A Flexible Role-based Secure Messaging Service: Exploiting IBE Technology in a Health Care Trial

Marco Casassa Mont, Pete Bramhall, Chris R. Dalton
Keith Harrison
Trusted Systems Laboratory
HP Laboratories Bristol

HPL-2003-21
February 6th , 2003*

E-mail: marco_casassa-mont@hp.com, pete_bramhall@hp.com, chris-r_dalton@hp.com, keith_harrison@hp.com

IBE, secure messaging, RBAC, access control, health care, security, privacy

The management of confidential and sensitive information is a major problem for people and organisations. Dynamic organisations require secure solutions to protect confidential documents against unauthorised access and to cope with changes to people's roles and permissions. Solutions based on traditional cryptographic systems and PKI show their limitations, in terms of flexibility and manageability. This paper describes an innovative technical solution in the area of role-based secure messaging that exploits Identifier-based Encryption (IBE) technology. It illustrates the advantages over a similar approach based on traditional cryptography and PKI. It discusses a few open issues. A secure messaging system based on our technology has been fully implemented and it is currently used in a trial with a major European health service organization.

* Internal Accession Date Only

Approved for External Publication

1. Introduction

Communication is a key aspect and peculiarity of human beings. People communicate in every context of their life, to share information, ideas, problems, plans, etc. Sometimes communication involves the exchange and disclosure of sensitive information. Only people with the right roles and permissions are entitled to access this information.

Proper actions need to be taken in order to ensure confidentiality and privacy. It is not straightforward to achieve this goal. Modern societies and organisations are more and more complex, dynamic and flexible.

In industry, employees cover different roles and activities, both within their working environment and whilst interacting with external organisations. Taskforces and working groups are dynamically created and torn down by collaborative organisations, in short periods of time, to achieve common objectives. People's roles, rights and duties change because of frequent reorganisations and because of the evolution of market and customers' needs. People with specific skills can be allocated to tasks on demand, for a predefined period of time, to solve problems or provide a service. Information is continuously generated and exchanged between all the involved parties, including confidential and private information.

Similar scenarios happen in government organizations, including health care organizations, tax offices, government agencies and military organisations. The need to increase the quality of the service and be more effective requires a better usage of the personnel and their skills. More and more frequently employees with similar skills are organised in pools of resources and they are interchangeable when asked to play specific roles. For example, in the health care service general practitioners, other doctors and specialist consultants are considered precious resources and they are allocated, on demand, on specific patients' problems. This dynamism has strong implications on how patients' confidential data has to be managed and how their privacy is preserved.

Dynamic organisations need secure solutions to store and exchange confidential information and protect it against unauthorised accesses or disclosures. These solutions need to be flexible enough to cope with dynamic changes of people's roles and permissions.

New technologies have progressively simplified the way people communicate. In particular, in the last decade there has been an exponential increase of the usage of the e-mail service to exchange any kind of digital document.

On one hand the e-mail service allows an easy, almost instantaneously and asynchronous transmission of digital information at a fraction of the costs of traditional mechanisms. On the other hand, providing a secure e-mail service is a non-trivial problem, especially when the exchanged information is confidential and private. Confidential data need to be protected to avoid unauthorised accesses both during its transmission and when it is stored at its destination.

Secure messaging solutions, based on traditional cryptographic systems and PKI show their limitations in dynamic contexts, in term of flexibility and manageability.

Innovative technical work has been done in this area by the Trusted Systems Laboratory (HP Labs, UK) by leveraging the Identifier-based Encryption (IBE) technology and related intellectual properties (IPs).

The next sections provide more details about the addressed problem, scenarios and our technical approach.

2. Addressed Problem

The key problem addressed is this paper is the enforcement of confidentiality and privacy in dynamic contexts, where people's roles and permissions are subject to frequent changes.

In dynamic contexts people can play different roles at different time, depending on workforce availability and on required skills. Disclosure policies can dictate the terms and conditions under which confidential information can be disclosed. An important aspect of the problem is making sure that these polices are enforced and cannot be subverted. Confidential information needs to be carefully protected and exchanged in a way that only the entities that satisfy predefined constraints and policies at specific point in time are allowed to access it.

We focus on the problem of providing a role-based secure digital messaging service to enable secure communication in dynamic organisations. We describe a few real-life scenarios in the health care context, we highlight key requirements and we present an innovative solution we believe has advantages against solutions purely based on traditional PKI/cryptographic systems, in terms of simplicity, manageability and flexibility.

3. Scenarios

This section describes a few scenarios that highlight real-life interactions between workers in a major European health service organization. At moment, most of these interactions happen by exchanging traditional paper-based documents. The objective is to automate this exchange of information by using a flexible and secure e-mail service and improve the overall quality of the service.

Figure 1 shows a high-level interaction model involving general practitioners (GPs) and members of a department of the health care organisation:

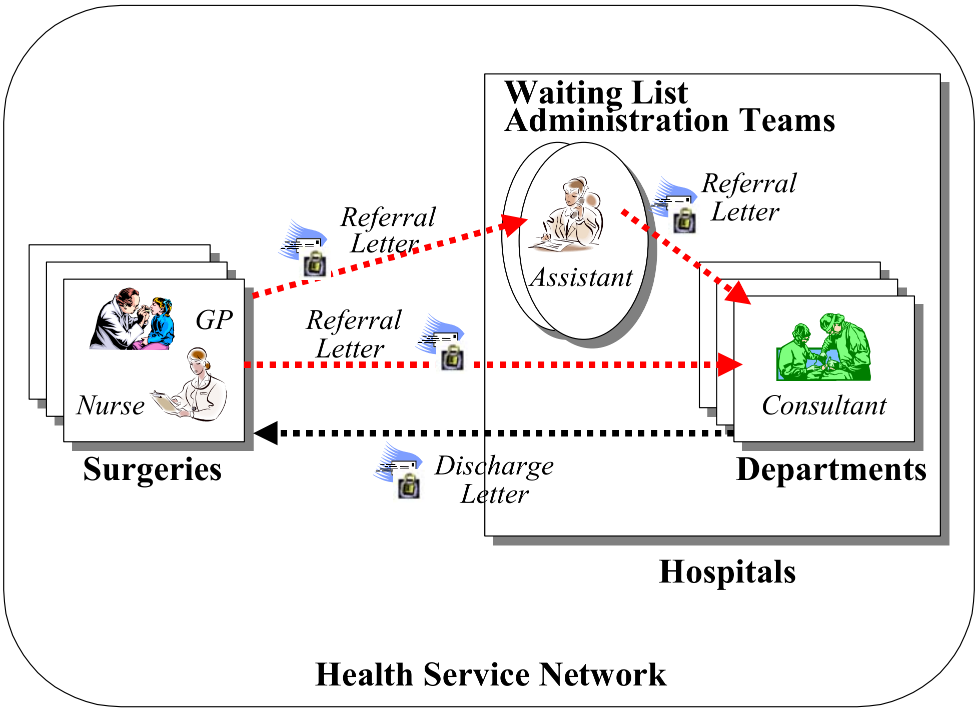


Figure 1: Interaction Diagram

A general practitioner (GP) or his/her assistants might need to send referral letters to hospital consultants, containing patients' confidential information. The GP might have no idea of which specific consultant is on duty or which specific person needs to be contacted. On the other hand, the GP has clear in his mind the role of the person he/she is willing to communicate to.

