MinExp-Card: Limiting Data Collection Using a Smart Card

Nicolas Anciaux[1,2], Walid Bezza[3], Benjamin Nguyen[1,2], Michalis Vazirgiannis[4]

[1]INRIA Rocquencourt
Domaine de Voluceau
78153 Le Chesnay, FR
nicolas.anciaux@inria.fr

[2]Université de Versailles
45 avenue des Etats-Unis
78035 Versailles, FR
benjie.nguyen@inria.fr

[3]Conseil Général des Yvelines
2 Place André Mignot
78000 Versailles, FR
wallid.bezza@gmail.com

[4]Ecole Polytechnique
Route de Saclay
91128 Palaiseau, FR
mvazirg@yahoo.gr

ABSTRACT

Online services such as social care, tax services, bank loans and many others, request individuals to fill in application forms with hundreds of private data items, in order to calibrate their offer. In practice, far too much data is requested, leading to over data disclosure. As shown in our previous works, avoiding this problem would (1) improve the privacy of the applicants and (2) decrease costs for service providers. We demonstrate here a prototype designed and implemented in partnership with the General Council of Yvelines District in France. The prototype targets forms used to calibrate social care for dependant people. To maintain the privacy of the decision process used to calibrate the social care, we propose a smartcard implementation. We will show that a 50% reduction of the items exposed in application forms can be achieved, explore the quality and scalability of our smartcard implementation, and demonstrate its scope.

Categories and Subject Descriptors

K.4.1 [Computers and Society]: Public Policy Issues – *Privacy*

General Terms

Algorithms, Management, Performance.

Keywords

Privacy, Limited data collection, Limited data disclosure.

1. MOTIVATION

Social assistance, which includes providing financial, material or human resources to dependent people, is operated in France by General Councils of each district. Regarding dependent people, help is requested through a set of application forms with several hundreds, if not thousands of fields (e.g., the GEVA application form, used to request assistance for dependant people, is used in this demonstration, and has 440 fields). Around 60.000 such forms are processed each year by the Yvelines General Council, our partner in this project. This involves 160 council workers checking information and taking decisions based on a complex process that can be formalized by a set of logical rules (called Collection Rules). Application forms and corresponding decisions are stored during the period of assistance for positive decisions, and for a fixed duration of three years for rejected applications for auditing purposes or in the case of a dispute.

EDBT/ICDT'13, Mar 18–22, 2013, Genoa, Italy
ACM 978-1-4503-1597-5/13/03.

Although disclosing personal data is unavoidable when applying to services that provide a customized solution to the specific situation of each applicant, nearly half the EU citizens report being asked for more information than necessary, and 70% of them are concerned by this issue [11]. If it were possible to minimize the set of personal data items filled in application forms (while maintaining the same final decision), this would: (1) improve the privacy of the applicants, (2) decrease data processing cost (which includes manual checking) for the service provider, and (3) limit financial loss in the case of a data breach (now considered a serious threat by organizations).

The difficulty arises from the fact that this minimization cannot be determined statically during form construction, e.g., by specifying mandatory and optional fields. Indeed, in practice application forms are obtained by constructing the union of *all* data items possibly considered by the decision making process to build an appropriate proposal. Yet for a given user, only a small subset may effectively impact that proposal.

Example. A dependent person can benefit from financial support for a home aid in the following cases: having (i) a pension under €30.000 and an age above 80, (ii) a pension under €10.000 regardless of age, or (iii) more than two lost abilities (e.g., dressing and bathing independently). This collection rule (a Boolean DNF formula), leading to the *home_aid* benefit, is composed of three atomic rules, each composed of one or more predicates, as illustrated in Figure 1.

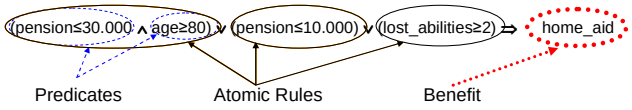

Figure 1. An Example Collection Rule.

For a user with values u_1 = [pension = €25.000, age = 81, lost_abilities= 1] the minimum data set would be [pension, age]. For a user with u_2 = [pension = €40.000, age = 60, lost_abilities = 2] it would be [lost_abilities]. Hence, the form cannot be specified a priori with the minimum set of attributes needed since it depends on looking at the values of all attributes available. Also note that in reality, decision rules are (much) more complicated.

