RationalExposure: a Game Theoretic Approach to Optimize Identity Exposure in Pervasive Computing Environments

Feng Zhu
Dept. of Computer Science
University of Alabama in Huntsville
Huntsville, Alabama, USA

Wei Zhu
Intergraph Corporation
Huntsville, Alabama, USA

***Abstract*—In pervasive computing environments, personal information is expressed in digital forms to a great extent. Daily activities and personal preferences may be easily associated with personal identities. Privacy protection is a serious challenge. The fundamental problem is the lack of a mechanism to help people expose appropriate amounts of their identity information. We propose the Hierarchical Identity model. It expresses one's identity information from precise to general. We model privacy exposure as an extensive game. By finding the subgame perfect equilibria in the games, our approach achieves optimal exposure. It finds the most general identity information that a user wants to expose and a service provider accepts.**

Keywords-identity management; game theory; pervasive computing; privacy

I. INTRODUCTION

We expose personal information frequently in our daily tasks. Often, we unnecessarily expose too much information. For example, Bob proves that he is an adult using his driver's license. He unnecessarily exposes his driver's license number, birth date, name, home address, sex, eye color, hair color, and height. Different amounts of exposure may have dramatic difference in sensitivity. If Bob just proves that he is older than a certain age, the verifying party only knows that he is one of billions of adults. In contrast, his driver' license information uniquely identifies him in the world. In pervasive computing environments, we interact with intelligent ambient environments. Much more personal information is expressed in digital forms, is communicated over networks, and is permanently stored. Multiple types of ID cards such as employee ID, driver's license, passport, and credit card are already using embedded processors and can communicate over wireless networks. Proper identity exposure becomes more critical to protect our privacy because identities are associated with our daily activities, preferences, context, and other sensitive information. Campbell et al. warn us that without privacy protection pervasive computing may become a distributed surveillance system [1].

Exposing the appropriate amount to the appropriate parties is challenging. First, we may have many identities associated with our different life roles. To access pervasive services, which we may or may not be familiar with, various identities need to be exposed. Second, users may not be able to make rational exposure choices. Many people's privacy awareness is very limited. For example, on the Internet, they carelessly provide their detailed personal information [2]. Third, unnecessary exposure may be lured, requested, and forced. Stores give discounts to customers who provide their personal information. At the checkout register, we are often asked for our home phone number in order to find our home address and name.

Laws and regulations protect privacy. They, however, only provide protection on data usage [3]. Privacy exposure is often left up to an individual's decision. Once personal information is unnecessarily exposed, it is out of users' control. Langheinrich suggests that privacy should be by design in pervasive computing systems because law makers and sociologists are still addressing yesterday's and today's information privacy issues [3].

Anonymity is an approach to prevent identity exposure [1, 4, 5]. It hides users' identities such that a user is not discernible from other users. Anonymity, however, is limited to applications for which service providers do not need to know any element of users' identities. Several research works use policy-based approaches [6-9]. Users' personal information is not exposed, unless service providers' policies meet users' preferences and policies. The systems require users to have special skills to specify policies. Users might sacrifice their privacy for convenient service access.

We propose our Hierarchical Identity model. It defines a tree structure to express a user's identity information from precise to general. More general identity information represents a larger set of users. If a piece of general identity information can be used in a service access, we protect a user's privacy because he is less likely to be identified.

The main contribution of this paper is that we propose a negotiation approach to find proper identity information for exposure. To the best of our knowledge, this is the first paper that uses game theory for identity exposure in pervasive computing environments. We model the service access and privacy exposure as an extensive game (an economic model). Our approach provides optimal results regardless of a person's negotiation skills. The exposed identity information meets authentication or verification requirements. In the meantime, it is as general as possible. Unlike other approaches, our model provides a good balance between the benefit of service

access and the risk of privacy exposure. In addition, we provide strategies to interact with unfamiliar service providers.

The rest of the paper is structured as follows. In Section II, we discuss related work. Section III illustrates our Hierarchical Identity model. In Section IV, we model privacy exposure as an extensive game. Last, in Section V, we outline our future work and conclude our contribution.

II. Related Work

Role-Based Access Control (RBAC) inspires our work. RBAC has been widely used in computer systems especially in database systems. Permissions are granted to roles and roles are assigned to users. With the role as a level of indirection, users can be easily reassigned from one role to another. There are different RBAC models [10]. Role Hierarchies in RBAC might seem similar to our Hierarchical Identity model. Both models use the hierarchical structures. However, it is important to distinguish between the two concepts. Roles in RBAC have two basic characteristics: role membership and role permissions. From a role, its users and granted permissions can be easily determined. Neither operation is essential in our Hierarchical Identity model. Instead, it is critical to express all identities of a single person in a hierarchy in our model. In RBAC, roles towards the root have less permission, whereas in the Hierarchical Identity model nodes towards the root express more specific identity information.

