

# An Identifiability-Based Access Control Model for Privacy Protection in Open Systems

Keith Irwin  
North Carolina State University  
kirwin@ncsu.edu

Ting Yu  
North Carolina State University  
yu@csc.ncsu.edu

## ABSTRACT

We argue that in open systems one's private information disclosure needs to be dynamically controlled based on both its sensitivity and the possibility that a user's identity is revealed. Then we propose an identifiability-based access control scheme, which not only properly control users' private information, but also increase users' access to web services.

## Categories and Subject Descriptors

K.6.5 [Management of Computing and Information Systems]: Security and Protection

## General Terms

Security

## Keywords

Identifiability-based access control

## 1. INTRODUCTION

Attribute-based access control (ABAC) is a new approach to trust management in open environments [6]. Unlike traditional identity-based access control, access control decisions in ABAC are made based on requesters' attributes, which are demonstrated through the disclosure of digital credentials. Since users' credentials often contain private information, privacy protection becomes a key issue in ABAC. This is especially true when ABAC is used to establish trust between strangers, who have no pre-existing knowledge about each other. Most existing approaches treat credentials or attributes as basic protection units and have access control policies to *statically* control their disclosures, without considering the context of trust establishment.

Practical pseudonym credential systems have recently been proposed [2, 4]. Through the use of pseudonyms, such credential systems allow users to prove their attributes (e.g., one's age, zip code, etc) without disclosing their identities. Further, two sessions of trust establishment cannot be linked even though the same credential is used in both sessions.

Pseudonym credential systems have important implications on privacy protection in ABAC. On one hand, since a user's identity is not revealed when pseudonym credentials are disclosed, the release of seemingly sensitive information

may not really be sensitive. On the other hand, though different sessions of trust establishment cannot be linked, the attributes disclosed in one session are all about one single user. By combining those attributes with public knowledge, the user's anonymity may be compromised. For example, by knowing a user's 5-digit zip code, gender, and birthdate, it has been shown that, with high probability, the user can be uniquely identified [7].

Clearly, how well a user can be identified affects the sensitivity of an attribute disclosure. Therefore, we need policies that take into account both the sensitivity and the level of identifiability, and *dynamically* control attribute disclosures.

## 2. AN OVERVIEW

Suppose a service provider Bob requires Alice to prove that her attributes satisfy a constraint  $q$ . If  $q$  requires the disclosure of sensitive information, then Alice would respond with an access control policy for  $q$ . To dynamically generate the access control policy, we need to consider three major information sources. The first is a *public database*, which contains publicly available information of individuals. The second is *inference rules*, which are well-known correlations between attributes. The third information source is a set  $\mathcal{R}$  of *known properties* of Alice, which Alice has already proven to Bob during the session of an ongoing trust establishment. Note that  $\mathcal{R}$  includes not only the exact value of some of Alice's attributes, but also more general constraints that Alice has satisfied (e.g., age  $\geq 25$ ). In essence, the public database, inference rules and the known properties of Alice form a *context* for  $q$ .

The *identifiability* of  $q$  is calculated based on its current context, which indicates how well one can narrow the range of Alice's unique identity if  $q$  is proven to be true. Intuitively, the larger  $q$ 's identifiability (i.e., the smaller the range of Alice's unique identity), the more restrict  $q$ 's access control policy needs to be. Note that even if  $a$  may only involve attributes not in the public database, due to the existence of inference rules, the disclosure of  $q$  may still help one recover Alice's identity.

Another factor related to  $q$ 's access control policy is  $q$ 's intrinsic sensitivity. It indicates how much Alice cares about letting others know  $q$ . The more sensitive  $q$  is, the more restrict  $q$ 's access control policy should be. Under the current context,  $q$ 's identifiability and its sensitivity together determine  $q$ 's policy. For example, even if  $q$  is very sensitive, as long as  $q$ 's identifiability is low, its access control policy may not need to be very restrict.

### 3. THE MODEL

The public database  $\mathcal{DB}$  correlates users' unique identity,  $\iota$ , with their attributes  $A_{PII}$ , whose values are publicly available. In practice, no such single database is likely to actually exist. Instead, we think of it as the totality of the information that could be collected about all users based on existing non-anonymous sources. There is also a second set of attributes  $A_U$ , which includes those attributes that do not appear in  $\mathcal{DB}$ . For simplicity, we assume the values of attributes in  $A_U$  are only known by the user.

An inference rule is a means of learning information about attributes given some existing knowledge of other attributes. An inference rule can be modeled as a logical implication of the form  $P \rightarrow Q$  where  $P$  and  $Q$  are constraints on attributes. We use  $\mathcal{IR}$  to denote all the possible inference rules, which not only includes rules explicitly defined but also those that can be derived from others. In other words,  $\mathcal{IR}$  is the closure of inference rules under logical inferences [1, 3]. Given  $\mathcal{IR}$ , we enhance the known property  $\mathcal{R}$  of Alice such that it includes not only those explicitly proven properties, but also those derived when  $\mathcal{IR}$  is applied.

