DOI: 10.2298/CSIS110315055M

Privacy Preserving in Ubiquitous Computing: Classification & Hierarchy

Tinghuai Ma[1], Sen Yan[2], Jin Wang[3], and Sungyoung Lee [3]

[1]Jiangsu Engineering Center of Network Monitoring, Nanjing University of Information Science & Technology,
210044 Nanjing, China
thma@nuist.edu.cn
[2]School of Computer & Software, Nanjing University of Information Science & Technology,
210044 Nanjing, China
perhaps2532@gmail.com
[3]Department of Computer Engineering, Kyung Hee University,
446701, Suwon, South Korea
{wangjin,sylee}@oslab.khu.ac.kr

Abstract. In this paper, we adopt the classification of personal information and hierarchy of services to build a privacy system, in which one communicates with each other via pipes with different security levels. In each level, one has the corresponding rights to access each other. The requesters are not able to be infringed based on the personal information that service obtains from service providers. Privacy system can decrease the interaction, while in other circumstance the system strengthens and enhances the privacy preserving. Thus we strike a balance between two goals of Ubiquitous Computing: interaction and privacy preserving.

Keywords: ubiquitous computing, privacy preserving, classification, hierarchy.

1. Introduction

Ubiquitous computing represents the concept of seamless „everywhere‟ computing and aims at making computing and communication essentially transparent to users. In ubiquitous computing, we will be surrounded by a comfortable and convenient information environment that merges physical and computational infrastructures into an integrated habitat [1]. Context-awareness will allow this habitat to take on the responsibility of serving users by tailoring itself to their preferences as well as performing tasks and group activities according to the nature of the physical space. So, the more an application is aware of the user‟s context, the better it can adapt itself to assist him. Richer contextual and deeper personal information facilitates better automation and adaptation [2], [33]. Unfortunately, transparent soliciting, acquiring and handling of personalization constructs raise privacy concerns [33]. Thus, privacy in ubiquitous computing has been a contentious issue. The privacy concerns have been raised to suggest that privacy may be the greatest barrier to the long-term success of ubiquitous computing [3]. The privacy preserving in ubiquitous computing becomes a hot topic recently [25], [30].

There are two privacy preserving techniques in ubiquitous computing, one is the anonymous / pseudonym oriented, and the other is the policy based.

For the anonymous based privacy enhancing technology (PET), there are *k*-anonymities that they can not be distinguished from each other. In other words, there are *k* entities having the same ID or in the same location at the same time. The higher value of *k*, the higher level of anonymity is [4]. The service provider doesn"t want to misuse of data because there is a mixture of true and *k*-1 sensor data. Even though there are *k* IDs, the privacy sensitive information may be revealed to an attacker through track analysis. The combination of frequently varying pseudonyms and dummy traffic is used to prevent it [5]. For some ID based services, the virtual identities are used to conceal the user"s real identity [6]. Anonym and authentication are always conflict. Diep et al [7] present a scheme using anonymous user ID, sensitive data sharing method, and account management to provide a lightweight authentication while keeping users anonymously interacting with the services in a secure and flexible way. He et al [8] use the blind signature to encrypt anonymous ID, which can support the anonymous ID"s authority.

For the policy based privacy enhancing technology, the main concept is to store privacy-compliant rules to process personal information. The stored privacy policies describe the allowed recipients, uses, and storage duration of users" data. Also, a policy engine is used to reason the compatible privacy policy. Now, privacy policy is described in XML [9], [10] and XACML [11], [27], [28], [30], [31]. Most privacy policy based PET are focus on one or more scenarios. The configuration of policy for each scenario is a huge work [12]. The management and deducing are the main concerns in policy based system [13], [14], [15], 16]. Actually, the policy based PET is using access control model to make sure the security of personal data and service. Policy based PET is simply, intuitive, but not easy to deployment in wide area.

Using XML or XACML to describe accessing control rule is the main idea of policy based PET"s. It focuses largely on conventional data management schemes to support user"s privacy preserving. With different situations and user-specific privacy granularity, the privacy preserving rules configuration is exhaustively. Reducing the rule number is helpful to configure the whole privacy preserving system.

The goal of this paper is to design a possible way to accomplish least interactions between services and users, and thus accentuate “seamlessness” between virtual world and physical world, making services transparent to users anywhere anytime, making users immerse to services anywhere anytime. The proposed system in this paper offers a new secure approach to decrease the interactions. Compared to the existed ones, this system is suitable for a small-scaled area. In this paper, we classify the user and service into three types separately. Then, the rules, which indicate what data can be accessed, will be simpler. It will simplify huge rule configuration into several rules setting.

The remainder of this paper is organized as follows: Section 2 presents the related work about policy based PETs. Then we present our new mechanism about how to tackle privacy in the system in section 3. We introduce some policies and strategies that will be of great help and integrate them as a policy-making system in Section 4, also the practical analyses are conducted. In section 5, an integrated privacy system is described, and an example system is implemented to demonstrate the performance. Finally, we concluded the whole paper in Section 6.

2. Related Work

There have been many works in the area of privacy preserving in ubiquitous computing [17], [23], [24]. They emphasized the importance of privacy preserving, clarified the problems that we must not ignore. Most of the work focuses on how to express privacy preserving polices and the architectures. But these solutions have not addressed the policies configuration automation and the configuration efficiency.

Approaches closely related to our work have been investigated in two different areas: XACML based policy configuration and access classification.

XACML based policy. XACML is an XML-based language which specifies access control policies, which is called the extensible Access Control Markup Language. Policy based PET focuses on controlling the data leakage, in other words, configuring the rule what data can be accessed.

In service access control model, Dai et al. [31] propose an access control model based on XACML in web services, which provides the process of access control for the requesters and services in web services. Malik et al. [32] use XACML to describe the access control rules which require prioritization and conflict handling mechanism. It is used for control web service sharing, which preserving the privacy of personal context and shared context.

For data access control model, XACML also acts an important role, especially for health data. Arunkumar et al. [30] use XACML policy to control the privacy and data access through handheld devices. Ardagna et al. [28] combine the XACML and SAML to enable privacy-preserving and credential-based access control. Kodeswaran et al. [29] propose a framework to allow users to control how their data is used, and extend existing access control languages to enable researchers to use the aggregate data avoiding privacy leakage.

