Självständigt arbete på grundnivå

Independent degree project – first cycle

Datateknik GR (C), Examensarbete 15hp

Computer Engineering BA(C), Final Project 15 credits

Privacy Preserving Implementation in the E-health System

Zongzhe Chen
MID SWEDEN UNIVERSITY
the Department of Information Technology and Media (ITM)

Examiner: Ulf Jennehag, Ulf.Jennehag@miun.se
Supervisor: Tingting Zhang, Tingting.Zhang@miun.se
Author: Zongzhe Chen, jesse.chen1314@gmail.com
Degree programme: Bachelor of Science with a major in Computer Engineering, 180 credits
Main field of study: Computer Science
Semester, year: 06, 2013
Abstract

E-health systems are widely used in today’s world, and have a still brighter future with the rapid development of smart phones. A few years ago, e-health system could only be carried out on computers. But recently, people are using this as a phone application, so that they can get information at any time and anywhere. In addition, some smart phones can already measure heart rate and blood pressure, for example, “Instant Heart Rate” and “Blood Pressure Monitor”. By using these kinds of applications, users can easily measure their health data and store them in their mobile phones. However, the problem of privacy has been attracting people’s attention. After uploading their data to the database, users do have the right to protect their privacy. For instance, even the doctor has the authority to obtain the health record; the user’s name can be hidden, so that the doctor does not know who the owner of this data is. This problem also includes anonymization, pseudonymity, unlikeliness, unobservability and many other aspects.

In this thesis work, an android application is proposed to solve this problem. Users can set their own rules, and all data requests should be dealt with by calling the rules. In addition, a module in the server is to be developed to carry out the whole process of privacy preserving and the users’ data should be stored in the database.

A standard for users to set rules is determined, which is both dynamical and flexible. The application realizes some additional rule checking functions to determine whether users have set a valid rule. Privacy rules can be created, deleted, or uploaded. In addition, users can update their health record and upload it to the database. The server will call different protocols to deal with different requests, and the data which obtains the requests is responded to by calling the users’ own privacy rules.

Keywords: e-health, privacy preserving, android application, database, server, client.
Acknowledgements

Thanks to Professor Tingting Zhang who has provided me with a great deal of assistance, particularly in relation to privacy rules. At the beginning of the design part, Tingting sent me some papers which taught me some examples about setting privacy rules, which became the basis for my project. Also, thanks to Xin Zhang who has given me significant help with regards to programming so as to improve my thesis work.
# Table of Contents

Abstract ................................................................................................................. iii

Acknowledgements ............................................................................................... iv

Table of Contents .................................................................................................. v

Terminology ........................................................................................................... 8

Abbreviations ......................................................................................................... 8

1 Introduction ......................................................................................................... 1
   1.1 Background and Problem Motivation ......................................................... 1
   1.2 Overall Aim ............................................................................................... 1
   1.3 Scope ....................................................................................................... 2
   1.4 Concrete and Verifiable Goals ................................................................. 2
   1.5 Ethical Issues ........................................................................................... 3
   1.6 Outline .................................................................................................... 3

2 Related Work ....................................................................................................... 4
   2.1 XML Mediation for Privacy Anonymization .............................................. 4
   2.2 Work Done By E-health Group ................................................................. 5

3 Methodology ....................................................................................................... 7
   3.1 Selection of Privacy Policy ...................................................................... 7
Table of Contents

3.2 Selection of Software Developing Environment ........................................7
3.3 Selection of Mobile Platform .....................................................................7
3.4 Android Emulator Model ...........................................................................8
3.5 Database Model ..........................................................................................8
3.6 Evaluation Method .......................................................................................8

4 Design .............................................................................................................9
4.1 Overall Framework Design .........................................................................9
4.2 Android Application Design .......................................................................11
4.2.1 Android GUI ..........................................................................................11
4.2.2 Data Processing .......................................................................................15
4.3 Database Design ..........................................................................................16
4.4 Privacy Rule Standard Design ...................................................................17
4.4.1 Default Rules ..........................................................................................21
4.4.2 Additional Functions .............................................................................23
4.5 Server Design ...............................................................................................27
4.5.1 Protocols ..................................................................................................27

5 Result ..............................................................................................................29
5.1 Login and Other Routine Activities ............................................................29
5.2 Local Data Processing ...............................................................................30
5.2.1 Updating Local Data ..............................................................................31
5.2.2 Viewing Local data ..............................................................................32
Table of Contents

5.3 Online Data Processing ................................................................. 32
5.4 Privacy Rules Implementation .................................................... 33
5.4.1 Rule Getting and Sending ......................................................... 33
5.4.2 Rule Implementing ................................................................. 34

6 Evaluation .......................................................................................... 40
6.1 The Diversity of Privacy Rules ....................................................... 40
6.2 The Scalability of Privacy Rules ..................................................... 40

7 Conclusions ....................................................................................... 41

References ............................................................................................ 42
Terminology

Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>Identification</td>
</tr>
<tr>
<td>PC</td>
<td>Personal Computer</td>
</tr>
<tr>
<td>ECG</td>
<td>Electrocardiography</td>
</tr>
<tr>
<td>GUI</td>
<td>Graphical User Interface</td>
</tr>
<tr>
<td>UI</td>
<td>User Interface</td>
</tr>
<tr>
<td>AVD</td>
<td>Android Virtual Device</td>
</tr>
<tr>
<td>ACID</td>
<td>Atomicity, Consistency, Isolation, and Durability</td>
</tr>
<tr>
<td>SDK</td>
<td>Software Development Kit</td>
</tr>
<tr>
<td>E-health</td>
<td>Electronic Health</td>
</tr>
<tr>
<td>AES</td>
<td>Advanced Encryption Standard</td>
</tr>
</tbody>
</table>
1. **Introduction**

E-health system is being widely used at the present time and it still has a very bright future with the rapid development of smart phones. An E-health (Electronic Health) system will be implemented not only on a PC (Personal Computer) client, but also in mobile phones. By using a mobile phone, a user can easily obtain personal data anywhere and at anytime. Additionally, many phones have functions such as measuring one’s heart beat, blood pressure, and so on. Thus, users can simply store their health data into SQLite (a database used in mobile phone) after measuring their symptoms.