Waiting list administration teams (at the hospital premises) are in charge of dealing with health care requests for patients and allocate them to the available resources. Members of the team could be asked to act as information "routers". In particular circumstances, they might not be allowed to access or read the content of the requests because of the confidentiality of patients' data. Roles can be dynamically (re)-allocated, depending on timetables, availability or lack of medical personnel. Documents, such as [1], describe guidelines to deal with the access of patients' confidential data, based on people's roles.

Figure 2 shows a scenario where a GP, at a surgery, sends patient's confidential information (referral letter) to a hospital, by e-mail. The GP directly interacts with the hospital's Waiting List Administration Team that is in charge of dispatching the confidential information to an appropriate consultant.

In this case, the GP is happy if any member of the "Waiting List Administration Team" group accesses the content of his message. On the other hand, nobody else must access it unless authorised by the waiting list administration team:

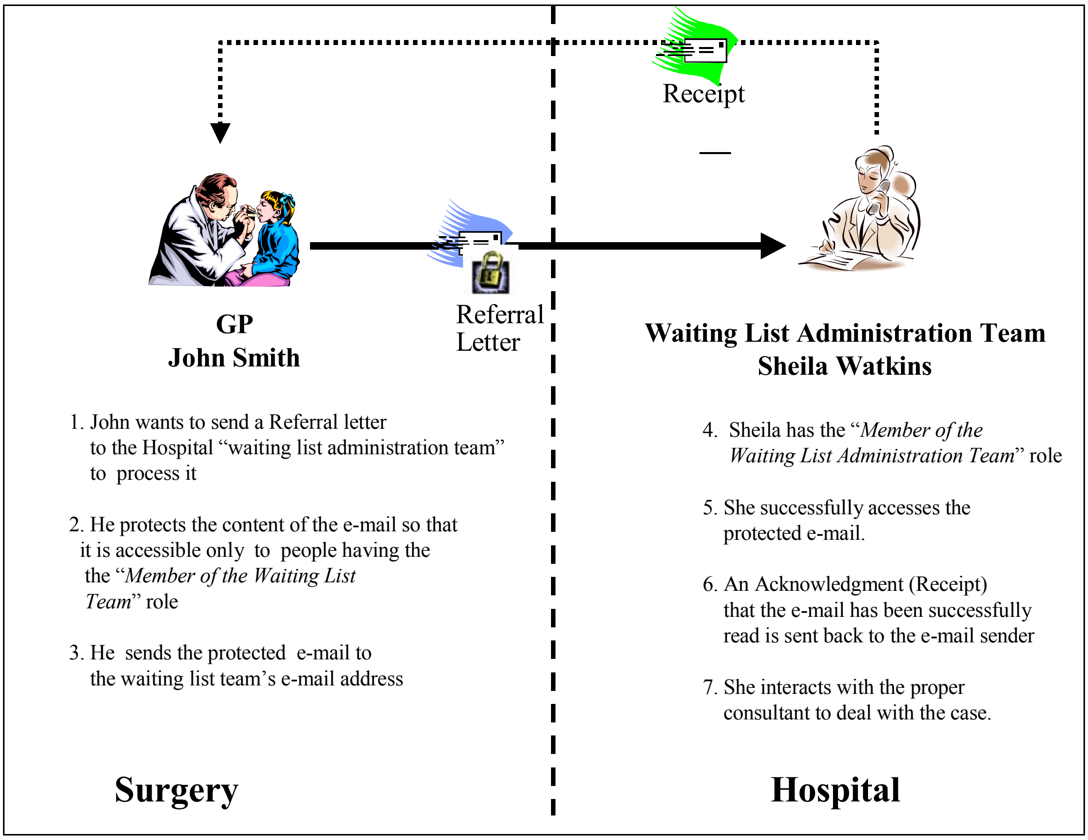
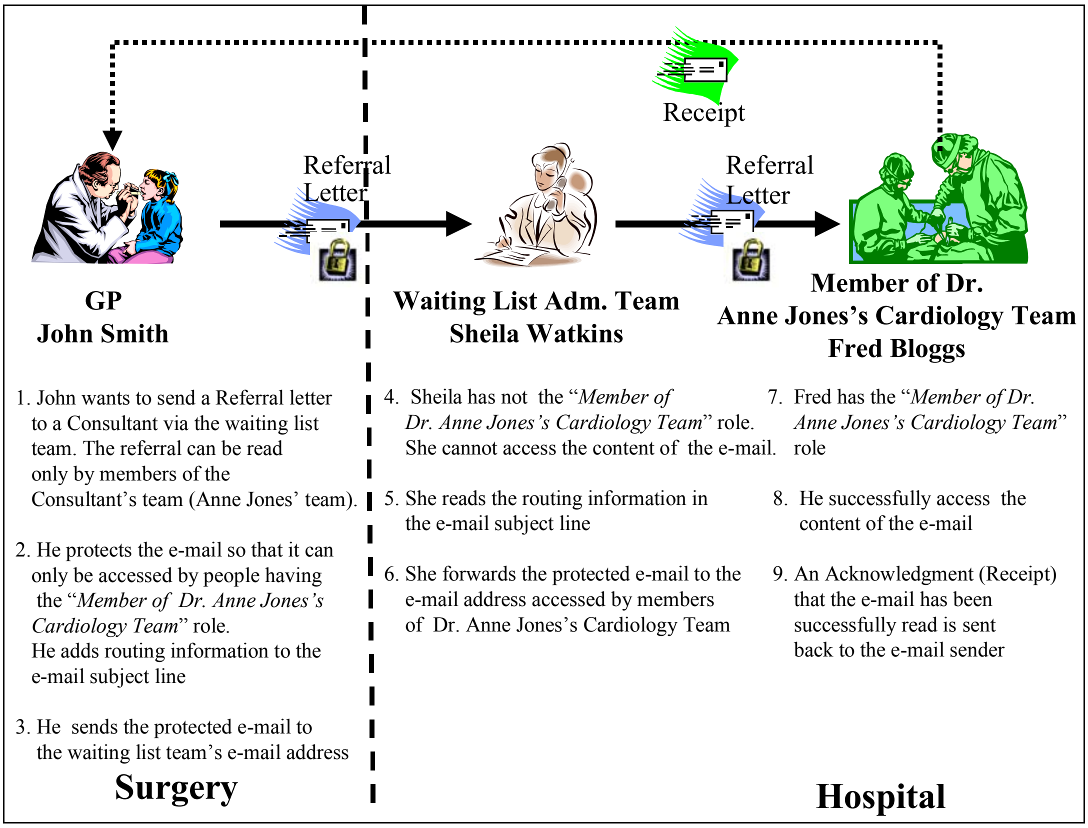