Access classification. Policy configuration is a huge work [12]. Automated policy configuration is the destination of policy based PET. But it is on the way for this scenario coming. Decreasing the rule number may reduce configuration pressure. First, the privacy data are classified into groups. Ning et al. [26] design a documents broadcasting approach, which is based on access control policies specifying which users can access which documents, or subdocuments. Second, Kodeswaran et al. [29] classify the access process into complete access, abstract access and statistical sccess. Then, policy infrastructure is described, which proves individual privacy is protected.

3. Classification & Hierarchy

In accordance with the previous fundamentals, we advance some new concepts about privacy preserving in Ubiquitous Computing.

3.1. Classification of Personal Information

Definition 1: Classification of Personal Information. Classification is taxonomy. According to the relevance to a real entity, information of such entity can be classified as:

1. Direct information, the part which can be directly referred to a person without any reasoning, such as name, ID number, address.
2. Indirect information, the part which might be referred to a person with reasoning, or have some relationship with a person‟s privacy, such as medical history.
3. Relevant information, the part which cannot be referred to a person, such as hobbies.

Most of us are family with Object Oriented Programming languages (OOP), one of its core ideas is that it introduces a concept of Class. The objects are encapsulated as three types or levels: Public, Protected and Private. Our idea of Classification is derived from this. From the previous work we have done, personal data comprises the identity and the profile two parts. We might categorize the information as Private Level (PriL), Protected Level (ProL) and Public Level (PubL), respectively. Here we draw an intuitive figure to illustrate our structure as Fig.1.

On the basis of Fig.1, Table I is listed, which presents the rights of corresponding service type. Full means accessing information without interaction is allowed. Limited means when necessary, it will need mandatory interaction to get permission to proceed. No means no permission and interaction, individual devices deny automatically. The data flow arrows in the Fig.1 represent the directions of access rights, that is, the back-end of the arrow can access the fore-head of the arrow. Generally speaking, the system grants high level rights to access the low level. In the system defined area column, since the data is strictly defined by the system, the access rights are thus strictly granted from high to low. In the user defined area column, there are some distinctions: considering the personalization of individual database, between the Private and Protected level, we grant the latter one the limited rights to access the former one with some kind of responses to the entity it affiliates such as alarm, vibration, sound, to remind the user to decentralize the accessing rights, so that it can proceed and collect the necessary information.

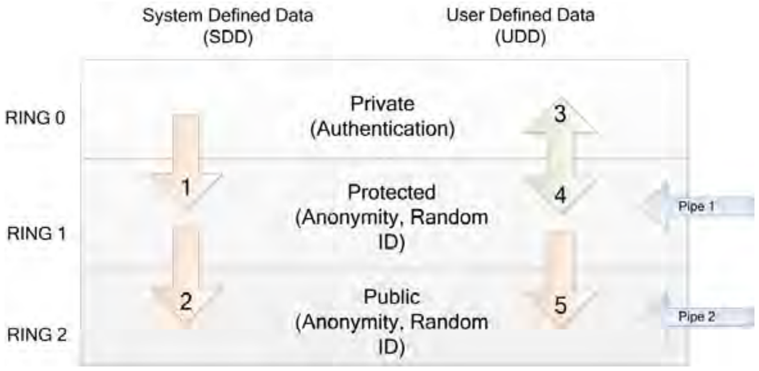

Fig.1. Structure of privacy system. The arrow means the accessible directions of the data flow. Privacy system is divided into two parts: the System Defined Data (SDD) area and the User Defined Data (UDD) Area.

Table 1. Rights Table of Accessing. The accessing types are classified according to SDD, UDD respectively.

Service Type	Public UDD	Public SDD	Protected UDD	Protected SDD	Private UDD	Private SDD
Service Private	Full	Full	Full	Full	Limited	No
Service Protected	Full	Full	Limited	Limited	Limited	No
Service Public	Full	Full	No	No	No	No

Public Service. After a piece of ubiquitous terminal enters the specific district of ubiquitous environment, the public service can contact directly with the terminal, read the whole information of public level without notifying the person who holds it, so there is no interaction. If the service matches with the preference of device and the policies of privacy system, the terminal notifies the user that a service called him. If one of the following conditions is satisfied:

1. The service is not satisfied the preference or policies;
2. The authorized service is trying to read the information of high level;
3. The existing preference is not enough to judge;
4. The terminal rejects all public service;

5. The terminal set the value that always rejects this kind of service.

The terminal rejects automatically without notifying.

Protected Service. It has the similar performance as the above. The protected service can not communicate directly with the terminal. Both of them must register in the environment. If a protected service needs to read the information in the protected level, the privacy system must notify the user. But this doesn‟t mean that every such kind of services need the permission to access, some of them just notify the user what they are doing. If the service meet the above five conditions listed in the public service part, the same operation will be made here.

Private Service. It is quite similar with the protected service. The main difference between them is the rights of accessing information of users.

As it can be seen from Fig.1, the whole system consists of three rows and two columns. Let us analyze them one by one in the following.

Row. PriL*:* Marked as “Ring 0 Level” in Fig.1. In this level, the identity includes name, ID number and so forth. The system provides no rights and direct access to external Service Providers (SP) for the protection of authentic ID information. The service that needs the information of this level must request from pipe 1. Depending on the willingness of entity, the system denies or accepts the requests.

ProL: Marked as “Ring 1 Level” in Fig.1. In this level, some aggregation of the profiles such as medical history is stored that are related to the privacy.

PubL: Marked as “Ring 2 Level” in Fig.1. In this level, the other types of the profiles are stored, mainly including some unimportant data that are generally irrelevant to the identity of a certain individual. That means, if the real identity is masked, no entity could utilize any mechanism to infer the real entity from the total profile it got.