Recently, location privacy in pervasive computing environments has attracted many researchers' attention. Snekkenes uses lattices to reduce the preciseness of location, identity, time, and speed information in location-based applications [6]. For example, if Bob is in his office at coordinate *(x, y, z)* and was observed outside the office at *(x', y', z'),* he uses a volume from *(x, y, z)* to *(x', y', z')* to represent his position. To achieve partial anonymity for an identity, Bob's and other persons' identities are provided as a set. The idea of providing more general information is similar to ours. Nevertheless, the preciseness of the exposure is left to users to determine. There are two problems in this approach. First, a user may not have enough knowledge to choose the proper anonymous set. Second, a chosen set may not be appropriate and may fail an authentication or an authorization process.

Mix Zone achieves location privacy via anonymity [4]. A short-term anonymous identity is used to represent a person in a geographical area. His identity is among a set of other anonymous identities registered in the same area. When he connects to a location-aware application, he cannot be identified from a set of people in the area.

Gruteser and Grunwald propose *k*-anonymous location information through spatial and temporal cloaking [5]. A person's location is expressed by three intervals, two spatial intervals $[x_1, x_2]$ and *[y1, y2]* and a temporal interval *[t1, t2]*. With the assistance of an anonymity server, a user can be only identified to a service provider as one of the *k* users in an area. The choice of the proper *k* value is left to the users and applications. Our approach helps users and developers to select proper *k* values based on the payoff values.

The research on Internet privacy provides good experience and lessons for pervasive computing. In ten years of evolving, Platform for Privacy Preferences Project (P3P) has provided a standard for websites and users' Internet browsers to communicate with each other about privacy preferences [11]. Using machine readable languages, websites express the data collected and their privacy practices. Internet browsers use user-defined policies to determine the release of identities and other personal information. Besides the machine readable version, an Internet browser may prompt a user for a human-readable version. If sensitive personal information will be collected, a user can make decisions. In pervasive computing environments, a standard for specifying a machine readable version and a user friendly version is important for heterogeneous devices to communicate with each other.

Cranor and Reagle's "buckets" approach [12] and the Privacy AWareness System (*pawS*) [8] are P3P-based privacy awareness systems. Because of the policy complexity, these P3P-based approaches are believed not to be the solution for pervasive computing environments [13].

Leonhardt and Magee use high level policies for access control and privacy protection in pervasive computing environments [7]. Their approach is based on two classical security models: Lampson's access matrix and the Bell and LaPadula's (BLP) security labels. Similarly, their approach suffers from usability issues.

Confab is a privacy sensitive framework for pervasive computing environments [9]. It enables application developers to enforce policies, send privacy notifications, and manipulate private data. In addition, it enables users to control their privacy information in three interaction patterns: optimistic (share information with others), pessimistic (detect privacy information abuse), and mixed-initiative (request users to make exposure decisions). Confab's data model is used to represent context information including locations, activities, and services. Confab and other frameworks [1, 13] do not provide sophisticated methods to help users make decisions. Our approach complements the frameworks and improves the usability by helping users to make rational decisions.

Identity federation is an approach to address the usability of identity management [14, 15]. For web applications, cross domain single sign-on systems have been developed and are evolving rapidly. Behind the scenes, service providers and identity providers link a user's accounts on different websites and share related attributes with each other. Therefore, a user needs to log in only once for accessing protected websites across administrative domains. Although identity federation improves usability, the usage of a single identity for various applications across different domains may sacrifice privacy. The identity providers are aware of all websites with which a user authenticates. Furthermore, the identities in federation systems are attractive targets for identity theft, and need to be properly protected [16].

Automated trust negotiation systems enable unfamiliar parties on the Internet to establish trust and exchange identity

information [17, 18]. During trust negotiation, a user and a service provider in turn request the other party's identity information and provide their own identity information. For example, the user may require the service provider to provide a certificate from a Better Business Bureau, and then a service provider may require credit card information from a user. The systems protect sensitive attributes by exchanging one piece of information in a message. In a round if both the user's and a service provider's requirement are met, the process proceeds. Since the main purpose of the systems is to establish trust, a party supplies whatever the other party requires. Thus, the systems may not be able to protect users' privacy and find optimal exposure strategies. Automated trust negotiation systems may also be integrated with identity federation systems [19]. Combination of identity federation and trust negotiation improves usability.

In our previous work [20], we propose a progressive and probabilistic exposure approach to protect privacy among familiar users and service providers. A user and a service provider exchange their identities and other sensitive data over multiple rounds. Each round, partial information (several encoded bits) is exchanged. If there is any mismatch in the identity verification in any round, the interaction stops. Since only part of their identities and other sensitive information are exchanged, the other parties are uncertain about the information received.

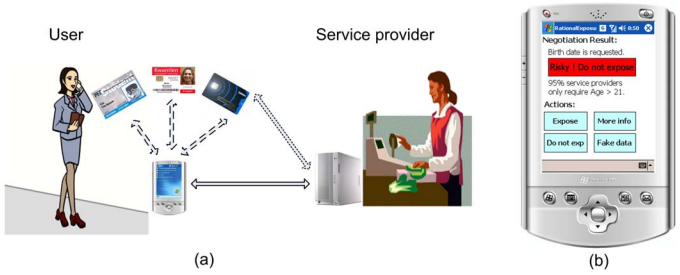

Figure 1 (a) Interaction among a service provider, a user, and digital identity tokens. (b) RationalExposure notifies a user of potential identity exposure.