However, there is a problem related to privacy preserving. When the user uploads his/her data to the server’s database, he/she has the right to make it anonymous, and then those doctors who are authorized to the data cannot reveal the users’ names. It is also the case that many rules concerning privacy preserving should be carried out, which must be obeyed by the privacy protocols when attempting to obtain data from the database.

1.1 **Background and Problem Motivation**

As for the development of a health system and technology, few people would like to record their health data based on paper. With the use of the e-health system, health records can be easily shared among doctors and health service providers.

In today’s world, people are increasingly concerned about preserving their privacy. Although many people share their photos as well as other status in facebook, they will take particularly care of their health data. How is it possible to protect people’s privacy during uploading their symptoms? This is what this paper is investigating.

1.2 **Overall Aim**

The overall aim of this thesis work is to achieve privacy preserving when users are using an android application to obtain the e-health service. For
the general goals, they are: First of all, develop an android application demonstration, then, develop a module in a server which can achieve user’s privacy preserving rules.

1.3 Scope
This work has its focus on data rejection and message rejection. The system or server will reject the requests concerning the privacy if the user is not authorized.

1.4 Concrete and Verifiable Goals
Here are the concrete goals:

P1: Set up a standard for users to create their own rules.

A standard for a privacy rule, which defines the format of the rules has to be determined so that users can set their own privacy rules based on this. The rule should be extendable and easy to realize.

P2: Develop an android application demonstration

The GUI (Graphical User Interface) includes these parts:

- Login page: For users to login;
- My page: After the user logs in, it will show up for more functions;
- Personal setting page: To set some personal information, such as address.
- Privacy setting page: To conduct some privacy setting according to the privacy rules;

P3: Create a database

An SQL database will be created in the server to collect some users’ (including normal users, patients, and doctors) data. The program should be able to CRUD (Create Read Update Delete, the four basic functions of persistent storage).

P4: Evaluate
To determine the capability of the entire system, particularly the privacy rule standard, some evaluations should be conducted.

1.5 Ethical Issues
As a part of e-health system, this privacy preserving module will take care of users’ privacy by allowing users to have complete control of their data. We will inform the users that they have the right to protect their health record, allow the users to set their rules individually, and give advice in relation to improving their privacy rules so they have total knowledge in relation of those who can obtain their data.

Also, strict rules to protect users’ health data, and narrow the authority for protecting data are to be set up.

1.6 Outline
Chapter 1 provides an introduction to this thesis work. Chapter 2 describes some important theories. In Chapter 3, related methods are discussed. Then the design of privacy rules will be shown in Chapter 4. Chapter 5 shows the results of the thesis work. Finally, overall conclusions will be drawn in Chapter 6.
2. Related Work

Before the functions are able to be realized, it is necessary to view some related work about privacy preserving and electronic health. Firstly, one work regarding privacy anonymization by using xml will be investigated, after which some works of e-health system being conducted by other members of our e-health group will be introduced.

2.1 XML Mediation for Privacy Anonymization

The privacy preserving target is of such importance that many ideas regarding the implementation of rules have appeared. However, many involve a great deal of hard-coded data in order to validate privacy preserving and other functions, and this kind of application is now out of date since the xml schema has been introduced. One practical idea is to use xml to deal with rules dynamically [1], as in Figure 2.1 An Example of Xml Message.

```xml
<envelope>
  <message type="response">
    <patient id="001">
      <attendingDoctor id="002"/>
      <status>not good</status>
      <course>…</course>
    </patient>
    <persons>
      <person id="001">
        <name>Dice-K</name>
        <address>Boston, MA</address>
        <birthday>20000101</birthday>
      </person>
      <person id="003">
        <name>Ichiro</name>
      </person>
    </persons>
  </message>
</envelope>
```

Figure 2.1 An Example of Xml Message
2.2 Work Done By E-health Group

Figure 2.2 E-health System Topology

As Figure 2.2 E-health System Topology shows above, the whole system can be divided into several parts.

Jun Tang [2] has performed some monitoring applications which focus on home healthcare. The paper mainly discusses the requirements for monitoring daily activities, the methods for carrying out the whole measuring process, the results as well as the analysis, and the challenges for this testing work.

Xuchen Lu [3] deals with the middleware process which can wipe out the differences among the source data which are heterogeneous. During the communication process, all data, regardless of whether it is a temporary or permanent data flow, are enabled to pass through the middleware.

Xin Zhang [4] makes the e-health system to be distributed based on the middleware which Xuchen Lu researched. In the paper, a group of multi-thread servers are utilised in order to use a standard format to store the data. In addition, some security methods are used to protect users’ health records.
Privacy Preserving Implementation in the E-health System
Zongzhe Chen

Figure 2.3 Android Application Topology

As shown in Figure 2.3 Android Application Topology, the application group has these tasks: Linzhen Chen will develop an Android application for doctors, which can store patients’ data and perform some other operations.

Fanchen Li will conduct some research on Bluetooth communication, which can be used in the communication between patient and doctor. The mobile platform is also an Android mobile phone. Some security methods are to be carried out to protect users’ data.

Jifeng Jin will be conducting a means of sending mobile messages from doctors to patients, which provides information to the patients in relation to some routine issues in daily life.
3. **Methodology**

One important step for every project is to choose some proper methods to address the targets in the introduction part. At this point, some appropriate methods are selected to realize the goals previously discussed.