Column. There are two reasons for system being divided into SDD and UDD: first, we strike the balance between security and personalization here, and the dotted line separates it as two independent storage space; second, since they are independent, the fails or threats in the User Defined Data Area have limited influence on the System Defined Data Area that is the core area of the system. In SDD Area, we can pre-define the following sets:

Private Data Entries ($PriDEs$) contains the crucial information such as name and ID, which are relevant to the identity of a certain person. Usually, only the Authentication System has the rights to read it for authentication in some certain circumstances. It is can be represented as follows.

$$PriDEs = \{name, IDnumber, Sex, \ldots \vdots \; Items \quad related \quad to \quad identity\} \qquad (1)$$

Protected Data Entries ($ProDEs$) are some data that are accessible in some circumstances by some objectives, but not accessible in other circumstances by some other objective. Take medical history for an example, not every hospital is trustworthy enough to read you medical history that exists in the Ubiquitous Computing Environments. It is a mathematical set:

$$ProDEs = \{medical\ history, \ldots \vdots\ Items\quad partially\ or\ partly\ related\quad to\quad privacy\} \tag{2}$$

Public Data Entries ($PubDEs$) are some data that are related to hobbies, interests and so on. This information can be read by any service providers (SP). Since the random ID stream displaces the real identification immediately when it is in the range of available services, it is hard to trace the entity. We present it by the following set:

$$PubDEs = \{(Potato, Love), (Tomato, Dislike), \ldots \vdots\ Items\quad irrelevant\quad to\quad privacy\} \tag{3}$$

3.2. Hierarchy of Services

Definition 2: Hierarchy of Services. In this paper, hierarchy of services refers to different service types that require different data type:

1. Privacy-based services, requiring private data and requesting the personal data via pipe 1, such as bank services, which need ID;

2. Protection-based services, requiring protected data and requesting the personal data via pipe 1, such as hospital services, which need Medical history;

3. Public-based services, requiring public data and requesting the personal data via pipe 2, such as weather services, which need temperature data.

Table 2. Basic needs of different services

Service Type	Security Agent	Authentication	Alarm
Service Private	Need	Need	Yes
Service Protected	Need	Need	When necessary
Service Public	No	No	No

Categorizing the services obviously facilitates the management of the services. Table 2 shows the basic needs for different sorts of services. In the ubiquitous computing environments, even in a district area, there could probably be many services. Since quite a lot of services just require the public information, it is feasible to make these services contact with entities directly without security agent and authentication towards public service. We also have to clarify the worry about possible leakage of personal information. Tinghuai MA et al. [18] have utilized a spatiotemporally-based anonymity method and demonstrated that real identity can be masked. On one hand, it can light the system burden in the trusty area; on the other hand, it lowers the doorsill of service-providing and booms the passion of service provider to provide abundant services. The entity can avoid harassment from those services irrelevant or not interested by configuring the terminals using preference, combining the system"s strategy (we will discuss a strategy named $N\mathrm{th}$ push strategy in the following), the entity can efficiently avoid the majority of unwelcome services and spam services. However, for the services needing protected information or private information, authentication and security agent are needed for entities to make sure the service he will receive is genuine and valid and the security agent authenticates the entities as well in the same time, confirming neither of two parties will jeopardize each other.

3.3. General Process

The three level services are donated as $Service_{\mathrm{Pr}iL}$, $Service_{\mathrm{Pr}oL}$ and $Service_{PubL}$ correspondently;

Thus we have the following process, the “have rights” means have rights to access information that is relevant to the respective service, or essential information that the service needs.

1. The Service Provider (SP) of $Service_{PubL}$ communicates with the entity directly through pipe 2, no intermediate Access Point (AP) needed, $Service_{PubL}$ are pushed to the entity without interaction, and no rights of accessing Ring 1 and Ring 0 are granted.

2. The Service Provider (SP) of . $Service_{\mathrm{Pr}oL}$. communicates with the entity through pipe 1. It has full rights to access Ring 2 via arrows 2 and 5, limited rights to access Ring 1, and strict rights to access Ring 0 via arrow 3, depending on the user"s response, mandatory interaction needed.

3. The Service Provider (SP) of . $Service_{PubL}$. communicates with the entity through pipe 1 too. It has full rights to access Ring 2 via arrow 2 and 5, full rights to access Ring 1 via arrow 1 and 4, and limited rights to access Ring 0, mandatory interaction needed.

4. Filter Policy

A satisfactory system consists of not only high-security architecture but also reliable policies to safeguard users" privacy. In this paper, we call the aggregation of those policies as Policy-making System.

4.1. Policy-making System

Here we introduce a system called policy-making system, its duty is to predicate the validation of present action via comparing with the former actions and determine the appropriate authorization and authenticating the entities when necessary.

The Fastest Velocity Policy (FVP). This policy can be expressed as follows: The system records location and time information of every access of an individual in Private Level and compares the corresponding information. Furthermore, if necessary, it calculates the fastest velocity and compare with the possible velocity. Let us see the exact mathematical table of certain service and interpret it firstly. Here we present policies of abnormality detecting and the details of Table 3.

Number. To record the sequence of the entry.

Access Time. It refers to the starting time and aborting time of a certain kind of service. Since it is the table of a certain kind of service, comparing with each starting time and aborting time makes some sense in some circumstances. With the increasing records, system will compute the time distribution to record or filter abnormal accessing requests, and raise the risk level to a certain degree.

Location. It refers to the contemporary location when one is requiring a kind of service, the location set is made up of three elements: x, y and z, representing location coordinates, respectively. We can easily calculate the distance between two places.

Velocity. The velocity can be calculated via the displacement and time interval between two neighboring entries. In practice, we have the possibly fastest velocity between the two neighboring place coordinates, when the calculated velocity value is beyond the possibly fastest velocity, the system can block this request because it must be an unauthentic or unauthorized request. The highest value of velocity can vary according to different areas and districts, and this causes different pre-set value.

Risk Level. The risk level is evaluated by the former three columns: Access Time, Location and Velocity. The initial value of risk level, of course, can be set as 0, that is, when the risk level is 0, the request is authentic and secure. When the security lever is below 0, it causes an alarm and the alarm level varies based on the degree.

Based on the former time distribution possibility and location distribution possibility, the present access time and access location can be detected and located in the probability area in accordance with problematic judgment, namely, a present access time and location can be compared with the former access time and location habits, as shown in Table 3. Its time probability distribution chart can be described as Fig.2.