Receipt
GP
John Smith
Referral
Letter
Waiting List Administration Team
Sheila Watkins
1. John wants to send a Referral letter to the Hospital "waiting list administration team" to process it
2. He protects the content of the e-mail so that it is accessible only to people having the the "*Member of the Waiting List Team*" role
3. He sends the protected e-mail to the waiting list team's e-mail address
4. Sheila has the "*Member of the Waiting List Administration Team*" role
5. She successfully accesses the protected e-mail.
6. An Acknowledgment (Receipt) that the e-mail has been successfully read is sent back to the e-mail sender
7. She interacts with the proper consultant to deal with the case.
Surgery
Hospital

Figure 2: Interaction Scenario 1

Figure 3 shows a more restrictive scenario. The GP wants that patient's confidential information is only accessible to a specific consultant or any member of his/her team - for example "Member of Dr. Anne Jones's Cardiology Team". The GP might not know who the members of the consultant's team are, who is currently on duty or what their e-mail addresses are.

The GP sends his confidential message to the Waiting List Administration Team. The members of this team cannot access the content of this message. By using additional "routing" information (such as the e-mail subject line) they re-route the confidential message to the appropriate people. The receiver(s) will be allowed to access the confidential information only if they have, at that very point in time, the requested role, for example the "Member of Dr. Anne Jones's Cardiology Team" role.

In both scenarios the consultant that is on duty or in charge of a specific patient can change, over time. Similarly the members of a consultant's team and the members of the waiting list administration team can vary depending on personnel availability and workloads.



1. John wants to send a Referral letter to a Consultant via the waiting list team. The referral can be read only by members of the Consultant's team (Anne Jones' team).
2. He protects the e-mail so that it can only be accessed by people having the "*Member of Dr. Anne Jones's Cardiology Team*" role. He adds routing information to the e-mail subject line
3. He sends the protected e-mail to the waiting list team's e-mail address
4. Sheila has not the "*Member of Dr. Anne Jones's Cardiology Team*" role. She cannot access the content of the e-mail.
5. She reads the routing information in the e-mail subject line
6. She forwards the protected e-mail to the e-mail address accessed by members of Dr. Anne Jones's Cardiology Team
7. Fred has the "*Member of Dr. Anne Jones's Cardiology Team*" role
8. He successfully access the content of the e-mail
9. An Acknowledgment (Receipt) that the e-mail has been successfully read is sent back to the e-mail sender

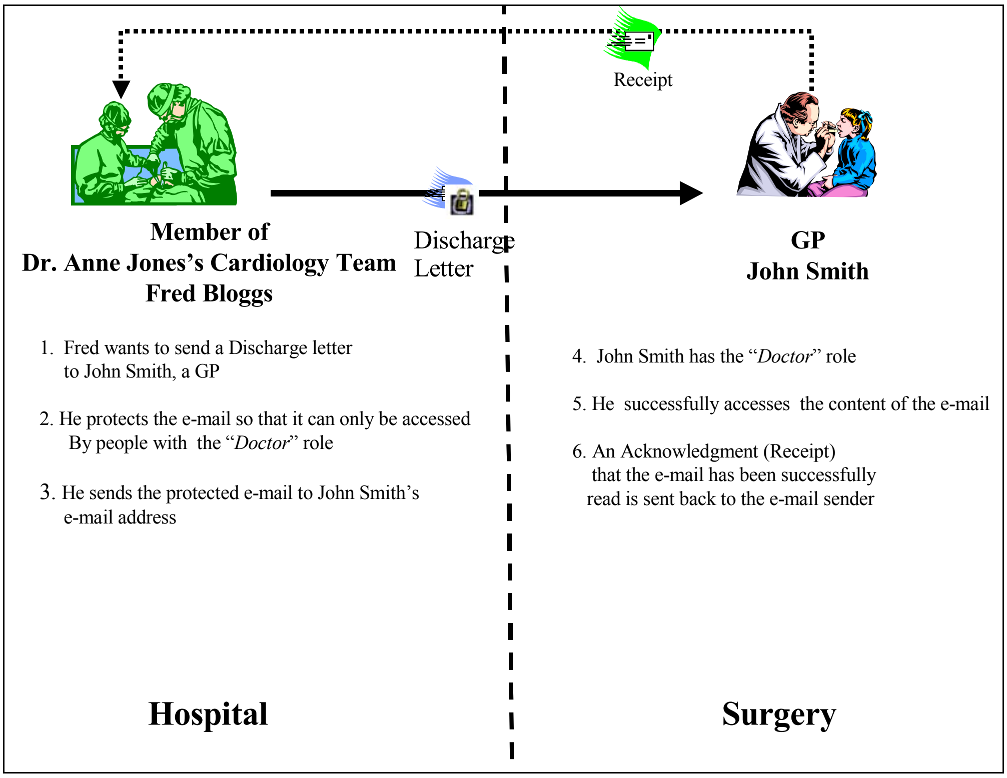
Figure 3: Interaction Scenario 2

Figure 4 and 5 shows two scenarios that are similar to the precedent ones. In this case a consultant, from the hospital, wants to send patient's confidential information (a discharge letter) back to the surgery that is in charge of the patient.

Figure 4 shows the case where the consultant sends a message directly to a GP, at a specific surgery, with the constraint that the receiver must have the "Doctor" role.

Figure 5 shows the case where the consultant decides that any member of the GP's team (at a specific surgery) can access the patient's information. He does not know specifically who is on duty. In this case the receiver must be a member of the "GP's team", in order to access the patient's information.

In both scenarios the doctor that is on duty or in charge of a specific patient can change over time. Similarly the members of a doctor's team can vary depending on personnel availability and workloads.