Underlying problem and resolution techniques. We have proposed in [3] a new approach called *Minimum Exposure* to strictly limit the data exposed in application forms. This is achieved by producing a set of collection rules formalizing the decision process, which enable a program to automatically expunge useless data filled in application forms. We have shown in [4] that this problem is an extension of the Min-Weighted Satisfiability optimization problem, and therefore NP-Hard. Due

to the hardness of the problem, an exact resolution (even using state of the art solvers) requires important processing time. Therefore, we have proposed heuristics for approximation algorithms that comply with scalability and time constraints.

Demonstration scenario. We instantiate the minimum exposure framework on a real social care application, in partnership with the General Council of Yvelines District. This use-case involves a novelty: the decision process must be kept private because of (1) the discretionary nature of decisions taken by General Councils, and (2) in order to discourage fraud. Thus in this demonstration, we introduce a smart card (used as secure intermediary) on the user side to manage secure data collection based on secret rules. The smart card is a trusted third party: it must be trusted by the service provider not to disclose the collection rules, and by the applicant not to disclose their personal data and to correctly remove any data and intermediate computations once the processing is finished. A low cost smart card, certified by a third party and (possibly) destroyed after use by the applicant, is a suitable candidate. This demonstration aims to show that smart card technology can be efficiently used to process form minimization, which will significantly reduce three types of cost: privacy, processing and data breach costs.

In particular, we will (i) show that the gains obtained (using several metrics) for applicants privacy and processing cost for the General Council in the particular case of the social application request, are very important (between 40% and 80% in average depending on the considered metrics) in both the case of hidden and public rules; (ii) demonstrate that in real use cases the gain achieved by an architecture supporting hidden rules (smart card implementation) is within 20% of the gain obtained using a powerful server; and (iii) show that our resolution techniques are adapted to a large scope of problem topologies, and are highly scalable up to forms holding thousands of fields.

In the rest of the paper, we rapidly overview related works, present the MinExp-Card Platform, explain its features and describe the proposed demonstration scenarios.

2. RELATED WORKS

Existing techniques partly address the problem of Limited Data Collection (LDC) in privacy aware computing systems. Examples include the P3P Platform for Privacy Preferences [10], policy languages like EPAL [6] or XACML [13], and Hippocratic databases [1]. P3P highlights conflicting policies but offers no means to minimize the data exposed by a user, and policy languages were not introduced with LDC in mind. Hippocratic databases [1] address LDC by maintaining for each purpose, the set of required attributes. Useful attributes are statically derived from purposes which may hold for simple cases (e.g., the address is required to deliver a bought product) but not in general decision making processes.

Existing works closer to our study are conducted in the area of automated trust negotiation where access decisions are granted after evaluating credential requests. For each request the minimum set of credentials is disclosed. A few previous works like [5], [9], and [19] address this minimization step. In [5], the target is to identify the minimum set of credentials containing a given pre-identified set of items (properties), while we address the complementary problem of minimizing a set of items to reach a given set of benefits. In [9], credentials are modeled with propositions (predicates), leading to take decisions given the existence of *true* propositions (independently of data values). This model assumes that all potential propositions are available beforehand, and that their existence is not sensitive, which makes no sense in our context. In [19], secure multi-party computation techniques (SMC) are proposed to solve a hidden knapsack problem: data items have a utility and a privacy value, the issue is to attain a given utility threshold while minimizing the privacy score. This is inherently incompatible with multi-label decisions [16] (i.e. that consider several benefits such as provide human support, material assistance, home improvement, financial help, etc.) Indeed, a single global utility threshold is not expressive enough. More generally, only binary access decisions are considered in trust negotiation. In addition, our scalability requirement is up to two orders of magnitude greater than in trust negotiation: thousands of items may be considered in application forms while only 20-30 credentials participate in each round of a negotiation process [5].

3. THE MINEXP-CARD PLATFORM

Our platform is based on the use of a smart card as very low cost tamper resistant device. It can be used by any device (laptop, phone) fitted with a card reader (costing around 10€). Many laptops already have this feature (e.g., used for authentication). Many smart card products can be appropriate for this application: we only require enough stable storage to store empty application forms, the collection rules, and the web application code (e.g. under 32KB for the GEVA scenario). We use under 10KB RAM during execution. In this demonstration, we use the STMicroelectronics STM32-Discovery time-accurate hardware emulator for 32 bit RISC microcontrollers (ARM Cortex-M3) with 8KB RAM and 128KB stable storage. Corresponding smart cards are very cheap (only a few dollars).