Table 3. Recording Table of a Certain Service

ID	Access Time	Location	Velocity	Risk Level	Remarks	Normal
1	[t1]	[l1]	N/A	N/A	start	N/A
2	[t2]	[l2]	N/A	N/A	abort	N/A
3	[t3]	[l3]	(l3- l2) / (t3-t2)	evaluation	start	evaluation
4	[t4]	[l4]	N/A	evaluation	abort	evaluation
5	[t5]	[l5]	(l5-l4) / (t5-t4)	evaluation	start	evaluation
…	…	…	…	…	…	…
…	…	…	…	…	…	…
…	…	…	…	…	…	…
2n-2	[t2n-2]	[l2n-2]		evaluation	start	evaluation
2n-1	[t2n-1]	[l2n-1]	(l2n-1-l2n-2) / (t2n-1-t2n-2)	evaluation	abort	evaluation
2n	[t2n]	[l2n]	N/A	evaluation	start	evaluation
2n+1	[t2n+1]	[l2n+1]	(l2n+1-l2n)/ (t2n+1 - t2n)	evaluation	abort	evaluation
2n+2	[t2n+2]	[l2n+2]	N/A	evaluation	start	evaluation

According to the probability distribution, the risk level can be assessed in the following way: if a certain present action is in the most probability area, it marks itself as value 0, if in the second most probability area, it marks itself as value -1, and the rest can be done in the same manner. The lower the score is, the higher the risk is. The grade of probability area can vary in line with the practical needs.

Remarks. To remark the system state.

Normal. To assess whether the present request is valid and authentic via the former judgment and assessment, and present crucial information via a kind of human-readable way in this column.

In the same way, we can also construct a space table: we demarcate a certain area into small ones, record the login places vaguely or precisely, compute the probability, and assign each one an addition value. So we have the following mathematical expression to determine the risk level preliminarily.

$$Compare\langle (V_{time} + V_{space} + \cdots) \,|\, (V_{threshold}\{\ldots\}) \qquad (4)$$

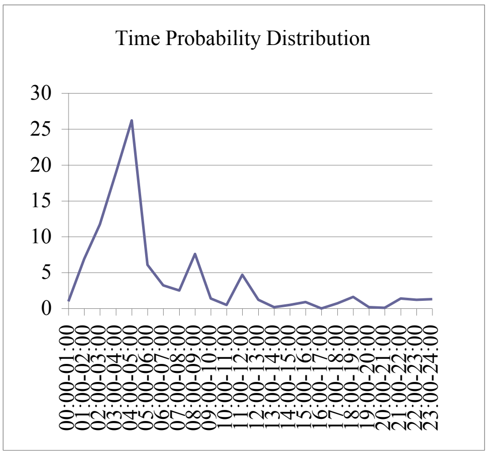

Fig. 2. Line chart of time probability distribution

If the value of former one is lower than the latter threshold value, it alarms. $V_{threshold}\{\ldots\}$ can have multiple values, and we can set alarming level, respectively.

Algorithm of Comparison. Suppose an entity requires logging in when it is entering a certain area:

Step 1: Record the present location and time information;

Step 2: Compare the former recordings, evaluate the values and add them;

Step 3: Compare the aggregation of the present value with the threshold;

Step 4: ($Value_{Present} \le threshold$) system denies the login request and informs the entity of authentication;

Step 5: ($Value_{Present} \ge threshold$) system accepts the login request.

N-th Push Strategy. Nth Push Strategy is a Service-Orient Strategy, the purpose of which is to reduce the unnecessary interaction, which is also one of the purposes of Ubiquitous System. We assume there are many kinds of services in a certain area, based on the profile and preference of an individual who enters the area the services can be categorized into several sorts as follows:

1. The services that one is interested in;
2. The services that one is not interested in;
3. The services that one is uncertain or that one doesn‟t mention.

Since it is a strategy, it can vary itself in accordance with the relevant ambience and relevant entities. Here we just adopt a simple scenario that consists of two simple groups, and the strategy is defined as follows:

1. Define the services that one interests in (judged by use frequency and individual preference) or concerns to the data of private level or protected level have rights to push 2 times.

$$(Service_{interests} + Service_{PriL} + Service_{ProL} \mid Times_{2}) \tag{5}$$

2. Define the services that one does not interest in have rights to push 0 times.

$$(Service_{no-\mathrm{int}erests} \mid Times_{0}) \tag{6}$$

3. Define the services that one is uncertain have rights to push 1 times.

$$(Service_{uncertain} \mid Times_{1}) \tag{7}$$

4. If the services that one is uncertain are relevant to his existing preference, set an additional times, that is 1+1 times

$$(Service_{uncertain-relevant} \mid Times_{1+1}) \tag{8}$$

5. If the services that one is uncertain are irrelevant to his existing preference, set subtraction times, that is 0 times.

$$(Service_{uncertain-irrelevant} \mid Times_{1-1}) \tag{9}$$

6. If the services that one is uncertain and cannot be judged through his existing preference, remains.

The user preference is defined by him as follows:

1. Accept the interested Nth times-based services, N≥1;

2. Filter the services not interested in to reduce the unnecessary interaction.

Suppose there is an entity who is entering the effective area, system generates a new random ID to identify the entity:

$$Entity\begin{pmatrix} \mathrm{Pr}\,operty\{ID_{random},\cdots\} \\ \mathrm{Pr}\,eference\{\ldots\} \end{pmatrix} \tag{10}$$

And there are services categorized by level:

$$Service\begin{pmatrix} Service\{Service_{private\ level}\} \\ Service\{Service_{protected\ level}\} \\ Service\{Service_{public\ level}\} \end{pmatrix} \tag{11}$$

Or categorized by user"s preference

$$Service\begin{pmatrix} Service\{Service_{\text{int}\,erest}\} \\ Service\{Service_{not\,\text{int}\,erest}\} \\ Service\{Service_{uncertain}\} \end{pmatrix} \tag{12}$$

So the services that may be useful to the entity are $Service_{usefull}$, M means Match.

$$\begin{aligned} Service_{useful} = {} & Service(\ldots) \\ & M \\ & Entity(\Pr eference\{\ldots\}) \end{aligned} \tag{13}$$

In this system, we separate the authentication and delivery of a certain service, though the integration may be more simple and convenient for us to build. The direct advantage is that even the Service Delivery fails because of the system"s heavy burden, or something else, it cannot influence the system"s authentication function. This can avoid one situation that we cannot even register.