III. THE HIERARCHICAL IDENTITY MODEL

We propose to aggregate all digital identities of a person on a single device. Handheld devices such as smartphones are good candidates because people always carry them. Via the devices, users can manage digital identities and interact with service providers as shown in Figure 1(a). We show a snapshot of the RationalExposure UI prototype in Figure 1(b). A message summarizes the requested identity information. We use simple color schemes, green, orange, and red, to notify users whether an exposure is risky. Statistics are also given to assure users that their choices are rational. Users may choose the suggested actions, override the system suggestions, or request for more information about exposures.

The major purpose for the Hierarchical Identity model is to facilitate proper identity exposure. The model expresses identities in different levels of details. For different activities and applications with different authentication and verification requirements, the proper identities or identity elements can be provided. For example, Bob proves that he is a resident of a city to access the city zoo at a discounted price. As a resident, he gains access to a community pool. Note that precise exposure (Bob's home address) meets both authentication requirements, but privacy is not well protected.

We express an identity and its elements in the hierarchical structure as shown in Figure 2 (a). From bottom to top, identity elements become more and more specific and eventually become the person's identity. An identity element (or a set of identity elements) is more general, if more people have the same identity element (or a set of elements). Thus, a user is less likely to be identified by exposing more general identity information. A piece of identity information is precise, if the number of persons in the set is small or even unique. Therefore, the person is more likely to be identified.

All identities of a person are aggregated together in a hierarchical tree structure as shown in Figure 2(b). The top-level node represents a person. The second-level nodes are the identities. Other nodes are identity elements. Some personal information may appear more than once in a hierarchy. For example, date of birth is under both driver's license and passport: one may use either element as proof. The following definition formalizes the above description.

Definition 1 The Hierarchical Identity Model has the following properties:

A person has a set of identities, $\{I_i\}_{i \in N}$.

I_n may have a set of children, $\{I'_j\}_{j \in N}$. I'_j may be an identity, an identity element, or a set of identity elements.

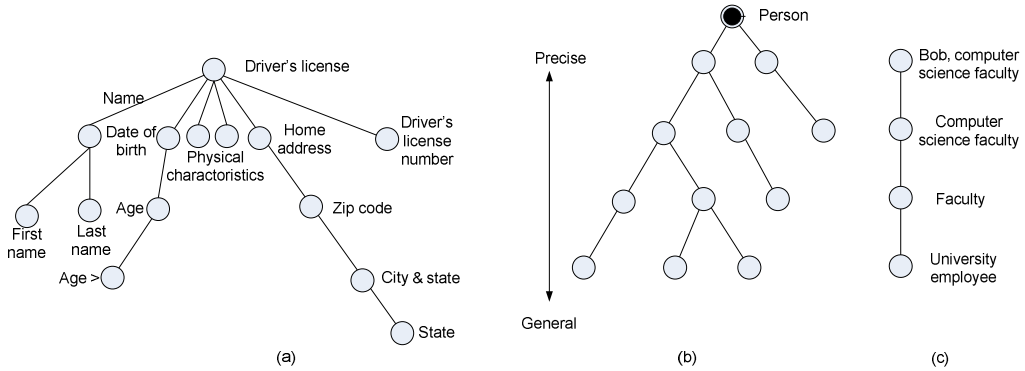

Figure 2 (a). An identity and its elements. (b) The Hierarchical Identity model. (c) Another Hierarchical identity sample.

If I'_m is a child of I'_n, the set of people that I'_m represents, $\{P_a\}_{a \in N}$, and the set of people that I'_n represents, $\{P_b\}_{b \in N}$, has a partial order, $\{P_b\} \leq \{P_a\}$. That is I'_m is at least as general as I'_n.■

The main benefit of the Hierarchical Identity model is that along the hierarchical tree structure from top to bottom, more and more general identity information can be found to represent a person. We use two approaches to build the Hierarchical Identities. First, we directly use the elements of an identity as shown in Figure 2. For example, age information may be used to prove adulthood. Zip code may be used to show one's residency of a city. Second, we may connect different identities as parents and children as shown in Figure 2(c). A child identity is more general than the parent identity.

IV. FINDING THE PROPER IDENTITY TO EXPOSE

In pervasive computing environments, we interact with many service providers. In the meantime, the intelligent devices that we wear and carry may interact with service providers' intelligent devices. We may or may not be familiar with the service providers. When we request to access a service, a service provider may ask for our identity. Should we give the identity? Or should we provide some elements of the identity? In traditional computing environments, this usually is not a problem. Users provide user names and passwords to access computers. Users and service providers agree on the exposure. In our current daily life, people make their identity exposure decisions. For example, Bob may be asked for his driver's license card at a checkout register. Should he give it?