1. Fred wants to send a Discharge letter to John Smith, a GP

2. He protects the e-mail so that it can only be accessed By people with the "*Doctor*" role

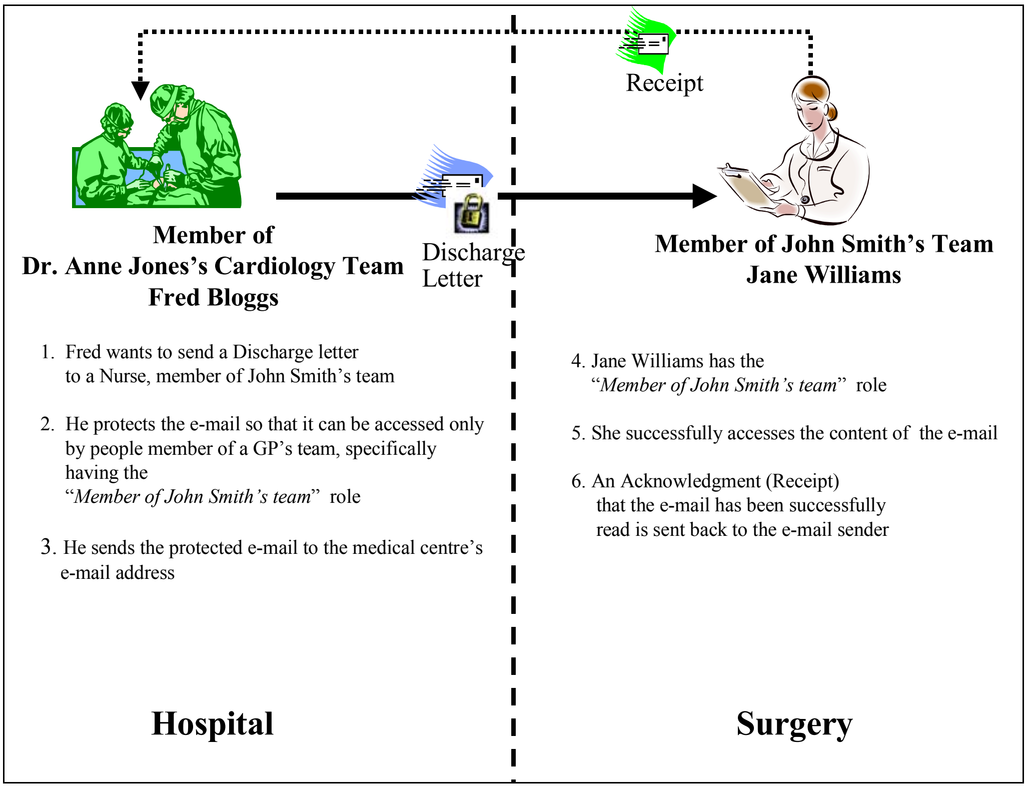
3. He sends the protected e-mail to John Smith's e-mail address

4. John Smith has the "*Doctor*" role

5. He successfully accesses the content of the e-mail

6. An Acknowledgment (Receipt) that the e-mail has been successfully read is sent back to the e-mail sender

Figure 4: Interaction Scenario 3



1. Fred wants to send a Discharge letter to a Nurse, member of John Smith's team

2. He protects the e-mail so that it can be accessed only by people member of a GP's team, specifically having the "*Member of John Smith's team*" role

3. He sends the protected e-mail to the medical centre's e-mail address

4. Jane Williams has the "*Member of John Smith's team*" role

5. She successfully accesses the content of the e-mail

6. An Acknowledgment (Receipt) that the e-mail has been successfully read is sent back to the e-mail sender

Figure 5: Interaction Scenario 4

In the highlighted scenarios, the initial sender of the secured e-mail is acknowledged when a legitimate receiver successfully reads it.

4. High Level Requirements

The above scenarios stress the need for a solution that, on one hand, ensures the privacy and confidentiality of the content of e-mails exchanged by health service employees (such as GPs, consultants, member of waiting list administration teams, etc.) and, on the other hand, copes with frequent changes of their roles and access rights.

It is important to notice that there is a neat separation between the concept of "e-mail address" and "role" of the receivers.

The e-mail service is purely a communication service. In general, confidential messages can be sent to e-mail addresses or mailing lists that are accessible by different kind of people, with different roles and rights. Only the people that have the required roles can be allowed to read the content of confidential messages.

On the other hand, the roles played by a receiver at a specific point in time are determinant to decide if that person can access a confidential message.

Confidential messages might be sent without knowing, a priori, who the final receiver (i.e. a person entitled to access their content) is. On the other hand, all the above scenarios describe situations where disclosure policies (for example a role membership) are well known and explicitly specified by the sender. These policies (constraints) need to be satisfied before access to confidential information is granted.

There is a need for a system that protects the content of confidential messages by means of flexible disclosure policies. This system must be restricted to reveal the content of confidential messages only to entities that satisfy these policies.

High-level system requirements follow:

- Privacy and confidentiality: messages need to be obfuscated by the system before being transmitted and securely stored at the receiver site, at least till a legitimate user is entitled to de-obfuscate and read them;

- Policy-based disclosure: disclosure policies need to be strictly associated to the obfuscated messages. The system must ensure that the disclosure of confidential information happens only if the associated policies (defined by the message sender) are satisfied. Disclosure policies can potentially be of any type: in this paper we specifically consider constraints based on role memberships, at a specific point on time. It must be possible to tell if policies have been tampered with;

- Strong authentication: people need to be strongly authenticated by the system. The system needs users' identities to decide if they are entitled to access obfuscated messages by retrieving their associated profiles (including their roles) and checking them against disclosure policies;

- Security: the overall system must be secure. Data need to be transmitted and stored in an obfuscated way. System components, such as authorization engines and role-based access control information need to be secured and protected;

- Flexibility: the system must allow users to flexibly specify policies to constrain the access to confidential information. In particular it must be possible to specify role-based disclosure policies. The system must allow users to obfuscate and send messages without knowing, a priori, the identity of the receiver. The system must support late-binding mechanisms for roles;

- Simplicity: the overall system must be simple to use, both for end-users (to define disclosure policies and protect messages) and system administrators.

The next section describes relevant technologies currently available on the market, a possible solution of the problem and some related problems.

5. Solution by Using Traditional Technologies

Traditional cryptographic systems [2] (based on public/private key, symmetric keys or any combination of them) and X.509 Public Key Infrastructure (PKI) [3] can be used to address the problem.