4.2. Practical Analyses

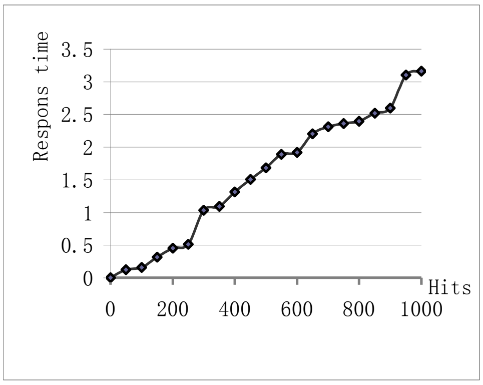

Fig. 3. Line chart of hit probability distribution

We conducted a simulation on our computers to prove our policy-making system"s efficiency. The computer simulates the hit probability trend when the times user used increase as shown in Fig.3.

The X axis represents the frequency in which service has been called. The Y axis represents the system"s response time of matching the suitable service. From the chart we can see that when a certain service had been used for initial 100 times, the policy-making system cannot work efficiently because of lacking of adequate data and distribution trend. The situation changed a lot when the service had been used more than 100 times, the policy-making system takes effect. It demonstrates that this system is not suitable for contemporary services or services that one doesn"t use so much.

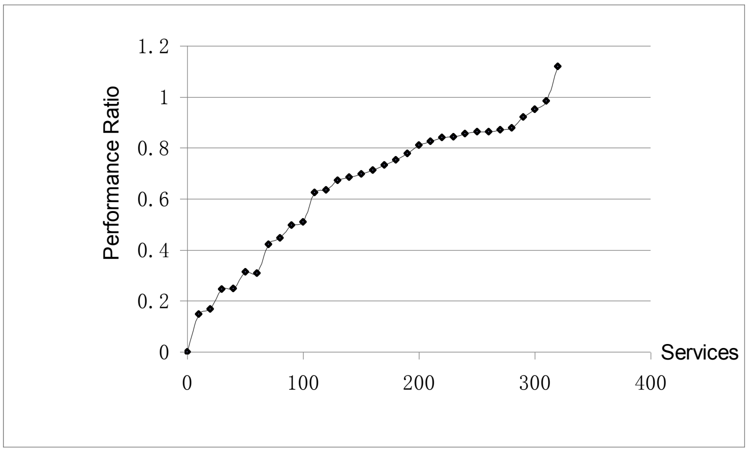

Fig. 4. The performance of system according the service number

We also examine the performance of policy-making system, which shown in Fig. 4.

We set the Performance Ratio $R = \tau_{actual} / \tau_{desire}$, if τ_{desire} is set to 1 second, the response time τ_{actual} should be below 1 second in order to get better performance. X axis is the number of available services.

According to our simulation, we can estimate that if there are about 320 services, the system works very well. However, if there are more than 320 services, the system is not able to respond in time. This is just a simulation, and later the mathematical calculation will be given to prove the simulation from another aspect.

5. System Overview

In this section, we integrate Policy-making System into Privacy System, and then give mathematical analyses. Finally we show a practical example to examine the whole system.

5.1. The Infrastructure

According to our previous design, the system is a critical part of infrastructure in a ubiquitous environment, which is in charge of access control of services, and the policy-making system as a supplementary part and the discrimination of services. The whole infrastructure can be illustrated as Fig.5.

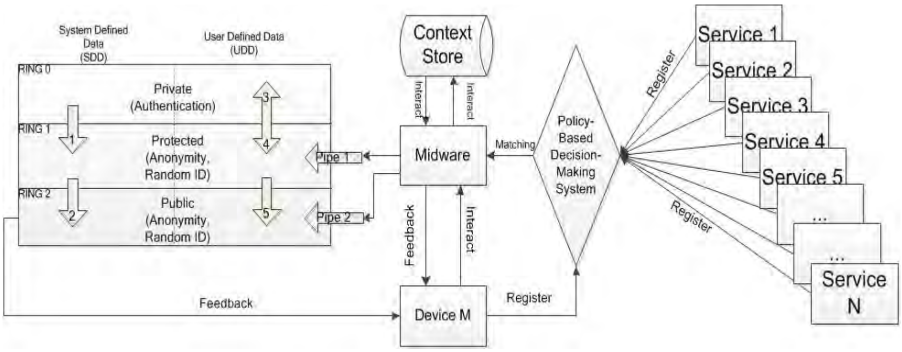

Fig. 5. Infrastructure of the privacy system

The infrastructure works as follows:

1. Available Services register in the Policy-Based Policy-Making System to inform the system of its existence and availability;

2. Device M enters the ubiquitous environment, registers anonymously in the Policy-Based Policy-Making System to inform the system of its existence and availability;

3. Policy-Based Policy-Making System synthesizes both service profile and device profile, and pushes the matching services to middleware;

4. Middleware check the Context Store and records the activities of Device M. Context Store records the history of Device M. If the current activities of Device M are abnormal, for example, alarm Device M to verify identity;

5. Middleware also separates the matching services, push them via Pipe 1 or Pipe 2, according to the information the services need;

6. Services are pushed to Device M.

5.2. XACML Description

XACML is adopted to express fine grained access data control, which describe the access process of Fig.1. Standard XACML includes three basic elements: <Rule>, <Policy> and <PolicySet>. <Rule> has two elements, <Condition> and <Target>, and an Effect attribute. The rule means: if the condition of the rule evaluates to be true, then the access control decision to perform <Actions> by the <Subjects> on the <Resources> are given by the Effect attribute.