We start our discussion with this simple example and model it as an extensive game. Then, we describe how we find the optimal solutions. Next, we define the payoff functions for users and service providers. Afterward, we discuss the strategy to interact with unfamiliar service providers. Last, we illustrate how extensive games are built from the Hierarchical Identity model. We assume that a user interacts with a service provider in the vicinity and the user wants to access the service and trusts the service provider.

A. Identity Exposure as an Extensive Game

We model the interaction between a user and a service provider as an extensive game. An extensive game means that a user and a service provider take turns to make decisions and take actions. We use a supermarket checkout process as our scenario. Bob provides his credit card information via his smartphone. The store's computer asks for his digital driver's license. Then, Bob makes a decision on whether to expose his driver's license information. Afterward, the store makes a decision.

In the following discussion, we use "Bob" to refer to him and his smartphone that aggregates all his digital identities. We use "the store" to refer to the store and the computer at a checkout kiosk. Figure 3 starts with Bob's turn to make a decision. Bob may quit the service; he may accept the request and give his driver's license information; or he may propose other identity information. Let's suppose that Bob proposes to provide only his name. Now, it is the store's turn to make decisions. Suppose the store either aborts the checkout service or finishes the transaction. Therefore, there are five outcome cases. In case 1, Bob provides only his name, and the store finishes the transaction. In case 2, he provides only his name, and the store aborts the process. In case 3, he quits the checkout process. In case 4, he gives his driver's license information, and the store aborts the transaction. In case 5, he gives his driver's license information, and the transaction finishes.

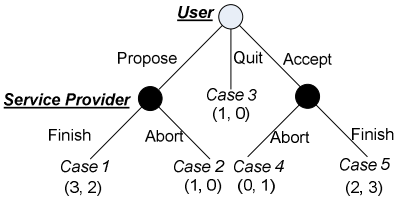

Figure 3. An extensive game between a user and a service provider.

In Figure 3, we also show the user's and the service provider's payoffs in the parentheses of each outcome. The first number indicates the user's payoff, whereas the second number shows the service provider's payoff. The numbers represent the preference of a user and a service provider in an ordinal order. In this example, Bob's preference is {case 1} > {case 5} > {case 3, case 2} > {case 4}. That is, Bob prefers to get the same service by giving only his name. He also prefers to get the service than not get it, even at the price of giving his driver's license information. His least preferred outcome is that he gives his driver's license information without getting the service. It is possible that two or more outcomes have the same payoff value. This means that the user or the service provider is indifferent about the outcomes. For example, case 2 and case 3 have the same payoff value for Bob. Thus, he may choose either outcome. We formally define the identity exposure game as follows.

Definition 2 An identity exposure game has the following components:

Two players, *u* and *s* – a user and a service provider.

A set of outcomes, $\{o_i\}_{i \in N}$: in our example, there are the five outcomes: *(Propose, Finish), (Propose, Abort), (Quit), (Accept, Abort), and (Accept, Finish).*

Each player chooses and takes one action $a_i^j \in \{a_i\}_{i \in N}$ at step j. Note that at different steps, the actions may be different.

The user and the service provider have their preferences, which are represented by payoff values. For example, the payoff values for the user and the service provider in case 1 are: $p_u(Propose, Finish) = 3$, *and* $p_s(Propose, Finish) = 2$, respectively. ■

B. Finding the Optimal Identity Exposure Strategy in an Extensive Game

We assume that users and services providers are rational. That is they will choose outcomes according to their preferences. A chosen action leads to an outcome. The outcome is at least as good as any other outcome. The payoff functions that we will define in subsection C are used to determine the preferences.

In this subsection, we assume that the outcomes are known. We also assume that the user's and the service provider's preferences are known. It is very likely because the user and the service provider may learn the outcomes and preferences from their previous interactions. Even if a user and a service provider interact with each other for the first time, the outcomes, actions, and preferences may still be known because a user may have interacted with many other similar service providers for the same service.

Given an extensive game, we start from the bottom of the tree and consider the subtrees of height 2. We walk through our example in Figure 3 to demonstrate the game. There are two subtrees for which the service provider makes choices. In the left subtree, the user has proposed not to give the driver's license. The service provider may choose "*Abort"* or "*Finish"*. "*Abort"* gives the service provider a payoff of 0, where as "*Finish"* gives him a payoff of 2. Thus, he will choose "*Finish"*. Similarly, in the right subtree, the service provider will choose "*Finish"* with a payoff of 3. He provides the service and also acquires the user's driver license.

After the selections of optimal actions for all subtrees, we move up one level of the tree and consider the subtrees with height of 1. In our example, there is one subtree for which the user makes choice. He may choose "*Propose", "Quit",* or *"Accept"*. From our discussion in the previous paragraph, if the user chooses "*Propose"*, the service provider will choose "*Finish"*. The user's payoff is 3. If the user chooses "*Accept"*, the service provider will choose "*Finish"*. The user's payoff is 2. If the user chooses "*Quit"*, his payoff is 0. Therefore, the user's best choice is "*Propose"*.