In our architecture (pictured in Figure 2), we require a Collection Rule Extractor module and Card Upload module on the service provider side used to create of a set of smart cards, and three modules on the user side (card) : Form Filling; Form Scoring; and Minimum Exposure used to generate a form containing a minimum amount of information.

The Collection Rule Extractor produces Collection Rules modeling the decision making system, to subsequently determine the sets of required data items which impact the service proposal. In this application, rules were produced manually by experts at the General Council. In other contexts, rules could be generated automatically, e.g., if the decision making system involves data mining tools, such as neural networks or support vector machines, algorithms like [7] have been proposed to transform them into sets of collection rules.

The Card Upload module uploads collection rules and empty application forms to the smart card. Smart cards are then distributed by the General Council to its local agencies for use with chip-enabled terminals available on-site. Alternatively, a card can be used at home, in a laptop fitted with a card reader.

The Form Filling module is used to fill in the application form. Each cell of the empty application form must be filled in with a data item (an attribute/value pair) for which authenticity can be checked. Some items may be digitally signed by data producers (e.g., income signed by tax services), while others can be declared and signed directly by the applicant. In our prototype, application forms are filled either manually or automatically by

using a key value store of (attribute, value) pairs available for each applicant.

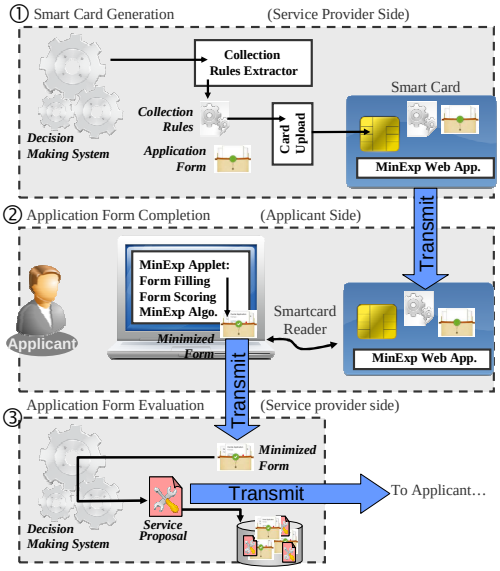

Figure 2. MinExp-Card Platform.

The Form Scoring module binds a Cost Function score to each data item entered in the application form. The difficulty resides in finding good metrics to capture different aspects: privacy for the applicant, and financial or breach costs for the service provider. Traditional information loss metrics like minimal distortion [15] or ILoss [17] can be considered good candidate functions. Indeed, since these metrics were created and used for privacy preservation, they accurately measure privacy. Furthermore they might also be accurate for measuring financial costs: it is obvious the (manual) checking cost for the service provider depends on the volume of the processed data. Moreover, the overhead induced by the cost of a data breach is also proportional to the amount of exposed data, as shown by a recent study [15]. Thus, any information loss metric will be a reasonable candidate cost function since service providers' costs and users' privacy are both tightly linked to such a metric. Our prototype can therefore accommodate any metric that associates an exposure value to each item independently (e.g., numeric values entered by the user herself, as well as the aforementioned metrics). In the GEVA application, we measure the processing cost in human minutes with the help of General Council experts. The form scoring module proposes default values for each data item which can be updated as desired. The demonstration includes comparing results obtained using various metrics.

The role of the MinExp module is to compute the benefits that the applicant can receive if she exposed all her data (input form), then reduce the amount of data to be exposed (output minimized form), while receiving the same benefits. The module is parameterized by the Service Provider Collection Rules on the one hand, and Cost Function on the other. Each collection rule predicate is a Boolean variable corresponding to one of the applicants' attributes, such as age, income, blood pressure, etc.

The minimized form is finally sent to the service provider, which conducts its decision process, produces a proposal to the applicant, and archives process information.

4. DEMONSTRATION SCENARIO

The demonstration runs on two types of input data: the real case GEVA application data, and a large scale example based on synthetic data that we will use to conduct a challenge with the audience. We divide the scenario into four parts: 1) collection or generation of the input data and collection rules, 2) execution of the Minimum Exposure Process and 3) graphical analysis of results. Figure 3 illustrates the demonstration steps.