Digital certificates along with PKI infrastructures are a suitable technology to underpin authentication, non-repudiation and confidentiality. On the other hand, it is well known that PKI suffers of flexibility, scalability and certificate lifecycle management problems. The issuance, validation, verification and revocation of digital certificates are critical tasks and can introduce management burdens, especially in dynamic contexts, both for administrators and end-users.

In case of secure messaging services, digital certificates and traditional cryptography schema (based on public/private key pairs) offer a viable solution when privacy criteria depends on the “identity” of message receivers: in this case a confidential message can be encrypted by directly using the public key of the receiver.

If privacy criteria do not depend on receivers’ identities but on other aspects, such as the satisfaction of predefined disclosure policies (including membership to roles), it is not possible to use digital certificates as specified before. In this case it is not known, a priori, which digital certificate (public key) must be used for encryption purposes.

To solve the problem, a further level of indirection is required. An additional system component can to be introduced to deal with policy management interpretation and authorization. This component must be a trusted element of the system.

A digital certificate (well known by the system users) can be associated to this trusted component. It can be used to encrypt confidential information along with its disclosure policies.

Encrypted message bundles can be used to represent, transmit and store secured messages. For example, a message bundle could include two pieces of data:

- The first piece of data contains the encrypted message. Encryption can be achieved by means of a symmetric key.

- A second piece of data contains the above symmetric key along with the disclosure policies. This second piece of data is encrypted by means of the trusted component’s public key.

Enveloping techniques, such as PKCS#7 [4] or signed XML [5] document, can be used for this purpose.

Encrypted bundles can be sent by e-mail to a receiver. The receiver has to interact with the trusted component to get the symmetric key and decrypt the secured message.

The trusted component only needs to access the second piece of the bundle, once it has identified the requestor. It decrypts the associated disclosure policies and check if they are satisfied. Only at this point it returns a decryption key (symmetric key) to the requestor. Role-based access control (RBAC) [6] mechanisms (along with a role based authorization engine) can be coupled to the above component to deal with role-based disclosure policies.

It is important to notice that with this approach, disclosure policies are used as passive entities. The trusted component's public key is actually used for encryption purposes. Disclosure policies are not actively used to protect the privacy and confidentiality of messages: they are only carried around as metadata.

To build the above solution traditional secure e-mail services based on S/MIME and users digital certificates are of little help. Additional mechanisms need to be implemented to deal with the authoring of disclosure policies, the management of encrypted bundles and late binding of roles. A trusted component has to be built from scratch. Traditional trusted third parties, such as Certification Authorities and OCSP responders, only deal with digital certificates verification and trust management aspects. They do not deal with the management of disclosure policies, at the "application level".

Although it is possible to solve the secure messaging problem by building hybrid solutions that use traditional cryptography and PKI (coupled with RBAC) this is not the most natural way of using the PKI model and digital certificates.

The work described in this paper is a research and development effort to provide an alternative solution to the secure messaging problem that is simpler and more flexible.

6. Proposed Solution

The technical solution proposed in this paper leverages the Identifier-based Encryption (IBE) schema, a new emerging cryptographic schema [7], [8]. Further work is also described in [12].

The next sections describe the key IBE principles along with the details of our lightweight and flexible ***role-based encryption system.***

6.1 IBE cryptography schema

The IBE cryptography schema [7], [8] has two core properties:

- 1st **Property**: any kind of string can be used as an IBE encryption key (public key). This "string" consists of any sequence of characters or bytes such as **a role** description, a text, a name, an e-mail address, a picture, a list of terms and conditions, etc. Information is encrypted by using this string along with a "public detail", uniquely associated to a specific trusted third party, referred in this paper as *trust authority (TA)*. This trust authority is the only entity that can generate the correspondent IBE decryption key. It only relies on a local *secre*t that is a critical resource and needs to be properly protected;

- 2nd **Property**: the generation of an IBE decryption key (associated to an IBE encryption key, i.e. a string) can be postponed in time. In other words an IBE decryption key can be generated (by a trust authority) a long time after the correspondent IBE encryption key was created.

Figure 6 shows the IBE interaction model:

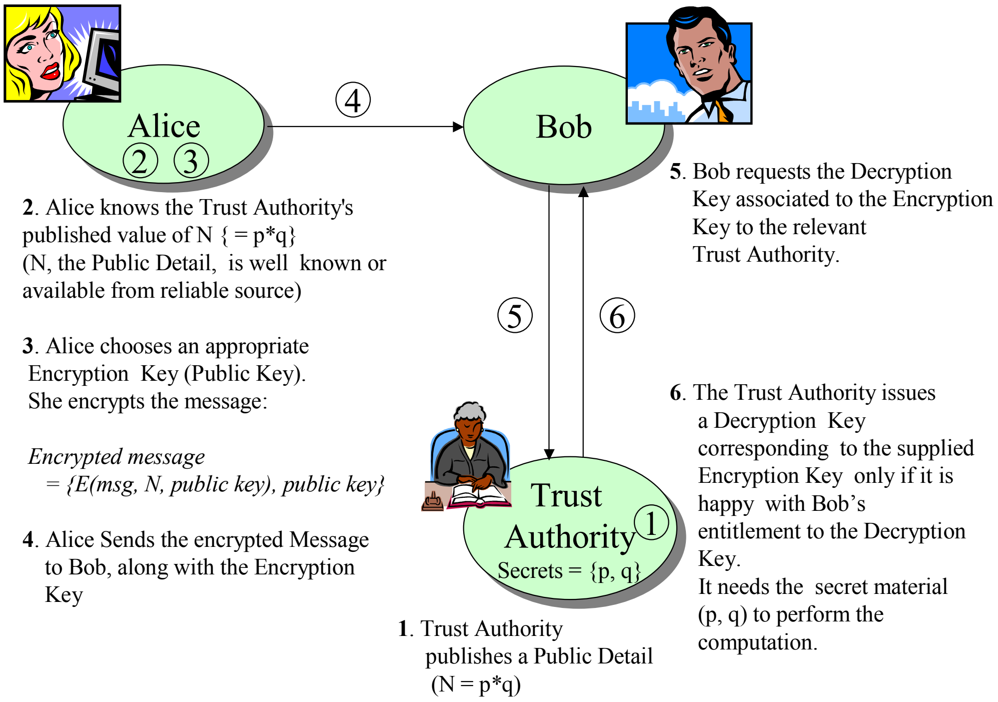

Alice
② ③
④
Bob
5. Bob requests the Decryption Key associated to the Encryption Key to the relevant Trust Authority.
2. Alice knows the Trust Authority's published value of N { = p*q} (N, the Public Detail, is well known or available from reliable source)
⑤
⑥
3. Alice chooses an appropriate Encryption Key (Public Key). She encrypts the message:
Encrypted message = {E(msg, N, public key), public key}
6. The Trust Authority issues a Decryption Key corresponding to the supplied Encryption Key only if it is happy with Bob's entitlement to the Decryption Key. It needs the secret material (p, q) to perform the computation.
Trust Authority ①
Secrets = {p, q}
4. Alice Sends the encrypted Message to Bob, along with the Encryption Key
1. Trust Authority publishes a Public Detail (N = p*q)

Figure 6: High-level IBE Interaction Model

Three players are involved in the above interaction model: a sender of an encrypted message (Alice), the receiver of the encrypted message (Bob) and a trust authority in charge of issuing decryption keys.