3.1 **Selection of Privacy Policy**

The privacy policy [5][6] aims to provide users with rights to protect their data, which means they can control how their health information can be used. On the one hand, users can provide data to individuals or organizations so that they can have a better health service. On the other hand, they have the right to preserve the information properly while allowing the data to be provided.

The privacy policy covers all the entities involved, including e-health users, health plans, as well as health service providers.

3.2 **Selection of Software Developing Environment**

Eclipse [7] is a multi-language software developing environment which also has many plug-ins to support various capabilities. Although it, in general, deals with the Java language, users can easily extend its abilities for various usages.

3.3 **Selection of Mobile Platform**

The Android platform [8] was chosen to be the mobile platform. As shown in Model, when the user wants to awake an Android application, he/she will call the application’s process by using the broadcast intent receiver. Then some services will start to run the application. Users can have several goals as well as UI (User Interface, used to interact with the user) in one application. In addition, the content provider takes the responsibility with regards to the sharing of data.
Figure 3.4 Android Application Model

In addition, some Android applications can already measure heart rate and blood pressure, for example, “Instant Heart Rate” [9] and “Blood Pressure Monitor” [10].

3.4 Android Emulator Model

The Android emulator [11], called AVD (Android Virtual Device), can emulate the hardware configurations and software settings, so that they can act as if they are on a real Android mobile phone.

3.5 Database Model

There are two kinds of database model, one is SQLite [12] model, and the other is the MySQL model [13]. Both are able to deal with relational database, which is based on ACID (Atomicity, Consistency, Isolation, and Durability). The former is relatively smaller so that it can be carried on the mobile phone; and the latter is larger, but has more functions.

3.6 Evaluation Method

To evaluate the standard of privacy rules, it is possible to perform some simple calculations to determine the diversity and scalability of the privacy rules.
4. **Design**

First of all, the design of the Android application will be shown, the assumption being that these are the users who are eager to view the application GUI. Then, the design of the standard of privacy rules will be displayed and finally, the server design will be conducted.

4.1 **Overall Framework Design**

As in Figure 4.5 Waterfall Model of Overall Design, this work contains three main phases: definition phase, development phase, and maintenance phase. One special part is that once the feasibility study has been performed it will not be changed because privacy preserving is not a new project to e-health, and has already been developed by many agencies.
The overall design is as Figure 4.6 Overall design, in which the rules will avoid both doctors and users from directly operating the database.

Of course, the rules will not "reject" the user’s request to his/her data, but it can store the user’s setting, and then implement them on the server. When the doctor (or health service provider) would like to obtain a user’s data, the rules will control the database, and then give the appropriate one back to the doctor.
4.2 Android Application Design

Two parts of the Android application will be introduced: one being the GUI of the application, and the other, the data processing.

4.2.1 Android GUI

An Android emulator is used to emulate the application in an Android mobile phone. Figure 4.8(a) is the screenshot of the entire Android emulator; however, only the screenshot of the key part of the emulator will be used in the remainder of the report so as to emphasize the key point and to save space.
Figure 4.8 (a) Screenshot of the whole Android emulator; (b) Login Page; (c) My Page
First of all, the user will enter the “login page” to login (Figure 4.8(b)). After entering the username and password, the application will connect the server to check the client’s username and password. Then the user enters “my page” to choose a service (Figure 4.8(c)).

![Image](image-url)

(a) (b)

Figure 4.9 (a) Deal with Data Page; (b) View Data Page

If the user chooses the “Personal Information” button, the application will jump into “Deal with Data Page” (shown in Figure 4.9(a)). First is an editable text, so that one item can be checked (by entering the item number and pressing “Get info” button) and also its value can be changed, which will be set in the “Value” editable text (by pressing “Update” button). Additionally, the entire user’s data can be viewed by pressing the “View” button. In addition the user can choose “Upload” to upload data to the server and is able to type in a user’s ID to attempt to obtain that user’s data. When the user attempts to view data, the application will jump into the “View Data Page” (shown in Figure 4.9(b)).
Privacy Preserving Implementation in the E-health System

Design

Zongzhe Chen

2013-06-29

Figure 4.10 (a) Privacy Setting Page; (b) View Rule Page; (c) Android GUI for Setting Rules
If the user chooses “Privacy Setting” button in “My Page”, the application will jump into “Privacy Setting Page” for the user to set his/her own privacy rules (shown in Figure 4.10(a)). All the items of spinners (shown in Figure 4.10(c)) are discussed in chapter 4.4. In this case, if the user clicks the “Save” button, then the current value will be stored as a new privacy rule. If the user wants to view all his/her privacy rules, then he/she can simply press the “View” button, and the application will jump into the “View Rule Page” (shown in Figure 4.10(b)). When the user clicks the “Back” button, he/she will return to “My Page”.

4.2.2 Data Processing

When the application starts, it will require the user to login, so that the program can check the user’s ID. After that, the user can deal with data locally or online (measuring data, searching hospitals, and connecting doctors face to face are not this thesis’s work).