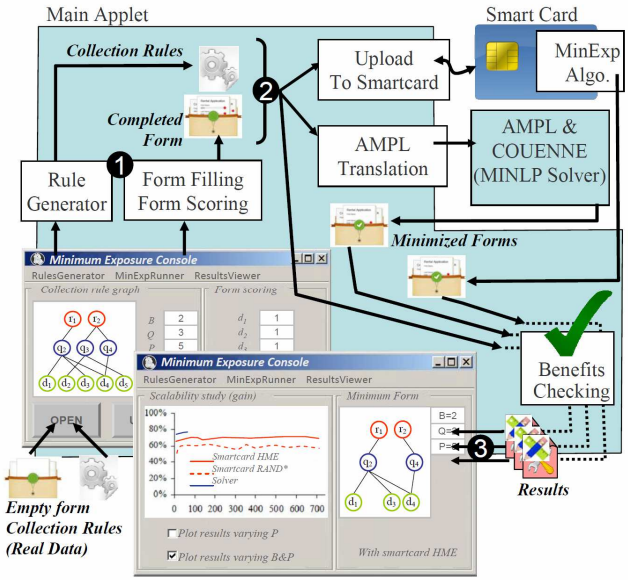

Figure 3. Demonstration Steps.

4.1 Input Data and Parameters

Collection Rules. We start off by using the View and Analysis module to show the collection rules (decision process), represented as a bipartite graph, illustrating the values of various parameters of a real case example. Indeed, we have shown in [2] that it is possible to characterize the topology of collection rules using the following parameters: number of predicates (P), number of benefits (B), number and distribution atomic rules a given predicate participates in (d_{PQ}), and number and distribution of atomic rules per benefit (d_{QB}). To investigate scalability, all parameters P and B will vary from the real case value up to a bound which can be fixed manually using the interface.

User Data. User data must be entered into forms issued by the General Council. Since these forms are large (more than 400 data items), they are automatically filled in this demonstration The data used are inspired by real applicants, but is not real for privacy concerns. A score is associated automatically to each data item. These scores have been determined by experts from the General Council based on the time estimated to check and process the data. Data values and scores can also updated on request.

In order to demonstrate the scalability of the approach to even larger forms, and diverse types of applications, we introduce the Synthetic Rule Generator module. This module produces a set of coherent collection rules, given the values of the parameters Q, B, d_{PQ}, and d_{QB}. In this case, we use as base values the ones obtained in the GEVA application, and we vary one or more parameters simultaneously.

To the same end, the Form Filling module is also able to generate synthetic user data. Given a set of collection rules, this

module generates a user whose data satisfies a certain proportion of collection rules predicates. The rest remains identical.

4.2 Minimum Exposure Computation

Collection rules and complete form are both inputs of the MinExp evaluation step. In the demonstration, this MinExp module is implemented in two different ways to compare results obtained using the smart card approximation algorithms with the exact solution. We use a) the smart card implementation of the algorithms HME, SA*, RAND* proposed in [4], and b) the state-of-the-art Binary Integer Program (BIP) solver *COUENNE* [8]. In case a) collection rules and complete form are uploaded to the smartcard executing the algorithms. In case b) we (straightforwardly) transform the collection rules to AMPL [12] format and input them to the BIP solver to produce an optimal result. The result of this step is a minimized form, whose privacy, processing and breach cost are evaluated. The demonstration will show that this form achieves well over 50% gain with regards to these costs, compared to the complete form.

The initial complete form and the minimized forms (produced by the smartcard and the BIP solver) are inputted into the *Benefits Checking* module which simulates the decision process to show that both forms yield the same benefits, attesting that no benefit was lost during minimization process.

4.3 Results Analysis

The Results Analyzer module is used to visualize the original complete form and their minimized counterparts, the collection rules (using the bipartite graph representation we introduced in [2]) and their statistics (number of edges of each type, average fan-out, etc.), the cost functions and when necessary the scores that have been considered at each step of the Min-Exp algorithms. This module is also used to analyze large quantities of forms to compare the quality of algorithms on synthetic data.