Alice wants to send an encrypted message to Bob. Alice and Bob trust a third party, the trust authority (TA). The following steps take place:

1. During the TA's initialisation phase, the TA generates a secret (stored and protected at the TA site) and a correspondent "public detail" that is publicly available.

2. Alice trusts the TA. She retrieves the public detail from the TA site;

3. Alice wants to send a message to Bob. She defines an appropriate IBE encryption key (public key) to encrypt this message. The IBE encryption key can be any type of string, for example Bob's role or Bob's e-mail address. Alice's message is encrypted by making use of this IBE encryption key and the TA's public detail.

4. Alice sends the encrypted message to Bob, along with the IBE encryption key she used to encrypt the message.

5. Bob needs the decryption key associated to the above IBE encryption key, to decrypt Alice's message. Bob has to interact with the trust authority. He might have to provide additional information (credentials) to prove he is the legitimate receiver of the message.

6. The trust authority generates and issues to Bob the IBE decryption key (associated to the IBE encryption key chosen by Alice) if it is satisfied by Bob's "credentials". The trust authority might decide to generate the IBE decryption key depending on the fulfilment of specific constraints as specified by the correspondent IBE encryption key. For example a trust authority might issue an IBE decryption key to Bob only if

he is compliant with a well-defined list of terms and conditions. Please notice that the IBE public key (i.e. a string), used to encrypt the document, would directly specify the list of these terms and conditions.

The IBE model fits very well to address the role-based secure messaging problem. First of all, it is possible to use the "role" of the intended e-mail receiver as an IBE encryption key (public key) and directly encrypt a confidential e-mail. Secondly, the TA can generate the correspondent IBE decryption key on the fly (when needed) if the receiver is currently playing the requested role. There is no need to share or store any secret between the sender and the receiver.

6.2 Technical Solution

6.2.1 Model

The model used for our solution derives from the IBE model. In our model confidential e-mails are directly encrypted by means of textual strings, representing IBE encryption keys. These strings explicitly describe the disclosure policies (terms and conditions) under which the content of an e-mail can be disclosed, specifically a list of roles. For example if a GP wants to send a confidential e-mail to any person that is a consultant, he/she can simply use the *"Consultant"* string to encrypt the e-mail. If a GP wants to send an e-mail that can be accessed by any member of the waiting list administration team, he/she can use the *"Member of the Waiting List Administration team"* string to encrypt the e-mail.

We do make use of a trust authority. The receiver of a confidential e-mail has to authenticate and interact with this trust authority to retrieve the appropriate decryption key.

The trust authority retrieves up-to-date lists of roles associated to users and checks them against the relevant disclosure policies. As for traditional RBAC system, in this model it is necessary to manage the associations of people's identities with their current roles.

The trust authority will generate and issue a decryption key only if the requestor has the required role(s). If the disclosure policies are tampered with, the generation of the correct decryption key is impossible.

6.2.2 Additional Technical Constraints

Our role-based secure messaging solution has been constrained by additional technical requirement expressed by a major European health care organisation. This organisation runs a trial of our solution and technology, jointly with HP Labs and a government organisation:

- People use Microsoft (MS) Outlook 2000 as e-mail browser; Microsoft Exchange Servers are used as e-mail servers;

- People are authenticated by using the *MS Windows* authentication mechanisms [9]. Each employee has a unique *MS Windows* logon and belongs to a predefined Windows trust domain;

- Multiple *MS Windows* trust domains are tactically used to reflect the structure of the health care organisation. Specifically, GPs and their assistants belong to a *GPDomain* trust domain; consultants, their collaborators and waiting list administration teams are part of the *HospitalDomain* trust domain.

- All the PCs used to exchange confidential information are part of a protected and secured organisation's Intranet.

The above constraints along with the high-level requirements influenced the design and implementation of our solution.

6.2.3 Technical Details

Figure 7 shows the high-level architecture of our solution.

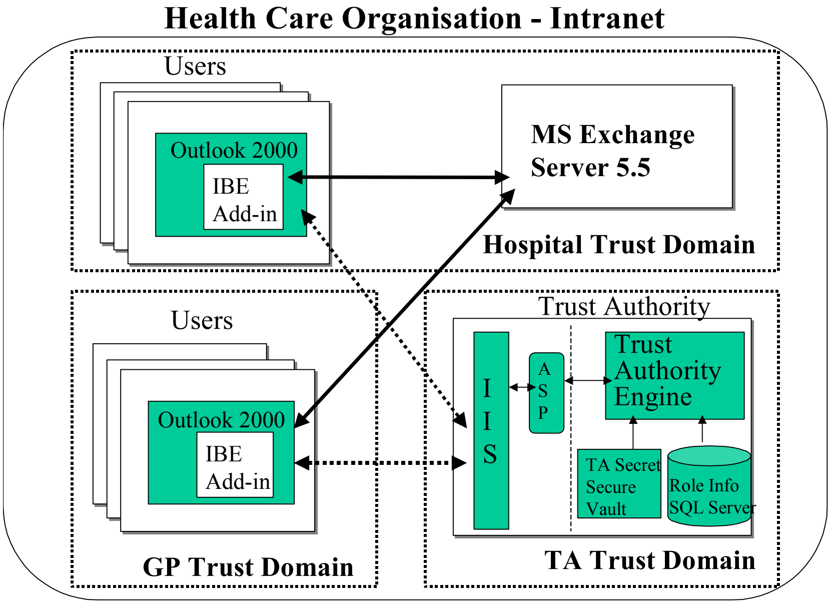


Figure 7: High level system architecture

The core architectural components are:

- ***A Microsoft Outlook 2000 Add-In***: it is a standard add-in implementing the Microsoft Office *IDTExtensibility2* interface [10]. It is deployed on users' Outlook 2000 e-mail browsers.
  It includes the following core sub-components:
  - A module providing the IBE cryptographic algorithms used to encrypt and decrypt e-mails;
  - A graphical UI to help users to easily author role-based encryption policies;
  - A secure communication module to remotely interact, via https protocol, with the trust authority in order to retrieve role lists and asks for IBE decryption keys.