The process of finding the best choices in the subtrees continues, until the root of the tree is reached. This process is known as backward induction in Game Theory [21]. In our example, we have reached the root of the tree. So, the user chooses "*Propose"*, and then the service provider chooses "*Finish"*. It is the subgame perfect equilibrium, the optimal choices for the user and the service provider. Using backward induction, we always find a subgame perfect equilibrium in an identity exposure game. We prove the following proposition.

Proposition 1 In an identity exposure game, an optimal exposure can always be found by using backward induction. (There is an assumption in the proof that an exposure game has finite levels and each subtree has finite branches. We will prove the assumption in subsection E.)

Proof. We use mathematical induction to prove the proposition.

Base: For a tree (game) of height 1, either the user or the service provider makes choice. Without loss of generality, we assume the user makes the choice. The user compares the payoffs of all the outcomes and can select one outcome that is at least as good as every other outcome. It is the subgame perfect equilibrium.

Inductive step: Assume that we can find a subgame perfect equilibrium of a tree with height of n or less using the backward induction. Then for a tree with height of $n+1$, we consider each child of the root as a subtree with height of n or less. Hence for each subtree, we find a subgame perfect equilibrium. Now, we move the root. The best payoff value of each child is known. This can be considered as a tree of height 1. Therefore, the user (or the service provider) can choose the optimal outcome. ■

C. Payoff Functions

We believe that users want to protect their privacy while they access services. Service providers want to acquire more users' personal information while they provide services. We define their payoff functions as follows.

Definition 3 A user's payoff function and a service provider's payoff function at each node are, respectively:

$$P_u = Access\ service \times Weight_u - User's\ exposure \quad (1)$$

$$P_s = Provide\ service \times Weight_s + User's\ exposure \quad (2)$$ ■

"Access service" and *"Provide service"* in equation (1) and (2) are the benefits that a user and a service provider get, respectively. Note that they may not be the same value for the user and the service provider. For different services, users and service providers may give different "*Weights"* to the services. For "*User's exposure*", a user obtains negative benefit by exposing his personal information. On the contrary, a service provider always obtains positive benefit. He might sell users' personal information and their preferences, or he might improve his services by using users' information data.

From a user's perspective, "*Access service* × $Weight_u$*"* brings him constant amount of benefit for a service, which is independent of his personal information exposure. Until at a certain point (threshold), the user may not want to expose more personal information and quit the negotiation. Then, the benefit becomes zero as shown in Figure 4(a). The "*User's exposure"* component in equation (1), however, keeps reducing the benefit as "*User's exposure"* increases. Thus, the payoff function that adds the two components together has two segments as shown in Figure 4(b).

Similarly, from a service provider's perspective as shown in Figure 4(c), *"Provide service* × $Weight_s$*"* brings him a constant benefit, unless the service provider does not get enough identity information from a user and aborts. Then, the benefit is zero. The "*User's exposure"* component in equation (2) is a non-decreasing function. The more the user exposes the more benefit a service provider receives. The service provider's payoff function is shown in Figure 4 (d) by adding the two components.

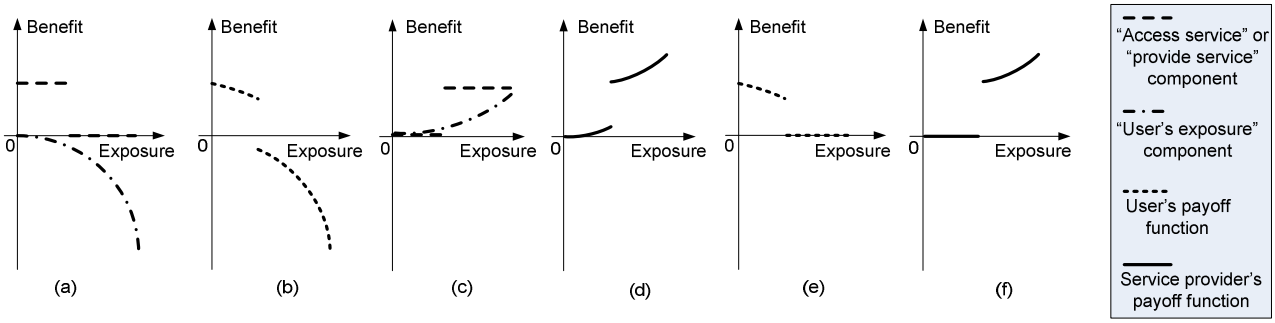

Figure 4(a). Two components of the user's payoff function. (b). User's payoff function. (c). Two components of the service provider's payoff function. (d). Service provider's payoff function. (e). The user's modified payoff function. (f). The service provider's modified payoff function.

From the two payoff functions shown in Figure 4(b) and (d), the game does not seem to be fair to the user. The user has risks to expose his personal information without getting the service. On the contrary, the service provider's payoff function is always positive.

To address the problem, the user plays a modified game. Instead of giving his identity information during negotiation, the user proposes what identity information he will provide. Only after the user and the service provider agree on the identity information and service access does the user expose his identity elements. If the negotiation is unsuccessful, the user will not expose any identity information. Therefore, the user's and the service provider's payoff functions are changed as shown in Figure 4(e) and (f), respectively. Both the user and the service provider have non-negative payoff functions.