4.4 Challenge to the Audience

We have shown in [4] that the MinExp problem has no polynomial-time approximation scheme. This is in theory a bad result, since this means that any polynomial algorithm will have a very bad worse case. In this third part of the demonstration, we propose a challenge to the audience to see if such a worst case is easy to find: find a collection rules topology for which the ratio between the solution computed by the SPT algorithm and the exact solution computed by the solver is the greatest. All the topologies proposed will be generated by the Synthetic Rule Generator. To simplify the problem, we suppose that the user triggers all the predicates, and generate her data accordingly. As the demonstration will have shown, this ratio is under 120% for real case graphs.

5. CONCLUSION

This demonstration illustrates a real world application of the Minimum Exposure problem, to limit over data disclosure in the case of application forms used when requesting social care in France. This application proposes a slightly different context than the initial theoretical study, since decision rules must remain secret, leading to the use of smart cards. The demonstration shows that despite their low cost and low power, they can be used as an efficient implementation of the Minimum Exposure framework, and can be promoted in the field to increase privacy and reduce processing and breach costs. Moreover we show the scalability of our implementation beyond that of expert generated rules, and envision that the MinExp-Card platform could be used in many other applications, even in presence of a large number of private decision rules and data, such as loans or insurance.

6. ACKNOWLEDGMENTS

This work is supported by KISS ANR-11-INSE-0005, DIGITEO LeTeVoNe and INRIA CAPPRIS grants.

7. REFERENCES

[1] Agrawal, R., Kiernan, J., Srikant, R., and Xu, Y., 2002. Hippocratic databases. In *VLDB*.

[2] Anciaux, N., Boutara, D., Nguyen, B., and Vazirgiannis, M. 2012. Limiting data exposure in multi-label classification processes. In *International Workshop on Privacy-AwaRe Intelligent Systems*.

[3] Anciaux, N., Nguyen, B., and Vazirgiannis, M. 2012. The minimum exposure project: limiting data collection in online forms. *ERCIM News*, vol. 90.

[4] Anciaux, N., Nguyen, B., and Vazirgiannis, M. 2012. Limiting data collection in online forms. In *IEEE PST*.

[5] Ardagna, C.A., De Capitani di Vimercati, S., Foresti, S. , Paraboschi, S., Samarati, P. 2012. Minimising disclosure of client information in credential-based interactions. *Journal of Information Privacy, Security and Integrity*, vol. 1(2-3).

[6] Ashley, P., Hada, S., Karjoth, G., Powers, C., and Schunter, M. 2003. Enterprise privacy authorization language 1.2 (EPAL 1.2). *W3C Member Submission*.

[7] Baesens, B., Setiono, R., Mues, C., and Vanthienen, J. 2003. Using neural network rule extraction and decision tables for credit-risk evaluation. *Management Science*, vol. 49(3).

[8] Belotti, P., Lee, J., Liberti, L., Margot, F., Wachter, A. 2009. Branching and bounds tightening techniques for non-convex MINLP. *Optimization Methods and Software*, vol. 24(4-5).

[9] Chen, W., Clarke, L., Kurose, J., and Towsley, D. 2005. Optimizing cost-sensitive trust-negotiation protocols. In *IEEE INFOCOM*.

[10] Cranor, L., Langheinrich, M., Marchiori, M., Presler-Marshall, M., and Reagle, J. 2002. The Platform for Privacy Preferences 1.0 Specification. *W3C Recommendation*.

[11] EU Commission. 2011. Attitudes on data protection and electronic identity in the European Union. *Eurobarometer S. Surveys*, vol. 359.

[12] Fourer, R., Gay, D.M., and Kernighan, B.W. 1990. A modeling language for mathematical programming. *Management Science*, 36.

[13] Moses, T. 2005. Extensible access control markup language (XACML) version 2.0. *Oasis Standard*.

[14] Ponemon Institute, LLC. 2011. 2010 Annual Study: U.S. Cost of a Data Breach.

[15] Samarati, P. 2001. Protecting respondents' identities in microdata release. *IEEE TKDE*, vol. 13(6).

[16] Tsoumakas, G., Katakis, I. 2007. Multi-label classification: an overview. *Journal of Data Warehousing & Mining*, vol. 3(3).

[17] Xiao, X., and Tao, Y. 2006. Personalized privacy preservation. In *ACM SIGMOD*.

[18] Yao, D., Frikken, K.B., Atallah, M.J., and Tamassia, R. 2008. Private information: to reveal or not to reveal. *ACM TISSEC*, 12(1)