As shown in Figure 4.11, health data process and privacy rule process are almost identical, as they both contain these three functions; getting data from the database, viewing current data, and uploading data to the database. However, they also have some individual functions, in that a user can select a certain health data item to check and is also able to update it by entering a new value. For privacy rules, instead of “updating”, the user can create a new one or clear the entire rules.
4.3 Database Design

The items in database are defined as Table 4.1 Database Items:

Table 4.1 Database Items

<table>
<thead>
<tr>
<th>No.</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>PersonalID</td>
<td>Name</td>
<td>Gender</td>
<td>Birthday</td>
</tr>
<tr>
<td>Type</td>
<td>VarChar</td>
<td>VarChar</td>
<td>VarChar</td>
<td>VarChar</td>
</tr>
<tr>
<td>Length</td>
<td>12</td>
<td>21</td>
<td>6</td>
<td>8</td>
</tr>
<tr>
<td>No.</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>Name</td>
<td>Address</td>
<td>Status</td>
<td>Profession</td>
<td>SBP</td>
</tr>
<tr>
<td>Type</td>
<td>VarChar</td>
<td>VarChar</td>
<td>VarChar</td>
<td>Int</td>
</tr>
<tr>
<td>Length</td>
<td>80</td>
<td>15</td>
<td>50</td>
<td>3</td>
</tr>
<tr>
<td>No.</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
</tr>
<tr>
<td>Name</td>
<td>DBP</td>
<td>BloodSugar</td>
<td>HeartBeat</td>
<td>DoctorList</td>
</tr>
<tr>
<td>Type</td>
<td>Int</td>
<td>Float</td>
<td>Int</td>
<td>VarChar</td>
</tr>
<tr>
<td>Length</td>
<td>3</td>
<td>2+1</td>
<td>3</td>
<td>100</td>
</tr>
<tr>
<td>No.</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
</tr>
<tr>
<td>Name</td>
<td>SubmitTime</td>
<td>VisitDay</td>
<td>VisitCount</td>
<td>PrivacyLevel</td>
</tr>
<tr>
<td>Type</td>
<td>VarChar</td>
<td>Int</td>
<td>Int</td>
<td>VarChar</td>
</tr>
<tr>
<td>Length</td>
<td>8</td>
<td>3</td>
<td>3</td>
<td>255</td>
</tr>
<tr>
<td>No.</td>
<td>17</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>PrivacyRule</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Type</td>
<td>VarChar</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Length</td>
<td>255</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(1) Personal Id: user’s ID, which is the primary key. When the user starts to use the Android application, the program will remember the personal ID of the user.

(2) Name: user’s name, which cannot be edited by the user.

(3) Gender: user’s gender, which can not be edited by the user.

(4) Birthday: user’s birthday, which can not be edited by the user. The system can use this to calculate the user’s age.

(5) Address: user’s address, which can be edited by the user.
Privacy Preserving Implementation in the E-health System  
Zongzhe Chen  
2013-06-29

(6) Status: user’s status (“user”, “doctor”, etc.), which cannot be edited by the user but can be edited by the health service provider.

(7) Profession: user’s profession, which cannot be edited by the user but can be edited by the hospital or government.

(8) SBP: The systolic blood pressure, which can be edited by the user and the doctor.

(9) DBP: The diastolic blood pressure, which can be edited by the user and the doctor.

(10) BloodSugar: blood sugar, which can be edited by the user and the doctor.

(11) HeartBeat: heart beat, which can be edited by the user and the doctor.

(12) DoctorList: doctor list, which contains all the IDs of the doctors who are the user’s doctors, and this can be edited by the user.

(13) SubmitTime: the submit time of this update process.

(14) VisitDay: the upper limit of days to obtain the user’s data, which can be edited by the user. Others cannot obtain this user’s data if the value is 0. However, when the value is 366, it means no limit for visiting days.

(15) VisitCount: the upper limit of counts to obtain the user’s data, which can be edited by the user. Others cannot obtain this user’s data if the value is 0. However, when the value is 100, it means no limit for visiting days.

(16) PrivacyLevel: The privacy level (“private”, “public”, etc.) of the whole data, which can only be decided by the user.

(17) PrivacyRule: The privacy rules which are set by the users. All the items are in a shortened form in order to save space.

4.4 Privacy Rule Standard Design

The aim of the system is to allow the users to set their own privacy rules, however, when this is not the case then the system will carry out some default rules such as in Figure 4.12 Server Judging process.
The privacy rules can be described as follows: [1]

R: Con \rightarrow Op
Con: BC \mid (Con \ AND \ Con) \mid (Con \ OR \ Con) \mid (NOT \ Con)
BC: Jnull(Para) \mid Jop(Para, Para) \mid Jop(MVE, Para) \mid Jop(Para, MVE) \mid Jop(MVE, MVE) \mid Jin(Para, Pset)
Jnull: Boolean Function composed of a term; is null \mid ...
Jop: Conditional operation composed of two terms; = \mid < \mid > \mid \leq \mid \geq \mid \neq \mid ...
Jin: Function that checks a data is included in a set of another data
MVE: Multivariable Expression; MVE(Para1, Para2, ..., Paran)
Para: Conceptual notation of three types of single data; PCst \mid PMsg \mid POut
PCst: Constants
PMsg: Conceptual notation of parts of the message
POut: Conceptual notation for data outside of the message
Pset: Conceptual notation of a set of data
Op: Simple Operation: Reject a message Rej | Anonymize a conceptual notation Ano (PMsg) | Reply a message Rep (PMsg) | …

The point is that the users can create their own privacy rules, and their rules are in the format shown below:

If (target) (property) (judge) (value), then (action).

The target includes “doctor’s” and “my”.

The doctor’s property includes "name", "profession", and "address"; and user’s property includes "SBP", "DBP", "blood sugar", "heart beat", "visit day", "visit count", and "privacy level".

The “judge” includes “is”, “is not”, “no greater than”, “no lesser than”.

“Value” part is an editable textbox which is decided by the users.

And the “action” includes “reject”, “reply”, “anonymize”, and “check others”, which will operate if the statement is fulfilled.

![Privacy Rule Syntax](image)

Figure 4.13 Privacy Rule Syntax

As shown in Figure 4.13, there are five parts to the privacy rule; target, property, judgment, value, and action. Each item will have several choices for the user. If the user makes a choice choose it will be in the form of a
sentence such as” If doctor’s profession is not Doctor, then reject”. What should be noticed is that the value part can be set by the user.

However, some additional checks should be carried out. The first of these is the syntax check, which points out whether the property, judge, and value match and the second is the authority check, which points out whether the users have the authority to set some particular privacy rules.