D. Strategies to Interact with Unfamiliar Service Providers

When interacting with an unfamiliar service provider, an essential question is whether a user and a service provider will reach an agreement. Different services offer a user different amounts of benefit. The payoff function may be higher or lower. For different services, a user may be willing to exposure different amount of identity information. Similarly, service providers may weigh their benefit differently and accept different types of identities.

The two boundary cases are shown in Figure 5(a) and (b). In Figure 5(a), a user provides the identity that the service provider originally requested. This is the maximum identity information that a service provider asks. In Figure 5(b), a user proposes minimal exposure that he is willing to give and the service provider accepts. Beyond the two boundary cases, a user and a service provider cannot reach any agreement. Therefore, the user does not gain the access to the service and he does not expose his identity information as shown in Figure 5(c). Between the two boundary cases, shown in Figure 5(d), a user and a service provider can reach an agreement on the identity exposure for a service access.

Our strategy is simple. During the negotiation with a service provider, the user proposes the minimal identity information and gradually increases his exposure. A rational service provider requests maximum identity information and gradually decreases his request. The other way to interpret the strategy is shown in Figure 5 (d). The user exposes more and his payoff values moves down along his payoff function curve from left to right. The service provider requests less and his payoff value moves down along his payoff function curve from right to left.

We use Figure 6 to illustrate a more detailed analysis. The game starts after a user receives a service provider's identity request. If a user "*Accepts"* the request, it is the case shown in Figure 5(a). Otherwise, the rational user will choose other actions. If the user chooses to "*Quit"*, the user does not want to negotiate and use the service. If the user chooses to "*Propose"*, he proposes the minimal identity information. If the service provider "*Accepts"* the proposal, it is the case shown in Figure 5(b). The proposal may be that the user does not expose any identity information. If the service provider "*Aborts"* the service, the service provider's last request is the minimum identity information that he needs. Or, the service provider does not want to negotiate. If the service provider "*Requests"* more identity information, he decreases his requirement as little as possible. Note that he cannot request the same identity information. The user and the service provider may keep negotiating, until they agree on the exposure or one party quits. If one party quits, the user's maximum exposure is still less than the service provider's

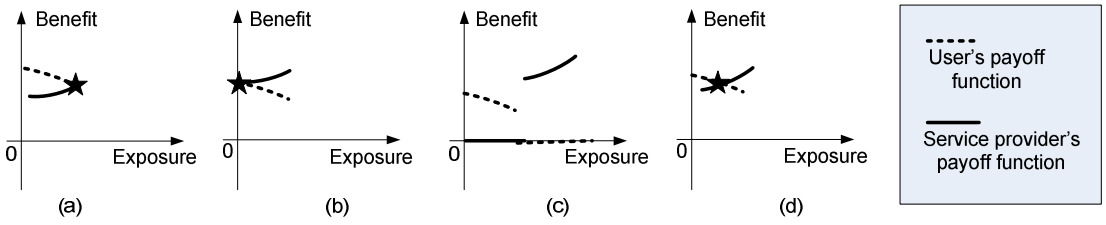

Figure 5. User and service provider interaction cases. (a-b) Two boundary cases that the user and the service provider find agreements. (c). The user and the service provider quit negotiation. (d). The user and the service provider find agreement.

minimum requirement. This is the case shown in Figure 5 (c). If the two parties agree on the exposure, we find an agreement, which is the case shown in Figure 5(d).

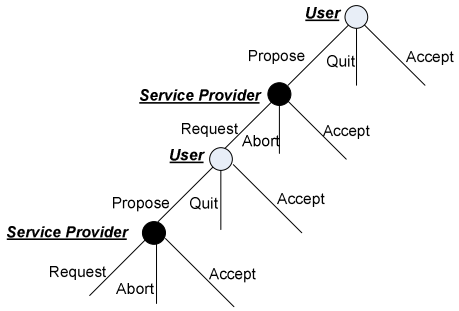

Figure 6. The user's strategy to interact with an unfamiliar service provider.

E. Using Hierarchical Identity Model to Build an Exposure Game

Each type of services has its exposure game. If there is no available tree to represent the game, a user builds the tree when he interacts with a service provider for the first time. The service provider asks for identity information. The user builds the top level branches. For example, when a user is asked for home address as shown in Figure 7, he has the choices of "*Propose*", "*Quit*", and "*Accept*". If the requested identity information in his Hierarchical Identity is found, all the children in the subtree are traversed. The most general identity information will be used as the user's proposal.

The service provider may take his actions for each action that the user takes (except "*Quit*"). He may choose to "*Request*" more or other information, "*Abort*" the service, or "*Accept*" the proposal and provide the service. If the negotiation is along the path in the Hierarchical Identity from the original requested identity to the most general identity the user proposes, the tree that represents the exposure game looks like the tree in Figure 6.