A lattice can be defined on attribute constraints based on the amount of information revealed. Specifically, given two constraints  $q_1$  and  $q_2$ ,  $q_1$  should dominate  $q_2$  if and only if  $q_1$  requires to reveal more information than  $q_2$ . To be complete, we need to consider inference rules when comparing two attribute constraints. Formally, we define  $q_1 \leq q_2 \leftrightarrow (q_1 \wedge \mathcal{IR} \rightarrow q_2)$

Identifiability is a measure of how specifically an attacker can narrow down the identity of a user given the attributes disclosed. We define an identifiability function  $I$ , whose inputs are the known properties  $\mathcal{R}$ , the public database  $\mathcal{DB}$  and the set of inference rules  $\mathcal{IR}$ . The identifiability function outputs a fraction in the form of  $1/n$ , where  $n$  is the size of the output set when  $\mathcal{R}$  is converted into a database query on  $\mathcal{DB}$ .  $n$  is similar to the quantity  $k$  in  $k$ -anonymity [7].

The second function we define is the sensitivity function  $S$ , which represents the user's concern about information being revealed assuming that his identity is known. The sensitivity of some information changes depending on what other information has been disclosed. So we define sensitivity to be a function from the lattice of attribute constraints to an implementation-specific sensitivity lattice. If one attribute constraint reveals strictly more information than another, then clearly it must be more sensitive:  $R_1 \leq R_2 \rightarrow S(R_1) \leq S(R_2)$ . One may ask, given that the space of all attribute constraints forms a lattice, what is the purpose of  $S$ ? The answer is that  $S$  allows us to potentially reduce to a simpler, more manageable lattice. Under the attribute constraint lattice, constraints dealing with any two different categories are always incomparable, but the actual sensitivity of information is probably not. The user can likely make judgments about whether, for example, their medical history or class enrollments is more sensitive, and we can use this to guide our access control policies.

We are now ready to discuss a function,  $F$  which formulates a security policy by considering the implications of the potential data release when combined with already released data both in terms of sensitivity and the resulting identifiability. Formally, we describe this as a function which takes as input the results of the identifiability function and the sensitivity function and returns an access control policy. We

have defined the set of access control policies to be a logical formula over attributes of the other party. In other words,  $F$  determines what credentials they must show the user in order for the user to release their credentials.

The definition of  $F$  will be implementation specific. Rather, we discuss the properties that it should have. First,  $F$  should never decrease in stringency as the sensitivity of the information increases and never increase in stringency as the sensitivity of the information decreases.

$$s_1 \leq s_2 \rightarrow F(s_1, i) \not\prec F(s_2, i) \forall s_1, s_2, i$$

Second,  $F$  should have a similar property regarding identifiability. The reason that the properties are written in terms of  $\not\prec$  and  $\prec$  rather than  $\leq$  and  $\geq$  is that many policies are incomparable in the request lattice. These two properties will ensure that the function behaves in a sensible manner, never requiring strictly less information in response to a more sensitive query or to a query which compromises the user's identity more, but they still leave quite a bit of room for the user to define her own policies.

### 4. CONCLUSION

Due to the existence of publicly available knowledge and inferences, one's privacy may still be compromised even if powerful pseudonym credential systems are used. Thus, privacy policies need to be designed dynamically according to the progress of online interactions. To do so, identifiability and information sensitivity both need to be considered. Many interesting issues remain, e.g., transaction linkability and probabilistic rules. Detailed discussion of these issues can be found in the full version of the paper [5].

### 5. REFERENCES

- [1] J. Biskup and P.A. Bonatti. Controlled Query Evaluation for Known Policies by Combining Lying and Refusal. In *International Symposium on Foundations of Information and Knowledge Systems*, Salzau Castle, Germany, February 2002.
- [2] S. Brands. *Rethinking Public Key Infrastructures and Digital Certificates: Building in Privacy*. The MIT Press, 2000.
- [3] A. Brodsky, C. Farkas, and S. Jajodia. Secure Databases: Constraints, Inference Channels, and Monitoring Disclosures. *IEEE Transactions on Knowledge and Data Engineering*, 12(6), November 2000.
- [4] J. Camenisch and E.V. Herreweghen. Design and Implementation of the *Idemix* Anonymous Credential System. In *ACM Conference on Computer and Communication Security*, Washington D.C., November 2002.
- [5] K. Irwin and T. Yu. An Identifiability-based Access Control for Privacy Protection in Open Systems. In <http://www.csc.ncsu.edu/faculty/pubs/IAC.pdf>, April 2004.
- [6] N. Li and J.C. Mitchell. RT: A Role-based Trust-management Framework. In *DARPA Information Survivability Conference and Exposition (DISCEX)*, Washington, D.C., April 2003.
- [7] L. Sweeney.  $k$ -Anonymity: A Model For Protecting Privacy. *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems*, 10(5), 2002.