Fig.6 shows an example XACML policy that specifies a permission to get data. Lines 3-9 define the policy"s target, which indicates that this policy is applicable to Private subject requesting permission to execute any action on UDD and SDD private data. Lines 11-37 indicate applicable requests to those requesting accesses to the resource data with the action GetData. Lines 38-44 indicate that if the data does exist, the request should be permitted.

```
1.<Policy PolicyId="GetPrivateData"
2.     RuleCombiningAlgId="permit-overrides">
3. <Target>
4.            <Subjects ServiceType= Private></Subjects>
5.            <Resources >
6.            <UDD Type=Private><SDD Type=Private>
7.            </Resources>
8.            <Actions><AnyAction/></Actions>
9.      </Target>
10. <Rule RuleId=" GetPrivateData " Effect="Permit">
11.     <Target>
12.           <Subjects><AnySubject/></Subjects>
13.           <Resources>
14.                 <Resource>
15.                       <ResourceMatch MatchId="string-equal">
16.                             <AttributeValue DataType="string">
17.                                        data
18.                                   </AttributeValue>
19.                             <ResourceAttributeDesignator
20.                                    AttributeId="resource-id"
21.                                         DataType="string"/>
22.                       </ResourceMatch>
23.                 </Resource>
24.           </Resources>
25.           <Actions>
26.                 <Action>
27.                       <ActionMatch MatchId="string-equal">
28.                             <AttributeValue DataType="string">
29.                                   GetData
30.                             </AttributeValue>
31.                             <ActionAttributeDesignator
32.                                        AttributeId="action-id"
33.                                        DataType="string"/>
34.                       </ActionMatch>
35.                 </Action>
36.           </Actions>
37.     </Target>
38.     <Condition FunctionId="Yes">
39.           <Apply FunctionId="data-exist">
40.                 <ResourceAttributeDesignator
41.                           AttributeId="new-data-id"
42.                           DataType="data"/>
43.           </Apply>
44.     </Condition>
```

```
    45. </Rule>
    46. </Policy>
```

Fig. 6. Get private data policy based on XACML

5.3. Cost of Evaluation

We assume *C* as cost of evaluation, m as the number of devices, n as the number of services, $p_{1,j}$ as failure probability of service from service 1 to service j, $p_{2,j}$ as the failure probability of system from service 2 to service j, i and j as two random services, $t(0 \le t \le 1)$ as the adjustment parameter set by system, then the following equation could be inferred:

$$C = \left[1 + \sum_{i=1}^{n-1} \prod_{j=1}^{i} p_{1,j}\right] + \prod_{i=1}^{n} p_{1,i} * \left[1 + \sum_{i=1, i \neq n-1}^{m-1} \prod_{j=1, j \neq n}^{i} p_{2,j}\right] * t + \left(1 - \prod_{i=1}^{n-1} p_{1,i}\right) * \left[1 + \sum_{i=1}^{m-1} \prod_{j=1}^{i} p_{2,j}\right] * t \tag{14}$$

The typical system can be inferred as:

$$C_{typical} = \left[1 + \sum_{i=1}^{m-1} \prod_{j=1}^{i} p_{2,j}\right] + \prod_{i=1}^{n} p_{2,i} * \left[1 + \sum_{i=1, i \neq n-1}^{n-1} \prod_{j=1, j \neq n}^{i} p_{1,j}\right] * t + \left(1 - \prod_{i=1}^{m-1} p_{2,i}\right) * \left[1 + \sum_{i=1}^{n-1} \prod_{j=1}^{i} p_{1,j}\right] * t \tag{15}$$

Since we want to compare the cost, we have:

$$F = C / C_{typical} \tag{16}$$

It is a large equation, but lucky enough if we just want to prove that our system is more efficient at a certain value range, we do not necessarily have to calculate all its values. We can just demonstrate $F \le 1$.

We set $t = 1$ and omit the process of verification here. Based on accurate calculation, the threshold (M, N) should be (136, 314). That means, in this ubiquitous environment, under ideal conditions, there could be at most 136 devices and 314 services existed. If more, the system is not advantageous compared to a typical system existed before.

6. Conclusions & Future Work

We present a privacy-preserving architecture utilizing the classification of personal information and hierarchy of services, which are derived from the concept of Class in the Object Orient Programming. Based on such concept, personal information is manually or automatically categorized into Private (it can infer the real identity of an entity), Protected (it may infer the real identity of an entity) and Public (it cannot infer the real identity of an entity). All the three kinds of services have two independent storage spaces logically in order to make sure they would not be able to affect one another even some part is cheated by malicious users. Exposing the public information is somehow no harm to the entities, at the same time it could lower the doorsill of service-providing and ultimately booms the quantity of services. The categorized information of entity and categorized services are concerted, and the latter require information via respective pipes, one has lower security, suitable for services that need public information only without interactions, one has higher security, suitable for the rest two types. When it comes to the information that is not public, mandatory interactions are needed to alarm the entity and request the permissions to proceed. Besides, we combine some filter policies which is probability-based policy aiming to execute the preliminary judgment to judge whether the present register is suspicious or not, then choose appropriate actions, alarm, vibration, and require entity authentication in another way, including password, pre-designed questions or other security mechanisms. The other is called *N*th Push Strategy, aiming to filter the services that one does not need to combine with users" preferences, avoiding unwelcome services and unessential interactions, to meet the requirements of least interactions in the Ubiquitous Computing.

Our system has the following significant advantages compared to other similar system:

1. Higher efficiency. Our system is highly efficient, especially when it comes to ubiquitous terminals, most of which are battery-powered with slower CPUs. Through executing a simple set of rules, system can block most of services that are irrelevant to users as well as spam services, although not all of them.

2. Highly customization. Our system allows users to specify their preferences in a very simple manner, matching and discarding most services without keeping alarming the users what it is doing.

3. Security. The system adopts the hierarchy and classification to ensure that compromise of a single part will not jeopardize the whole system, especially the SDD area.

4. Compatibility. Ubiquitous environment is a device-rich environment, which means compatibility is a major issue to a variety of different devices. Basically speaking, any device that is applied to this system can be compatible with one another.

5. Adaptability. The system allows the same task to be performed differently in different environments. The only thing that one should do is to achieve this and to adjust the system configurations in various environments.