All the items will be compressed into some special characters as shown below in table 4.2 in order to save space.

Table 4.2 Item names and their shortened form

<table>
<thead>
<tr>
<th>map_target</th>
<th>t_num</th>
<th>t_name</th>
</tr>
</thead>
<tbody>
<tr>
<td>map_property</td>
<td>p_num</td>
<td>p_name</td>
</tr>
<tr>
<td></td>
<td></td>
<td>p_num</td>
</tr>
<tr>
<td>map_judge</td>
<td>j_num</td>
<td>j_name</td>
</tr>
<tr>
<td>map_action</td>
<td>a_num</td>
<td>a_name</td>
</tr>
</tbody>
</table>

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>t_num</td>
<td>doc</td>
<td>m</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t_name</td>
<td>doctor’s</td>
<td>my</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>p_num</td>
<td>nm</td>
<td>pro</td>
<td>add</td>
<td>sbp</td>
<td></td>
</tr>
<tr>
<td>p_name</td>
<td>name</td>
<td>profession</td>
<td>address</td>
<td>SBP</td>
<td></td>
</tr>
<tr>
<td>p_num</td>
<td>dbp</td>
<td>bs</td>
<td>hb</td>
<td>vd</td>
<td></td>
</tr>
<tr>
<td>p_name</td>
<td>DBP</td>
<td>blood</td>
<td>heart beat</td>
<td>visit</td>
<td></td>
</tr>
<tr>
<td>p_num</td>
<td>vc</td>
<td>pl</td>
<td></td>
<td>visit</td>
<td></td>
</tr>
<tr>
<td>p_name</td>
<td>visit</td>
<td>privacy</td>
<td>count</td>
<td></td>
<td></td>
</tr>
<tr>
<td>j_num</td>
<td>is</td>
<td>not</td>
<td>ngt</td>
<td>nlt</td>
<td></td>
</tr>
<tr>
<td>j_name</td>
<td>is</td>
<td>is not</td>
<td>no greater</td>
<td>no lesser</td>
<td></td>
</tr>
<tr>
<td>a_num</td>
<td>rej</td>
<td>rep</td>
<td>ano</td>
<td>check</td>
<td></td>
</tr>
<tr>
<td>a_name</td>
<td>reject</td>
<td>reply</td>
<td>anonymize</td>
<td>others</td>
<td></td>
</tr>
</tbody>
</table>
In addition, “|” is used as a block to distinguish different items, and “###” to distinguish different rules. For example, rule “If doctor’s profession is not ‘Doctor’, then reject.” will be compressed into “doc|pro|is|Doctor|rej###”. “m|vd|ngt|0|rej###” which means “If my visitday is not greater than 0, then reject”. Additionally, for the combinations, such as for the two rules just set, this will be “doc|pro|is|Doctor|rej### m|vd|ngt|0|rej###”.

4.4.1 Default Rules

In case the users do not set their own privacy rules, there are six default rules for use.

(r1) Rule written in natural language: “The user’s personal information has to be anonymized when the user is not a patient of the doctor or the health service provider.”

Constraint condition: NOT (Jin (doctor.PersonalId, user.doctor)) → Ano (user(x).personalId), where Jin is the judgment about “exist in”, and user(x).personalId is a Para that identifies a patient’s personal information.

Implementation: When the doctor wants to obtain the data of a user, the server will check the user’s doctor list. If the doctor’s personal ID (Identification) is not on the list, then the user’s personal ID should be anonymized.

(r2) Rule written in natural language: “The message has to be rejected when the person attending to the patient is not actually ‘Doctor’ or ‘Provider’.”

Constraint condition: NOT (Jop (user.Profession, (“Doctor” OR “Provider”))) → Rej, where Jop is the “=” judgment which point out whether the user’s profession is a doctor (or a provider) or not.

Implementation: If the user wants to obtain another user’s data, the server will check his profession. The request message should be rejected if the user’s profession is not doctor (or provider).
(r3) Rule written in natural language: “A public doctor can only obtain data in his field. For example, a “cardiologist” can only obtain data about heart disease.”

Constraint condition: \( Jop (\text{user.Profession}, \text{“Heart disease”}) \rightarrow \text{Rep} (\text{patient.heartbeat}) \mid Jop (\text{user.Profession}, \text{“Diabetes”}) \rightarrow \text{Rep} (\text{patient.blood sugar}) \mid Jop (\text{user.Profession}, \text{“Hypertension”}) \rightarrow \text{Rep} (\text{patient.blood pressure}) \), where \( Jop \) is the “=” judgment.

Implementation: If the user wants to obtain another user’s data, the server will check his profession. The reply message should only be concerned with the user’s profession.

(r4) Rule written in natural language: “When the user doesn’t set his data available to an unknown doctor, the doctor cannot obtain the user’s data.”

Constraint condition: \( (Jop (\text{user.PrivateLevel}, \text{“public”})) \land (\text{Jin(doctor.PersonalID, user.DoctorList)}) \rightarrow \text{Rej} \), where \( Jop \) is a “≠” judgment; user.PrivateLevel means the privacy level of his data; and user.DoctorList includes a list which shows all the doctors authorized.

Implementation: If the user wants to obtain another user’s data, the server will check the patient’s doctor list. The request should be rejected if the user’s personal ID is not included in the list.

(r5) Rule written in natural language: “When the user wants to obtain his own data, the server should approve it.”

Constraint condition: \( Jop (\text{user.PersonalID}, \text{req.PersonalID}) \rightarrow \text{Rep} \), where \( Jop \) is a “=” judgment.

Implementation: If the user wants to obtain his/her own data, the server will check his/her personal ID and then give a reply.

(r6) Rule written in natural language: “The request message should be rejected if the user sets an upper limit of the times to visit his/her data, and the time is equal to zero.”

