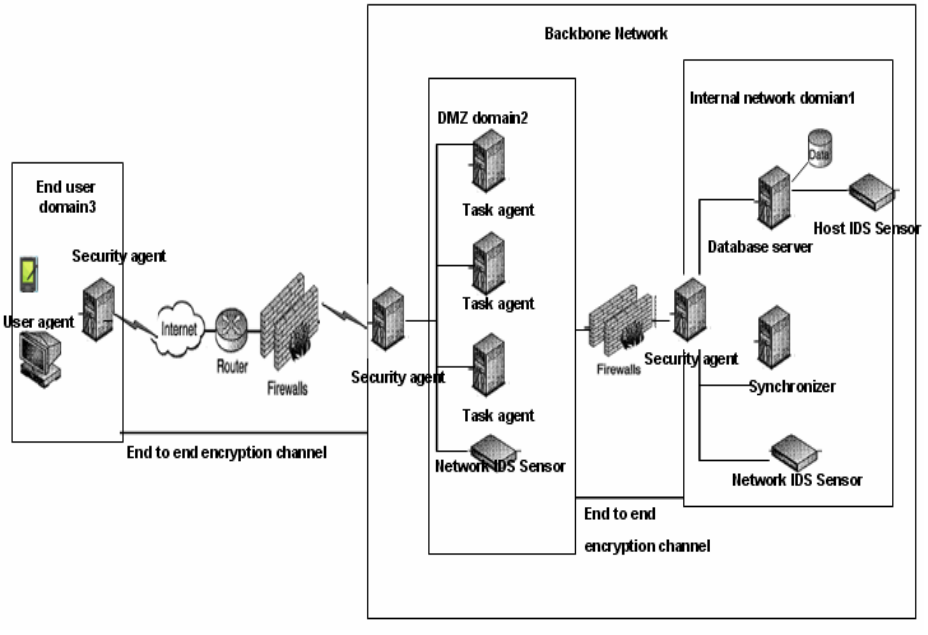

Fig. 2. E-hospital Security Architecture

4.2 Secure Communication Channel by Identity-Based Cryptography

In the backbone network, we use identity-based cryptography in the computation of the session key. The session key will be used as the key to encrypt the real communication data between agents in the backbone network. The agents' ID is its public key. We will deploy our secure agent in both communication parties. If the

session is over or times out, the session key and all of the information protected by this session key will be erased, which guarantees no medical data can be used outside of the hospital. The algorithm we use to compute session key is similar to section 4 in [10]. The algorithm we use to encrypt or decrypt the real data is a symmetric algorithm AES. The algorithm and how it works are explained below:

1. System set up: Parameters set up
In this stage, parameters of the elliptic curve and master key pair of the TA are generated.

Params=<q, G1, G2, e, n, P,H1, H2, s, P_S > where

q is the large prime over which the elliptic curve is defined;

P is the generator of G1;

e is a mapping function: e:G1×G1->G2;

n is the order of G1 and G2;

H1 is a hash function: H1: {0,1}*->G1*;

H2 is a hash function: H2: G2->{0,1}*

s is the master key and a random number where s∈ Zp;

Ps is the public key of the TA and Ps=s*P;

2. End-Agent's key pair generation
Each agent's public key Q is its identity Qi=H1(IDi) where i is the id of the agent;
The corresponding privet key Si=s*Qi where s is the TA's private key
3. Secure communication channel establishment:

In this step the two communication parties use Identity-Based cryptography to compute a shared secret as their session key and use this session to encrypt the following communication data.

Agent A chooses a random number a ∈ Zp and send Wa=a*Qa to agent B where Qa is A's public key;

Agent B chooses a random number b ∈ Zp and send Wb=b*Qb to agent B where Qb is B's public key;

Agent A computes Kab=e(Sa, Wb+a*Qb) where Sa is agent A's private key and Sa=s*Qa

Agent B computes Kba=e(Sb, Wa+b*Qa) where Sb is agent B's private key and Sb=s*Qb

Kab=e(Sa,Wb+a*Qb)=e(s*Qa,b*Qb+a*Qb)=e(s*Qa,Qb*(a+b))=e(Qa,Qb)$^{s(a+b)}$

Kba=e(Sb, Wa+b*Qa)=e(s*Qb, a*Qa+b*Qa)=e(s*Qb, Qa*(a+b))=e(Qa,Qb)$^{s(a+b)}$

Kab=Kba and can be used as a shared secret between agent A and agent B;

The session key K=H2(Kab);

4. Secure communication:

The session key is used to encrypt the communication between the agents in the backbone network. The algorithm used is AES which is a symmetric algorithm.