If multiple leaf nodes are found in the Hierarchical Identity, those identity elements may be alternative identity elements to propose as shown in Figure 7 (dotted line). Rules or statistics may be used to determine which identity to propose. For example, eye color is rarely used and should be a lower priority identity element to propose. If several identities are good choices, a user may play a mixed game, whereby he tries different identities in different games.

During the negotiation, if the service provider requests an identity or an identity element that is not in the subtree of the originally requested identity, then the user searches the new requested identity in the Hierarchical Identity and finds the most general identity element in its children. The user's proposal will be the top level branch (an alternative proposal) as shown in Figure 7 (dotted line). If the service provider's request is in the subtree, but it is not on the path of the most general identity information to the originally requested information, then the common parent of the two is found. In Figure 7, we show that an alternative proposal (dashed line) is added at the third level of the tree.

A user and a service provider make their proposals and requests. The orders of requests express their preferences. From the service provider's request order, the user finds the service provider's preferences in the games. Similarly, the service provider finds the user's preferences.

An exposure extensive game may be dynamic. The tree that represents the game might expand as the user interacts with a service provider multiple times or after the user interacts with different service providers. Moreover, a user's identity may expire, in which case the user needs to prune the trees of extensive games that use the expired identity. Similarly, after a new identity is acquired, a user may update his games with the new identity.

The above discussion about a tree representing an exposure game is general. A user may have many different choices of identity information to expose at any step during an interaction. He may propose different identity elements or different identities to a service provider. The service provider may not accept the user's proposal and request new information. The process may continue between a user and a service provider. Such process includes any identity exposure interactions between a user and a service provider. Is an interaction guaranteed to finish? We prove the following proposition to show that a game always finishes.

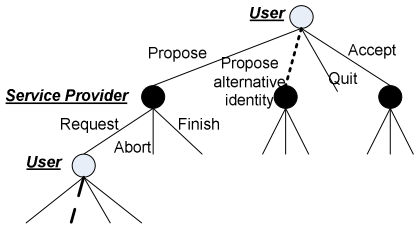

Figure 7. Build an identity exposure extensive game from the Hierarchical Identity model.

Proposition 2 A tree representing an exposure game has finite levels and finite branches.

Proof. First, we prove that at any level of a tree, there are a finite number of branches. If it is a user's turn to make a decision, he has finite number of actions to choose from. He may choose "Accept", "Quit", or "Propose". The number of alternative proposals is bounded by the finite number of leaf node identity elements in the hierarchy. Therefore, the user creates a finite number of branches for each node in which he makes a decision. Similarly, if it is a service provider's turn to make a decision, he may "Accept" a user's proposal, "Abort" the service, or "Request" other identity information from a finite set of possibilities. Thus the service provider creates finite numbers of branches for each node in which he makes a decision.

Second, we prove that a tree has finitely many levels. A user has a finite number of identities. In the extreme case that the service provider requests all identity information (n pieces of identities and identity elements) that the user has and the user replies, the tree's height will be up to 2*n. If the service provider asks some identity information that he has already asked, the user quits. If the service provider asks for some identity information that the user does not have, the user may quit or propose alternative identity information (the user may

propose only the identity information that has not yet been proposed). Therefore, the tree will not grow infinitely. Although a user may acquire new identities in the future, he has finite number of identities at any given time. ■

V. Conclusion and Future Work

We model identity exposure between a user and a service provider as an extensive game. The user and service provider negotiate on the identity information that the user wants to provide and the service provider accepts. We express one's identities and their elements in our Hierarchical Identity model. It facilitates the use of identities and elements of identities in a general manner. Therefore, users are less likely to be identified. We provide strategies to interact with unfamiliar service providers and learn their preferences. We present formal definitions and prove the properties of our model.

To help us understand user reactions and their acceptance of the game theoretic models, we will conduct a three-step process in usability studies. In the first step, participants input their identity information into PDAs and they will be trained to use the RationalExposure model on the PDAs. In the second step, we mimic purchases in stores and participants response to identity requests via the PDAs. In the third step, participants will complete a survey on their experience in the study. We will identify the effectiveness of the rational choices, users' perception, and identities and identity elements that they believe are risky to expose. We will ask them to compare their experiences of using traditional ID cards with the digital forms on PDAs.

In this paper, we do not address special exposure situations. For example, in a medical emergency, Bob wants to expose as much patient identity information as possible. So, medical emergency personnel may find his medical history quickly. These special exceptions require a special implementation. Our model is capable of expressing the payoff functions in this case as well. The "*User's exposure"* component is expressed as non-decreasing functions. Thus, both Bob and medical personnel acquire better payoffs, when Bob exposes more information. It can be proved that negotiations in these cases still converge.

At present, a tree structure is used to express the hierarchical identity information. Ideally, the same identity elements need to be easily found, so that alternative identity information may be proposed and exposed. A lattice might be used to express the identity information. Nevertheless, identities are usually acquired from different identity providers and might have different signatures. Using a single node in a lattice to represent two identity elements from two identity providers will complicate the data structure and the rational exposure algorithms. We will improve the tree-based data structure to facilitate connections between the same identity elements and simplify search for them.