Constraint condition: \( (Jop (\text{user.VisitTime}, \text{“0”})) \lor (Jop (\text{user.VisitCount}, \text{“0”})) \rightarrow \text{Rej} \), where \( Jop \) is a “≤” judgment; user.VisitTime means the
Privacy Preserving Implementation in the E-health System

Zongzhe Chen

2013-06-29

period limit (based on day and is an integer) for visiting his/her data; and user.VisitCount means the time limit (is an integer) for visiting his/her data. (Note: When the VisitTime = “366” and VisitCount = “100”, it means the data has no upper time limit for visiting)

Implementation: If the user wants to obtain another user’s data, the server will check the patient’s time limit. The message should be rejected if the time limit equals to “0”.

4.4.2 Additional Functions

In order to assist users to set their privacy rules properly, some additional functions, including “Syntax check” and “Authority check” are available.

Syntax check: As is the case for its name, syntax can check one rule’s syntax part. It has two aspects, one checking whether there is an unreachable comparison, for example, in Figure 4.14, for “name”, “address”, as well as “profession”, these items cannot make comparisons, that is, one “name” is never said to be “bigger” than another.

If (target) (property) (judge) (value), then (action).

If doctor’s name not greater than 60, then reject.

Figure 4.14 Example1 of Syntax Error
When a user is attempting to set item “name” and “no greater than”, he/she is attempting to make comparisons on “name” (as in Figure 4.15(a)), and the system will inform him/her “Syntax Error. Only numerical items can make comparison, please check again.” (Like Figure 4.15(b)) The second part of the syntax check is in checking the type of value types in by the users. In this system, users are free to type what they like, but when involving comparisons, only numerical values can be involved as in Figure 4.16.

```
If (target) (property) (judge) (value), then (action).
If my SBP not greater than abc, then reject.
```

Figure 4.16 Example2 of Syntax Error
Privacy Preserving Implementation in the E-health System

Zongzhe Chen
2013-06-29

Figure 4.17 (a) Syntax Checking Example2; (b) Syntax Checking Result2

As shown in Figure 4.17(a), when the user is attempting to set the value of SBP, he/she types in the character “abc”, which, of course, is not numerical. Thus, as is the case in Figure 4.17(b), the system will give another warning “Syntax Error. The value is not numerical, please check again.”

Authority check: As shown in Figure 4.18, due to the various properties which the users can set, some are for users, while others are for the doctors. This function will check whether the user has the authority to set this kind of rule.

Figure 4.18 Example of Authority Error

Figure 4.19 shows that a user is attempting to set privacy about a doctor’s SBP value (figure (a)). However, a user does not have the authority to set rules about a doctor’s SBP value. Thus, in this case, the system will inform him “Authority Error. You have no authority to check the doctor’s SBP, so please check again.” (Figure (b))
Figure 4.19 (a) Authority Check; (b) Authority Check Result Example

Figure 4.16 shows the different parts for setting privacy rules. When the user chooses “doctor’s” as a target, then it is only possible to choose “name”, “address” or “profession” as its property. When the user chooses “my” as the target, then it becomes possible to choose the remaining properties, which are “SBP”, “DBP”, “blood sugar”, “heart beat”, “visit day”, “visit count”, and “privacy level”. In this manner there are $1 \times 3 \times 2 \times 4 + 1 \times 6 \times 4 \times 4 + 1 \times 1 \times 2 \times 4 = 128$ rules as a whole.

Figure 4.20 Authorities for Setting Rules
4.5 Server Design

The server has three main parts; the first is for connecting clients, the second is for calling protocols and the last one is for dealing with the database, which includes inserting, updating, deleting and querying.

![Server Processing Flow Chart](image)

Figure 4.21 Server Processing Flow Chart

As shown in Figure 4.21, after the server starts, it will keep waiting for clients’ connecting requests. When the connection succeeds, the server will listen to what kind of service the client requests and then it can call the correct protocol. Different protocols have different usages. After the user or client logs off, the server waits for others to connect again.

4.5.1 Protocols

There are different protocols for different usages, as in table 4.3:

Table 4.3 Protocols and usages

<table>
<thead>
<tr>
<th>No.</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>User login. This protocol is used for login. When a user attempts to login, the application will send this code to the server, and then the server can check whether this user exits. In addition, both the application and</td>
</tr>
</tbody>
</table>
the server will remember the user’s personal ID, for future usage.

2 Uploading data.
This protocol is used for uploading data to the server, including health data and privacy rules. The server can pick up keywords and parse them, then update them in the database.

3 Querying data.
This protocol is used for querying the user’s data. Because the system will remember the user’s personal ID when he/she logs in, this step does not require so many verifications and is mainly for users to update their data when they are using a new mobile phone.

4 Getting privacy rules.
This protocol is used to obtain privacy rules when users enter the privacy setting page. Because rules would not be stored on the mobile phone, users must obtain it from the server each time they want to set rules.

5 Querying other users’ data.
This protocol is for doctors to obtain other users’ health data. When the doctor wants to receive another user’s data, he/she should type in that user’s personal ID. Then a set of verifications will be brought out in order to determine whether the doctor has the authority.

6 Updating privacy rules.
This protocol is used for users to update their privacy rules. As stated, the privacy rules will not be stored on the phone, so all the rules the users set will be uploaded to the server when the users jump out.
5. **Result**

This chapter will show the results of the whole privacy preserving process.

The server will run at first, followed by the Android application. In this way, the server can wait for the application to send requests.

5.1 **Login and Other Routine Activities**

When the user opens the application, he/she is required to login first. In this way, the system can remember the personal ID of the user, and this will improve the efficiency of the following operation.