5. Session over: In this stage, the session key and information will be erased from both communication parties.

4.3 Secure Communication between End User and Backbone Network by Policy Based Cryptography

We use Molva&Bagga policy based cryptography scheme in the e-hospital to establish a secure communication channel between end users and backbone system. We also use policy based cryptography in end users access control. For different access policy we have different corresponding session keys. Those session keys are symmetric keys and used to protect the communication between the end user and the backbone network. Those session keys are protected by the corresponding policies. First the end user sends a communication initialization request to the task agent in the backbone system; the task agent searches the policy and the session key for the end user, sends back the session key protected by the corresponding policy; the end user tries to decrypt using the corresponding credential issued to him by a private key generator and get the session key; after the above steps all the communication between the end user and the task agent in the backbone network will be encrypted by this session key. When the session is over the session key will be erased from end users.

The access policy is presented by conjunctions ($\wedge$) and disjunctions ($\vee$) of conditions. Each condition is defined through a pair of <TA,A> where A is an assertion and TA is a trusted authority who is responsible for checking and certifying A's validity and issuing credentials corresponding to valid assertions. An assertion can be the information about the subject's attributes, properties or capabilities etc. For each condition <TA,A>, the credential denoted as $\varsigma(R,A)$ can be generated only by TA using its secret master-key s and the validity of the credential can be checked using TA's public key R. In policy-based cryptography, the pair <TA,A> is a public key for the condition and its corresponding credential is its private key.

The user has been issued the credential $\varsigma(R,A)$ if and only if he fulfils the condition <TA,A>. Let the expression 'user$\leftarrow \varsigma(R,A)$' denotes the fact the 'user' has been issued credential $\varsigma(R,A)$ and le t the expression 'user$\leftrightarrow$<TA,A>' denotes the fact that the 'user' fulfills the condition <TA,A>. Then we got the following property: user$\leftrightarrow$<TA,A>$\Leftrightarrow\varsigma(R,A)$. A policy denoted as 'pol' is written in conjunction-disjunction normal form: $Pol=\wedge_{i=1}^{m}[\vee_{j=1}^{mi}[\wedge_{k=1}^{mi,j}<Tai,j,k,Ai,j,k>]]$.

If a user fulfills a policy, he has been issued all the corresponding credentials of the conditions in the policy. Let 'user$\Leftrightarrow$pol' denotes the fact that user meets the policy and let the expression 'user$\leftarrow\varsigma(R,Ai,j,k)$' denotes the fact that the user has been issued all the credentials included in the policy. Then the following property holds. User $\Leftrightarrow pol\Leftrightarrow\forall i\in\{1,\ldots,m\},\exists ji\in\{1,\ldots,mi\}$:user$\leftarrow\varsigma ji,\ldots jm(pol)$. From the description we can see if the user has been issued all the corresponding credentials in a policy then he is supposed to meet the policy. In this way policy based cryptography can be used in access control.

The algorithm used in policy based cryptography is similar to identity-based cryptography described above. The parameters set up and key pair generations are almost the same while in policy based cryptography the public key of a condition is H1(A) where A is an assertion and its corresponding private key is s*H1(A) where s is the master key of TA. The difference is the encryption and the decryption procedures. Each conjunction of conditions is associated to a kind of mask denoted $\mu_{i,j}=H2(g_{i,j}^{r}||i||j)$. For each index i, a randomly chosen key ti is associated to the disjunction $\vee_i=\vee_{j=1}^{ai}\wedge i,j$. Each ti is encrypted ai times using each masks $\mu i,j$ and thus the end user can decrypt if he has one of the credentials in the disjunction conditions. The details can be found in [3].

4.4 Database Security

In the database protection, we plan to encrypt the database contents at rest in the database. We use a symmetric key cryptography to perform real-time I/O encryption and decryption of the data and log files. The encryption uses a database encryption key (DEK). The DEK is a symmetric key secured by master key of the database which is a asymmetric key.

Encryption of the database file is performed at the page level. The pages in an encrypted database are encrypted before they are written to disk and decrypted when read into memory. Thus this symmetric key cryptography is totally transparent to application system and no changes are needed in the application system.

This symmetric key cryptography is very suitable to protect data from intruders and insiders. If the DBA has no privilege to access the original data, he/she cannot access the data. But to guarantee the DBA cannot access what he/she is not supposed to, we should adopt duty separation. The key generations and backups in this symmetric key cryptography should be done by a security administrator other than the DBA.