- ***A Trust Authority Service***: it is a web service, hosted by a secure and protected server, and accessible via a Microsoft IIS web server. Only secure https connections are accepted from authenticated users. The web service includes:
  - A front-end wrapper (.asp scripts), to deal with remote invocation;
  - A back-end persistent trust authority engine. This engine makes authorization decisions and relies on IBE cryptographic algorithms to generate the IBE decryption keys;
  - A cryptographically secure vault used to store the trust authority secret;

  Trusted administrators run the server and the trust authority service.

- **A protected and secured Microsoft SQL Server database**, associated to the trust authority. It containing up-to-date lists of roles and up-to-date associations of people's identities to their current roles. Trusted administrators run and update the database.

In term of *privacy and confidentiality* the Outlook add-in uses IBE cryptography to encrypt confidential messages. Specifically, user's disclosure policies (list of roles) are used as IBE encryption key. The decryption key is not known a priori. The trust authority will generate it on the fly, if the requestor is entitled to it.

All the elements of a confidential e-mail are encrypted, including its subject line, body and attachments.

The Outlook add-in intercepts a confidential e-mail, before it is sent to the receiver(s) and replaces it with a new e-mail containing protected information within an attached "data bundle". This "data bundle" contains the original e-mail's subject line, body and attachments, in an encrypted format. The relevant disclosure policies are also part of the bundle. A variant of the bundle format allows also the encryption of the disclosure policies, if required. In this case a random ephemeral key is used as an IBE encryption key. For convenience and flexibility, we used XML [11] to represent the data bundle. The subject of the new e-mail is by default a generic string (such as "Encrypted e-mail"). The user can modify it, for example to contain useful "routing" information for the e-mail.

At the receiver side, the Outlook add-in manages all the interactions with the trust authority service, in order to retrieve an IBE decryption key. The user triggers this interaction by pushing a "Decrypt" button, within the e-mail window. The decryption key is transmitted from the trust authority to the Outlook add-in via a secure https connection. The Outlook add-in automatically decrypts the e-mail (by using IBE decryption algorithms) and shows all the e-mail's original confidential elements, including the original e-mail subject, body and attachments. In case of successful decryption, a receipt is sent via e-mail to the initial sender.

The trust authority shares the semantics of the disclosure policies with the Outlook add-in. After authenticating a user, it checks if the user has the required role(s) by looking at tables in an associated SQL database. Only in case of success, it generates (on-the-fly) the decryption key. The trust authority never accesses the content of confidential messages, as only the disclosure policies are sent to it.

For the solution developed for the trial, the SQL database contains two simple tables:

- A table with the list of roles relevant for specific locations (such as a surgery, a department in a hospital, etc.);
- A table with the association of users' identities (windows logon name and trust domain) and their current roles.

It is important to notice that a role can be any kind of string. For example a user's e-mail address can be used as a "role". An e-mail encrypted by using an e-mail address can be decrypted if the receiver has the e-mail address string associated as a role, in the database table.

In case of dynamic changes of people's roles the administrator has to update the content of the database tables accordingly. Note that no wider publication of these is required.

A user can virtually author any kind of *disclosure policies* and use them as an IBE encryption key to protect a confidential e-mail.

For the purpose of the trial, we limited these policies to a list of the roles that the intended receiver needs to play. A special case consists of using the *e-mail address* of the receiver as an IBE encryption key (as explained before, in this case the "e-mail address" is considered as a role and it has to be added into the database).

A simple and intuitive authoring tool has been built and integrated in Outlook: it is accessible by pushing the "Add Rules/Receiver …" button, in the e-mail editor. The authoring window allows the user to:

- Select the e-mail address of the intended e-mail receiver (from a local or Exchange Server's address book);
- Retrieve the list of significant roles for a specific location. For example, if a GP is sending an e-mail to a person working at the hospital, he/she might be interested to know what are the significant roles that can be played by people at the hospital e.g. "Consultant", "Member of the Waiting List Administration Team", "Member of Dr. Anne Jones' Team", etc.;
- Select the intended role(s) of the receiver. This is the actual disclosure policy.

The add-in automatically associates the authored disclosure policy to the e-mail and uses it for encryption, when the e-mail is actually sent.

This tool can be easily extended to allow the authoring of more complex policies.

In term of *authentication*, because of constraints dictated by the trial, we make use of *Microsoft Windows* authentication to authenticate users.

Every user has a unique *MS Windows* logon. Users authenticate to the trust authority by using the *MS Windows* authentication mechanism, via the IIS web server. This process is mediated by the Outlook add-in and it is transparent to users.

In order to make this authentication process more scalable, an ad-hoc trust domain has been explicitly associated to the trust authority. This trust domain trusts (by means of trust relationships) the *GPDomain* and the *HospitalDomain* trust domains, as shown in Figure 8:

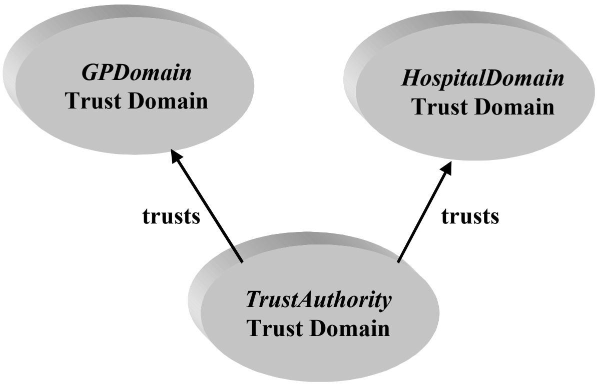


Figure 8: Trust Domains relationships

In term of *security*, encryption and secure https channels are used to exchange information. E-mails, by default, are stored in their encrypted format even if they have been successfully decrypted.

The trust authority service is run in a very secured and protected PC. This PC is a critical resource: it is located in a secluded area and accessible only by authorised administrators. It is configured to include anti-virus and a local firewall product.

In addition, the trust authority **secret** is physically stored in a pass phrase protected vault, on the PC hard disks. Copies of this secure vault are made on CD-ROM and other persistent storages, and stored in safes, as a precaution for disaster recovery.

In term of *simplicity* of usage, users can encrypt and decrypt confidential e-mails by using intuitive functionalities that are completely integrated in the Microsoft Outlook browser. System administrators only need to start the trust authority engine and provide to the system the pass phrase to access the trust authority secret.

7. Discussion

We currently have a fully working implementation of the solution. It took three months and two researchers' effort to achieve this result.