User Login Figure 5.22 below is a screenshot of the moment that a user logs in successfully. The items are described as below:

1: The personal ID the user types in;

2: The secret code;

3: The system shows “Server connected!” when it connects successfully;

4: When the server starts, it prints “server start!”;

5: The protocol for login, whose code is “100”;

6: The server will check the user’s identity, and then prints the user’s personal ID.


5.2 Local Data Processing

When the user enters the data processing page, the system will connect the server for data updating, which downloads the data from the server into the mobile phone. In this way, the user can deal with data on the phone.
5.2.1 Updating Local Data

As in Figure 5.23 shown above, data updating mainly involves six items, and these are described below:

1: User enters in the item number he/she wants to update;

2: After clicking the button “Get info”, the item name as well as its value will show on the editable text.

3: Item name, which can be edited by the user;

4: Item value, which can also be edited;

5: If the user finishes modifying the item, he/she can click “Update” to finish the update process;

6: The system will show “Update success!” as a response.
5.2.2 Viewing Local data

When the user enters the data-processing page, he/she can choose to deal with data locally or online. At this point, users are not allowed to create or delete an item to avoid them making strange changes deliberately or none deliberately (as in Figure 5.24(a)). However, if they want to store their data into an empty database, they can merely click “update” so that both the local database as well as the online database will change to that with the latest “submit time”. After that, it is possible to click the “View” button to see their local data (like Figure 5.24(b)).

5.3 Online Data Processing

<table>
<thead>
<tr>
<th>PersonId</th>
<th>Name</th>
<th>Gender</th>
<th>Birthday</th>
<th>Address</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>19640908T528</td>
<td>Emily</td>
<td>Female</td>
<td>19640908</td>
<td>Sundsvall</td>
<td>User</td>
</tr>
<tr>
<td>19730305K319</td>
<td>Vivian</td>
<td>Female</td>
<td>19730305</td>
<td>Shanghai</td>
<td>User</td>
</tr>
<tr>
<td>19800802T138</td>
<td>Cecilia</td>
<td>Female</td>
<td>19800802</td>
<td>Stockholm</td>
<td>User</td>
</tr>
<tr>
<td>19810101T134</td>
<td>Jesse</td>
<td>Male</td>
<td>19810101</td>
<td>Jiangsu</td>
<td>Patient</td>
</tr>
<tr>
<td>19831103T253</td>
<td>Corneli</td>
<td>Male</td>
<td>19831103</td>
<td>&lt;NULL&gt;</td>
<td>&lt;NULL&gt;</td>
</tr>
<tr>
<td>19910304F341</td>
<td>BiBei</td>
<td>Male</td>
<td>19910304</td>
<td>Sundsvall</td>
<td>User</td>
</tr>
</tbody>
</table>
In order to test this project, some pseudo-information was created in the database based on the design part in chapter 4 (shown in Figure 5.25). When the user wants to obtain information from the server, it will firstly check for privacy rules (which are discussed later), then check the database and collect the required information. Finally, the server will send that information in a plain character stream.

5.4 Privacy Rules Implementation

The privacy rule implementation contains two parts; one being the rule setting by the user, including creating, deleting, and uploading while the other is rule judging by the system, including replying, rejecting, and checking others.

5.4.1 Rule Getting and Sending

As long as a user enters into the “Privacy Setting Page”, the Android application will connect the server in order to obtain this user’s privacy rules, which are stored in the server. This means all privacy rules would not be saved in a local memory, but, online. In addition, when the user returns back by clicking “Back” button, the application will connect the server again to update the user’s privacy rules.
When the user enters the privacy setting page, the server will provide four responses in Figure 5.26 above, namely:

1: User’s personal ID;

2: Which personal ID the user is querying, which in this case is the actual user as he/she is obtaining his/her privacy rules which are stored on the server;

3: Querying result, which in this case is the privacy rules;

4: The data stream, the server sends back to the mobile phone.

5.4.2 Rule Implementing

The assumption is that there are two users, and their health records are as table 5.1 below:
Privacy Preserving Implementation in the E-health System
Zongzhe Chen
2013-06-29

Table 5.4 User Data

<table>
<thead>
<tr>
<th></th>
<th>User 1</th>
<th>User 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Personal ID</td>
<td>19800802T138</td>
<td>19810101T134</td>
</tr>
<tr>
<td>Name</td>
<td>Cecilia</td>
<td>Jesse</td>
</tr>
<tr>
<td>Gender</td>
<td>Female</td>
<td>Male</td>
</tr>
<tr>
<td>Birthday</td>
<td>19800802</td>
<td>19810101</td>
</tr>
<tr>
<td>Address</td>
<td>Stocholm</td>
<td>Jiangsu</td>
</tr>
<tr>
<td>Status</td>
<td>User</td>
<td>Patient</td>
</tr>
<tr>
<td>Profession</td>
<td>User</td>
<td>User</td>
</tr>
<tr>
<td>SBP</td>
<td>&lt;Null&gt;</td>
<td>80</td>
</tr>
<tr>
<td>DBP</td>
<td>&lt;Null&gt;</td>
<td>100</td>
</tr>
<tr>
<td>BloodSugar</td>
<td>&lt;Null&gt;</td>
<td>5.0</td>
</tr>
<tr>
<td>HeartBeat</td>
<td>&lt;Null&gt;</td>
<td>60</td>
</tr>
<tr>
<td>DoctorList</td>
<td>&lt;Null&gt;</td>
<td>19800802T138|19730305K319|19831103T253</td>
</tr>
<tr>
<td>SubmitTime</td>
<td>&lt;Null&gt;</td>
<td>20130412</td>
</tr>
<tr>
<td>VisitDay</td>
<td>&lt;Null&gt;</td>
<td>5</td>
</tr>
<tr>
<td>VisitCount</td>
<td>&lt;Null&gt;</td>
<td>6</td>
</tr>
<tr>
<td>PrivacyLevel</td>
<td>&lt;Null&gt;</td>
<td>Private</td>
</tr>
<tr>
<td>PrivacyRule</td>
<td>doc</td>
<td>pro</td>
</tr>
<tr>
<td></td>
<td>m</td>
<td>vd</td>
</tr>
</tbody>
</table>

In this test, user 1 will be allowed to attempt to obtain the data of user 2.
Figure 5.27 Privacy Rules on the Server Figure 5.27 above shows the rules stored on the server; and it can be seen that all the items are compressed into the keywords. In addition, two kinds of blocks are taken into use. The former is “###”, which is used for blocking two different rules and the latter one is “|”, which is used for blocking different items in one rule. As each rule will take 25 characters, a 255-character large field can have approximately 10 rules.