Though our system can work appropriately and satisfactorily, there still exist many problems compared with other existing systems. Basically speaking, it is a behavior-based system. When it comes to a service that a user has used many times, with the former behaviors profile, the system can discriminate and adjudge the present behaviors appropriately. However, when it comes to a new service, the policy-making part of whole system is nearly useless. So one emergent problem we must solve is that we should accomplish how to make this part of system work appropriately and satisfactorily even when it confronts a new service. The other work we must solve is that we should optimize the algorithms to fit with the needs of the large scale computing and accurate computing.

Acknowledgement. This work was supported in part by the National Science Foundation (61173143), and also supported by A Project Funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions. This research was also supported by the MKE (Ministry of Knowledge Economy), Korea, under the ITRC (Information Technology Research Center) support program supervised by the NIPA (National IT Industry Promotion Agency)"(NIPA-2009-(C1090-0902-0002)).

References

1. Weiser, M.: The future of ubiquitous computing on campus. Communications of the ACM, Vol. 41, No. 1, 41-42. (1998)
2. Tent ori, M., Favela, J., Gonzalez, V.: Quality of Privacy (QoP) for the Design of Ubiquitous Healthcare Applications. Journal of Universal Computer Science, Vol. 12, No. 3, 252-269. (2006)
3. Hong, J. I., Ng, J. D., Lederer, S., Landay, J. A.: Privacy Risk Models for Designing Privacy-Sensitive Ubiquitous Computing Systems. In Proceedings of ACM conference on Designing Interactive Systems (DIS2004). ACM, Cambridge, Massachusetts, 91-100. (2004)
4. Sweeney, L.: k-Anonymity: a model for protecting privacy. International Journal on Uncertainty, Fuzziness and Knowledge-based Systems, Vol. 10, No. 5, 557-570. (2002)
5. Cheng, H. S., Zhang, D., Tan, J. G.: Protection of privacy in pervasive computing environments. In Proceedings of International Conference on Information Technology: Coding and Computing, (ITCC 2005). IEEE Computer Society, Washington, USA, Vol. 2, 242-247. (2005)
6. Papadopoulou, E., McBurney, S., Taylor, N., Williams, M. H., Dolinar, K., Neubauer, M.: Using User Preferences to Enhance Privacy in Pervasive Systems. In Proceedings of Third International Conference on Systems (ICONS 08). IEEE Computer Society, Cancun, Mexico, 271-276. (2008)
7. Diep, N. N., Lee, S. Y., Lee, Y. K., Lee, H. J.: A Privacy Preserving Access Control Scheme using Anonymous Identification for Ubiquitous Environments. In Proceedings of the 13th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications. IEEE Computer Society, Daegu, Korea, 482-487. (2007)
8. He, Q., Wu, D., Khosla, P.: The quest for personal control over mobile location privacy. Communications Magazine, IEEE, Vol.42, No.5, 130-136. (2004)

9. Langheinrich, M.: A Privacy Awareness System for Ubiquitous Computing Environments. In Proceedings of the 4th International Conference on Ubiquitous Computing(UbiCom2002). Springer-Verlag, Goteborg, Sweden, 237-245. (2002)
10. Myles, G., Friday, A., Davies, N.: Preserving privacy in environments with location-based applications. Pervasive Computing, IEEE, Vol.2, No.1, 56-64. (2003)
11. Zheng, Y., Chiu, D.; Wang, H., Hung, P.: Towards a Privacy Policy Enforcement Middleware with Location Intelligence. In Proceedings of 11th IEEE International Enterprise Distributed Object Computing Conference. IEEE Computer Society, Maryland, USA, 97-104. (2007)
12. Pallapa G., Roy, N., Das, S. K.: A scheme for quantizing privacy in context-aware ubiquitous computing. In Proceedings of IET 4th International Conference on Intelligent Environments. IET, Seattle, USA, 1-8. (2008)
13. Yee, G.: Using privacy policies to protect privacy in UBICOMP. In Proceedings of 19th International Conference on Advanced Information Networking and Applications. IEEE Computer Society, Taiwan, Vol.2, 633-638. (2005)
14. Kang, Y., Lee H., Chun, K., Song, J.: Classification of Privacy Enhancing Technologies on Life-cycle of Information. In Proceedings of The International Conference on Emerging Security Information, Systems, and Technologies. IEEE Computer Society, Valencia, Spain, 66-70. (2007)
15. Lee, B., Kim, H.: Privacy Management for Medical Service Application Using Mobile Phone Collaborated with RFID Reader. In Proceedings of Third International IEEE Conference on Signal-Image Technologies and Internet-Based System. IEEE Computer Society, Shanghai, China, 1053-1057. (2007)
16. Martimiano, L. A. F., Goncalves, M. R. P., dos Santos Moreira, E.: An ontology for privacy policy management in ubiquitous environments. In Proceedings of IEEE Network Operations and Management Symposium. IEEE Computer Society, Bahia, Brazil, 947-950. (2008)
17. Ma, T. H., Kim, S. D., Wang, J., Zhao, Y. W.: Privacy Preserving in Ubiquitous Computing: Challenges & Issues. In Proceedings of IEEE International Conference on e-Business Engineering. IEEE Computer Society, Xi"an, China, 297-301. (2008)
18. Ma, T. H., Yang, S., Tian, W., Liu, W. J.: Privacy Preserving In Ubiquitous Computing: Architecture. Information Technology Journal, Vol. 8, No.6, 910-916. (2009)
19. Machanavajjhala, A., Kifer, D., Gehrke, J., Venkitasubramaniam, M.: l-Diversity: Privacy Beyond k-Anonymity. In Proceedings of 22nd IEEE International Conference on Data Engineering. IEEE Computer Society, Atlanta, USA. (2006)
20. Kaya, S. V., Savaş, E., Levi. A., Erçetin, Ö.: Privacy-Aware Multi-Context RFID Infrastructure Using Public Key Cryptography. In: Ian F. Akyildiz, Raghupathy Sivakumar, etc (Eds.): Networking 2007. Lecture Notes in Computer Science, Vol. 4479. Springer-Verlag, Berlin Heidelberg New York, 263-274. (2007)
21. Peris-Lopez, P., Hernandez-Castro, J. C., Estevez-Tapiador, J. M., Ribagorda, A.: RFID Systems: A Survey on Security Threats and Proposed Solutions. In Proceedings of the 11th IFIP International Conference on Personal Wireless Communications (PWC'06). Springer-Verlag, Albacete, Spain, 159-170. (2006)
22. KIM, I., LEE, B., KIM, H.: Privacy Protection Based on User-defined Preferences in RFID System. In Proceedings of International Conference on Advanced Communication Technology-ICACT'06. IEEE, Phoenix Park, Korea, 858-862. (2006)
23. ROBINSON, P., BEIGL, M.: Trust Context Spaces: An Infrastructure for Pervasive Security. In Proceedings of First International Conference on Security in Pervasive Computing. Springer-Verlag, Boppard, Germany, 157-172. (2003)