5 Conclusion and Future Work

In this paper, we introduce the security architecture of an e-hospital. We propose our idea through layered protection. We use a different protection mechanism to provide different security levels for the system. We use cryptography to strengthen the network security and access control to meets the security goals of the e-hospital.

In this paper, we focus more on the technical security design of the e-hospital system. We don't consider much more about the administrative and physical controls. For example, some administrative controls include security policy, user awareness training, etc. Without effective administrative controls, even if we have strong technical protection, the application system may still suffer from attacks outside or inside. The physical control such as locks, CCTV, security guards, etc. plays the same important role in the design of the whole security features of the application system too. In the future, we will consider more about physical and administrative controls, such as security policy, physical security, and some operational controls, such as disaster recovery and business continuity for the fulfillment of the availability requirement of PHIPA.

Additionally, the performance test and improvements will be future work as well.

References

1. Michalowski, W., O'Sullivan, D., Matwin, S., Wilk, S., Farion, K.: Designing an Agent to Support the Retrieval of Medical Evidence to Support Emergency Physician Decision Making at the Point of Care. In: CSM 2007: The 21st Workshop on Complex Systems Modeling, Laxenburg, Austria (August 2007)
2. Boneh, D., Frankliny, M.: Identity-Based Encryption from the Weil Pairing. Siam J. of Computing 32(3), 586–615 (2003); Extended abstract In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 213–229. Springer, Heidelberg (2001)
3. Bagga, W., Molva, R.: Policy-Based Cryptography and Applications. LNCS, pp. 72–87. Springer, Heidelberg (2005)
4. Garson, K., Adams, C.: Security and Privacy System Architecture for an e-Hospital Environment. In: Proceedings of the 7th Symposium on Identity and Trust on the Internet (2008)
5. Bellifemine, F.: Telecom Italia, Italy Giovanni Caire, Telecom Italia, ItalyDominic Greenwood, Whitestein Technologies AG, Switzerland. In: Developing Multi-Agent Systems with JADE, ISBN: 978-0-470-05747-6 (HB)
6. http://jade.tilab.com/
7. SECG: STANDARDS FOR EFFICIENT CRYPTOGRAPHY, http://www.secg.org/download/aid-385/sec1_final.pdf
8. HL7 Secure Transactions Special Interest Group: Health Level Seven Security Services Framework (2008), http://www.hl7.org/Library/data-model/SOP_980123_final.zip
9. Personal Health Information Protection Act (PHIPA 2004), http://www.health.gov.on.ca/english/public/legislation/bill_31/personal_info.html
10. Chen, L., Kudla, C.: Identity Based Authenticated Key Agreement Protocols from Pairings. Trusted systems Laboratory HP Laboratories Bristol HPL-2003-25 (February 12, 2003)
11. Personal Information Protection and Electronic Documents Act (PIPEDA 2000), http://www.privcom.gc.ca/legislation/02_06_01_01_e.asp
12. Understanding Transparent Data Encryption, http://msdn.microsoft.com/en-us/library/bb934049.aspx
13. Chess, D., Harrison, C., Kershenbaum, A.: Mobile Agents: Are They a Good Idea? IBM Research report, J.Watson Research Center (March 1995)
14. Farmer, W.M., Guttmann, J.D., Swarup, V.: Security for Mobile Agents: Issues and Requirements. In: Proceedings of the 19th National Information Systems Security Conference, vol. 2, pp. 591–597 (1996)
15. Gray, R.S.: Agent Tcl: A Flexible and Secure Mobile Agent system. In: Proceedings of Fourth Annual Usenix Tcl/Tk Workshop, pp. 9–23 (1996)
16. Karjoth, G., Lange, D.B., Oshima, M.: A Security Model for Aglets. In: Vigna, G. (ed.) Mobile Agents and Security. LNCS, vol. 1419, pp. 188–205. Springer, Heidelberg (1998)
17. Ksrnik, N., Tripathi, A.: Agent Server Architecture for the Ajanta Mobile-Agent System. In: Proceedings of the 1998 International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA 1998), July 1998, pp. 66–73 (1998)
18. Peine, H., Stolpmann, T.: The architecture for the Ara Platform for Mobile Agents. In: Rothermel, K., Popescu-Zeletin, R. (eds.) MA 1997. LNCS, vol. 1219, pp. 50–61. Springer, Heidelberg (1997)
19. Wong, D., Pacoprek, N., Walsh, T., Dicelie, J., Yong, M., Peet, B.: Concordia: An infrastructure for Collaborating Mobile Agents. In: Rothermel, K., Popescu-Zeletin, R. (eds.) MA 1997. LNCS, vol. 1219, pp. 86–97. Springer, Heidelberg (1997)