Figure 5.28 Viewing Rules on Mobile Phone

However, as Figure 5.28 shows, after users receive the rules from the server, the Android application will transfer the key words into normal words and then make them into a sentence, which can easily be read by the users. When the user1 sends the request for user2’s data, the system will react as shown below, like Figure 5.29 Rule Judging Process:
According to the data shown in Table 5.4 User Data, it can be seen that the privacy rules of user2 are 
```
doc|pro|not|Doctor|rej|###m|vd|ngt|0|co|###m|vc|ngt|0|co|###m|vc|ngt|0|co|###
```
which will be described in natural language in Table 5.2.

Table 5.5 Rules in Natural Language

| Rule 1 | If doctor’s profession is not ‘Doctor’, then reject. |
| Rule 2 | If my visit day is not grater than 0, then check others. |
| Rule 3 | If my visit count is not grater than 0, then check others. |

Additionally, it can be noted that the profession of user1 is actually “user” instead of “doctor” and thus access to user2’s data can be denied. The system’s response is as below:
Figure 5.30 System’s Response When Rule Not Fulfilled

As Figure 5.30 shows, when the system rejects the user’s request, it will say:” Get data failed. Due to the user’s privacy rules, you are unable to get data.”

However, this time, for the second test, the profession of user1 is changed into “Doctor Hypertension” and the server’s response is then:

Figure 5.31 System’s Response When Rule Fulfilled
On this occasion, all the privacy rules are fulfilled and thus the doctor can obtain the data, as in Figure 5.31.

Again, for the third test, the profession of the doctor is changed into “Doctor Heart Disease” and at the same time, all the privacy rules of user2 are deleted. On this occasion, the system will call the six default rules to preserve the user’s privacy (see 4.4.1 Default Rules in detail). The picture below is the reaction.

![Figure 5.32 System’s Response When Default Rule is Fulfilled](image)

As in Figure 5.32, as the doctor mainly deals with heart disease, the system only responds with data involving heart rate.

Up to the present time, all the privacy situations have been implemented.
6. **Evaluation**

To test the performance and capacity, the Android demonstration is used to perform different operations including login, local data processing, online data processing, and privacy rule implementation. At the same time, notice will be given in relation to the application’s performance and the server’s reaction.

6.1 **The Diversity of Privacy Rules**

If (target) (property) (judge) (value), then (action).

Figure 6.33 Different Properties for Different Targets

Figure 6.33 shows the different parts for setting privacy rules. When the user chooses “doctor’s” as a target, then it is only possible to choose “name”, “address” or “profession” as its property. When the user chooses “my” as the target, then he/she can choose the remaining properties, which are “SBP”, “DBP”, “blood sugar”, “heart beat”, “visit day”, “visit count”, and “privacy level”. Thus, there are $1*3*2*4 + 1*6*4*4 + 1*1*2*4 = 128$ choices in one rule.

6.2 **The Scalability of Privacy Rules**

If the requirement is to add some other items for users to choose, only the additional part would be inserted and the whole system will still work without much modification. Thus, the items below are not required to change if something is added to the privacy item in the future:

- The structure of privacy rules
- The syntax of the expression of rules
- The previous rules

Thus it can be stated that the privacy rules are flexible.
7. Conclusions

When a comparison is made with a paper-based health system, the electronic health system has so many advantages that it will soon replace the paper-based system. In addition, privacy preserving has made people more aware and this area will involve significant efforts.

In this thesis work, some correct means of storing data have been discussed as well as protocols based on the real demands of an e-health system. In addition, some correct methods for implementing privacy rules were discovered. Additionally, the programming for the complete process of privacy preserving, including android interface, server, database, and protocols has been completed.

To evaluate the performance, an Android demonstration was developed to perform different operations and, simultaneously, the reaction of the server in relation to the application performance was noted.

However, no technology comes without challenges, and this application still has some limitations for future improvement:

(1) When the users want to set their own privacy rules, the relationship between target and property is only mentioned in passing. Although a warning will be provided to the users when they set an inappropriate rule, one more efficient way is that when the user chooses one target, the system will then only show the matching properties or vice versa.

(2) When users set an inappropriate rule, he/she can only choose “Clear all”, but is not able to merely delete that particular rule. In addition, all the rules are displayed as a whole, not one by one and thus in the future it should be possible to realize one means of storing rules dynamically and to enable them to be operated separately.
References


Department of Information Technology and Media, Mid Sweden University, Sen-sorNet 2013, February 2013


[4] Xin Zhang, “Distributed Electronic Health Record System based on Middleware”, Department of Information Technology and Media, Mid Sweden University, May 2013

http://www.hhs.gov/ocr/privacy/hipaa/understanding/summary/


http://en.wikipedia.org/wiki/Eclipse_(software)
Privacy Preserving Implementation in the E-health System
Zongzhe Chen 2013-06-29

http://moss.csc.ncsu.edu/~mueller/g1/app.pdf
Published 2008-12-03. Retrieved 2013-03-25.

Published 2012-03-12. Retrieved 2013-05-01


http://en.wikipedia.org/wiki/SQLite

http://en.wikipedia.org/wiki/MySQL