24. ORTMANN, S., Langendörfer, P., MAASER, M.: A Self-Configuring Privacy Management Architecture for Pervasive Systems. In Proceedings of 5-th ACM International Workshop on Mobility Management and Wireless Access (MobiWAC). ACM New York, Chania, Crete Island, Greece, 184-187. (2007)
25. Moncrieff, S., Venkatesh, S., West, G.: A Framework for the Design of Privacy Preserving Pervasive Healthcare. In Proceedings of the IEEE International Conference on Multimedia and Expo. IEEE, Cancun, Mexico, 1696-1699. (2009)
26. Ning, S., Nabeel, M., Paci, F., Bertino, E.: A Privacy-Preserving Approach to Policy-Based Content Dissemination. In Proceedings of the IEEE 26th International Conference on Data Engineering. IEEE, Long Beach, California, USA, 944-955. (2010)
27. Xu, M., Wijesekera, D., Zhang, X.: Runtime Administration of RBAC Profile for XACML. IEEE Transactions on Services Computing, 1-1, Vol.3, No. 1. (2010)
28. Ardagna, C. A., De Capitani di Vimercati, S., Neven, G., Paraboschi, S., Preiss, F.-S., Samarati, P., Verdicchio, M.: Enabling Privacy-Preserving Credential-Based Access Control with XACML and SAML. In Proceedings of the IEEE 10th International Conference on Computer and Information Technology (CIT). IEEE, Bradford, UK, 1090-1095. (2010)
29. Kodeswaran, P., Viegas, E.: Towards A Privacy Preserving Policy Based Infrastructure for Social Data Access To Enable Scientific Research. In Proceedings of the 8th Annual International Conference on Privacy Security and Trust (PST). IEEE Computer Society, Ottawa, Ontario, Canada, 103-109. (2010)
30. Arunkumar, S., Rajarajan, M.: Healthcare Data Access Control using XACML for Handheld Devices. In Proceedings of the Developments in E-systems Engineering (DESE). IEEE, London, UK, 35-38. (2010)
31. Dai, C. Y., Gong, W. T., Liu, J.: The Research of Access Process in Web Services Based on XACML. In Proceedings of the 2nd International Workshop on Database Technology and Applications (DBTA). IEEE, Wuhan, China, 1-4. (2010)
32. Malik, A. K., Dustdar, S.: A Hybrid Sharing Control Model for Context Sharing and Privacy in Collaborative Systems. In Proceedings of the IEEE Workshops of International Conference on Advanced Information Networking and Applications (WAINA). IEEE, Biopolis, Singapore, 879-884. (2011)
33. Oyomno, W., Ja ppinen, P., Kerttula, E.: Privacy Preservation for Personalised Services in Smart Spaces. In Proceedings of the Baltic Congress on Future Internet Communications (BCFIC Riga). IEEE, Latvia, Riga, 181-189. (2011)

Tinghuai Ma is an associate professor in Computer Sciences at Nanjing University of Information Science & Technology, China. He received his Bachelor (HUST, China, 1997), Master (HUST, China, 2000), PhD (Chinese Academy of Science, 2003) and was Post-doctoral associate (AJOU University, 2004). From Nov.2007 to Jul. 2008, he visited Chinese Meteorology Administration. From Feb.2009 to Aug. 2009, he was a visiting professor in Ubiquitous computing Lab, Kyung Hee University. His research interests are in the areas of Data Mining and Privacy Protected in Ubiquitous System, Grid Computing. His research interests are data mining, grid computing, ubiquitous computing, privacy preserving etc. He has published more than 70 journal/conference papers. he is principle investigator of several NSF projects. He is a member of IEEE.

Sen Yang is a graduate at School of Informatics and Computing, Indiana University – Bloomington. He received his Bachelor degree in NUIST in 2009. His research interests are in the related areas of Human-Centered Computing, including Human-Centered Interaction / design, Ubiquitous / Pervasive Computing and Data Mining.

Jin Wang received the B.S. and M.S. degree in the Electronical Engineering from Nanjing University of Posts and Telecommunications, China in 2002 and 2005, respectively. He received Ph.D. degree in Ubiquitous Computing laboratory in the Computer Engineering Department of Kyung Hee University Korea. Now, he is a professor in Nanjing university of Information Science & technology. His research interests include routing protocol and algorithm design, analysis and optimization, and performance evaluation for wireless ad hoc and sensor networks.

Sungyoung Lee received his B.S. from Korea University, Seoul, Korea. He got hisM.S. and Ph.D. degrees in Computer Science from Illinois Institute of Technology (IIT), Chicago, Illinois, USA in 1987 and 1991 respectively. He has been a professor in the Department of Computer Engineering, Kyung Hee University, Korea since 1993. He is a founding director of the Ubiquitous Computing Laboratory, and has been a.liated with a director of Neo Medical ubiquitous-Life Care Information Technology Research Center, Kyung Hee University since 2006. Before joining Kyung Hee University, he was an assistant professor in the Department of Computer Science, Governors State University, Illinois, USA from 1992 to 1993. His current research focuses on Ubiquitous Computing and applications, Context-aware Middleware, Sensor Operating Systems, Real-Time

Received: March 15, 2011; Accepted: May 2, 2011.