IBE cryptographic libraries (based on [8]) have been fully implemented by HP Labs and optimised to achieve cryptographical performances comparable to traditional RSA algorithms.

All the key components of the solution have been designed and implemented to satisfy both the high level requirements and the specific technical constraints. The solution has been deployed and tested within an IT test environment at HP Labs and it is currently under deployment and testing at the health service organisation.

Based on the experience and evidence we accumulated so far, we believe that the proposed solution has advantages in term of *simplicity* and *flexibility* if compared to a similar solution build with traditional cryptography schemas and PKI infrastructure.

The key point of this work is to provide flexibility in managing and enforcing disclosure policies for confidential messages, in dynamic environments. Specifically it is important to be able to encrypt confidential e-mails without having to depend on the identity of the receiver.

Our solution uses flexible disclosure policies directly as IBE encryption keys. These keys are self-explanatory as they are "the constraints" to be satisfied by secured e-mail readers. Because of the IBE properties, it has been straightforward to implement a mechanism that supports "late-bindings" of roles.

The semantic of disclosure policies can be extended in a simple way, independently by the underlying encryption algorithms. In this case, only the trust authority engine needs to be extended. This has no impact on the underlying IBE cryptography system.

We believe that our solution is *simple to manage*. No complex enveloping techniques need to be used. No public key/digital certificates need to be issued, managed and revoked, *at least* for encryption purposes. In the current solution, no secret need to be stored at users' PCs or exchanged among them. The Outlook add-in (installed at the users' sites) only needs to know what the trust authority's public detail is (necessary for encryption purposes). This can be locally stored (at the installation time of the Outlook add-in) or downloaded from the trust authority web site.

The solution deployed in the trial relies on Microsoft Windows authentication mechanisms to authenticate users. This is a quite specific approach to authentication and it simplified the way we solved the problem (although during the setting of the trial this has caused a few concerns, including the need to configure trust relationships between trust domains and configure network firewalls).

At the current state of our research we are exploring IBE-based challenge/response schemas for authentication purposes but we do not yet have evidence that they are better than traditional PKI-based authentication and they simplify users' experiences and the overall management.

In general, we believe IBE can be used as a complementary technology to traditional PKI, especially when exploiting its encryption features.

An issue of the current solution is the maintenance of the content of the SQL database tables (containing roles, and role associations). Keeping the access control information up-to-date is a well-understood RBAC problem (and more in general an access control problem). Further automation can be introduced in case this information is available from other sources, such as directory services containing up-to-date organisation data. However role definitions change less frequently than membership of groups of users that perform roles.

At the end, the trial (currently run by the health care organization) will give us valuable evidence about the validity and scalability of our solution.

8. Current and Future Work

Our secure messaging solution is currently under investigation in a trial with the health service organisation. We are monitoring its usage and problems or issues encountered by the users. The lessons we are learning in this phase will be presented and discussed in an additional HP Labs report.

In parallel, we are investigating how our solution can be extended or re-engineered in order to be used in other dynamic contexts (including government, financial and military environments) that require secure messaging services and lightweighted, policy driven encryption mechanisms. In particular we are exploring how to extend the disclosure policies to include time-based constraints, terms and condition constraints, obligation policies, etc.

We are also exploring how to extend our solution to include multiple trust authorities, run by independent authorities, in order to avoid the reliance on only one third party and at the same time, avoid the complexity of PKI Certification Authorities' hierarchies.

9. Conclusion

The access management to private and confidential digital documents is a major problem for modern dynamic organisations, especially when people frequently change roles, rights and permissions.

We focused on the problem of providing a role-based secure messaging service in a health care context, where people's functions are subject to changes and it is imperative to deal with "late-bindings" of roles.

Current technologies, such as traditional cryptography schemas and PKI, can be used to solve the problem by they suffer of flexibility and management problems. Additionally, their underlying models do not naturally fit.

We described an alternative approach to the problem that makes use of the IBE cryptography schema. IBE encryption keys are used to directly represent disclosure policies associated to confidential e-mails, including the list of the required roles. An IBE trust authority generates (on the fly) IBE decryption keys. This component, coupled with a RBAC system, satisfies, in a very simple way, the requirement for "late-binding of roles".

We believe that our approach is more flexible and simpler than an equivalent approach based on traditional cryptography and PKI technology. A few issues still need to be explored in a broader context, especially regarding the authentication of users with the trust authority.

A working secure messaging solution has been implemented and deployed within a HP Labs' IT messaging infrastructure. At the moment, it is also used in a trial with a major European health care organisation. A following report will describe the lessons we learnt and the feedback we received.

10. Acknowledgements

We would like to thank David Soldera for his very valuable work in researching, designing and implementing key elements of the secure messaging solution along with providing us with the core IBE cryptography code.

A special thank also to the RIT people at HP Labs, Bristol, for their support in setting-up and managing the IT environment we used to deploy and test our solution.

11. References

[1] The Caldicott Committee – Report on the review of patient-identifiable information. UK-http://www.doh.gov.uk/confiden/crep.htm - 1997

[2] W. Diffie, M.E. Hellman – New Directions in Cryptography - 1976

[3] R. Housley, W. Ford, W. Polk, D. Solo - RFC2459: Internet X.509 Public Key Infrastructure Certificate and CRL profile, IETF - 1999

[4] RSA Laboratories – *PKCS#7: Cryptographic Message Syntax Standard*. Version 1.5 - 1993

[5] D. Eastlake, J. Reagle, D. Solo – XML-Signature Syntax and Processing, draft-ietf-xmldsig-core-08, IETF - 2000

[6] D. Ferraiolo, R. Kuhn – *Role-based Access Control*. NIST - 1992

[7] C. Cocks - An Identity Based Encryption Scheme based on Quadratic Residues. Communications-Electronics Security Group (CESG), UK. http://www.cesg.gov.uk/technology/id-pkc/media/ciren.pdf - 2001

[8] D. Boneh, M. Franklin – Identity-based Encryption from the Weil Pairing. Crypto 2001 – 2001

[9] R. E. Smith – Authentication: From Passwords to Public keys. Chapters 10, 11. Addison Wesley - 2002

[10] D. Gifford, K. Slovak, C. Burnham – Professional Outlook 2000 Programming – Wrox - 2000

[11] W3C – Extensible Mark-up Language (XML) - http://www.w3.org/XML/ - 2003

[12] L. Chen, K. Harrison, A. Moss, D. Soldera, N.P. Smart - "Certification of Public Keys within an Identity Based System", Proc. 5th International Information Security Conference (ISC), 2002. LNCS 2433, Springer-Verlag - 2002.