In this paper, we assume that a user interacts and trusts a service provider in the vicinity. If the assumption does not hold, trust establishment needs to be addressed before or during the negotiation of identity exposure.

Acknowledgment

The authors are grateful to Dr. Danny Soroker for helping them to greatly improve this paper. The authors thank Dr. Sandra Carpenter for fruitful discussions and thank the anonymous reviewers for their valuable comments.

References

[1] R. Campbell, J. Al-Muhtadi, P. Naldurg, G. Sampemane, and M. D. Mickunas, "Towards Security and Privacy for Pervasive Computing," presented at International Symposium on Software Security, Tokyo, Japan, 2002.

[2] E. Dyson, "Privacy Protection: Time to Think and Act Locally and Globally," First Monday, vol. July, 2006.

[3] M. Langheinrich, "Privacy by Design - Principles of Privacy-Aware Ubiquitous Systems," in Ubicomp 2001 Proceedings, volume 2201 of Lecture Notes in Computer Science, 2001, pp. 273–291.

[4] A. Beresford and F. Stajano, "Location Privacy in Pervasive Computing," IEEE Pervasive Computing, vol. January-March, pp. 47-55, 2003.

[5] M. Gruteser and D. Grunwald, "Anonymous Usage of Location-Based Services Through Spatial and Temporal Cloaking," presented at 1st international conference on Mobile systems, applications and services, New York, NY, 2003.

[6] E. Snekkenes, "Concepts for Personal Location Privacy Policies," presented at 3rd ACM conference on Electronic Commerce, Tampa, Florida, USA, 2001.

[7] U. Leonhardt and J. Magee, "Security Considerations for a Distributed Location Service," Journal of Network and Systems Management, vol. 6, pp. 51-70, 1998.

[8] M. Langheinrich, "A Privacy Awareness System for Ubiquitous Computing Environments," presented at 4th international conference on Ubiquitous Computing, Göteborg, Sweden, 2002.

[9] J. Hong and J. Landay, "An Architecture for Privacy-Sensitive Ubiquitous Computing," presented at 2nd international conference on Mobile systems, applications, and services, Boston, MA, 2004.

[10] R. Sandhu, E. Coyne, and C. Youman, "Role-Based Access Control Models," IEEE Computer, 1996.

[11] L. Cranor, B. Dobbs, S. Egelman, G. Hogben, J. Humphrey, M. Langheinrich, M. Marchiori, M. Presler-Marshall, J. Reagle, M. Schunter, D. A. Stampley, and R. Wenning, "The Platform for Privacy Preferences 1.1 (P3P1.1) Specification," W3C, November 2006.

[12] L. F. Cranor and J. Reagle, "Designing a Social Protocol: Lessons Learned from the Platform for Privacy Preferences Project," in Telephony, the Internet, and the Meda, D. Waterman, Ed.: Mahwah: Lawrence Erlbaum Associates, 1998.

[13] A. Soppera and T. Burbridge, "Maintaining privacy in pervasive computing — enabling acceptance of sensor-based services," BT Technology Journal, vol. 22, pp. 106-118, 2004.

[14] B. Ferg, B. Fitzpatrick, C. Howells, D. Recordon, D. Hardt, D. Reed, H. Granqvist, J. Ernst, J. Bufu, J. Hoyt, K. Turner, M. Scurtescu, M. Atkins, and M. Glover, "OpenID Authentication 2.0," http://openid.net/specs/openid-authentication-2_0.html, December, 5 2007.

[15] N. Ragouzis, J. Hughes, R. Philpott, and E. Maler, "Security Assertion Markup Language (SAML) V2.0 Technical Overview," OASIS Open, http://www.oasis-open.org/committees/documents.php?wg_abbrev=security, 9 October 2006.

[16] A. Bhargav-Spantzel, A. C. Squicciarini, and E. Bertino, "Establishing and Protecting Digital Identity in Federation Systems," Journal of Computer Security, vol. 14, pp. 269-300, 2006.

[17] M. Winslett, T. Yu, K. E. Seamons, A. Hess, J. Jacobson, R. Jarvis, B. Smith, and L. Yu, "Negotiating Trust on the Web," IEEE Internet Computing, vol. November-December, pp. 30-37, 2002.

[18] P. Bonatti and P. Samarati, "Regulating service access and information release on the Web," presented at 7th ACM conference on Computer and communications security, Athens, Greece, 2000.

[19] A. Bhargav-Spantzel, A. Squicciarini, and E. Bertino, "Trust Negotiation in Identity Management," IEEE SECURITY & PRIVACY, vol. March/April, pp. 55-63, 2007.

[20] F. Zhu, W. Zhu, M. Mutka, and L. Ni, "Private and Secure Service Discovery via Progressive and Probabilistic Exposure," IEEE Transactions on Parallel and Distributed Systems, vol. 18, pp. 1565-1577, 2007.

[21] M. Osborne, An Introduction to Game Theory. New York: Oxford University Press, 2004.