2013 IEEE Security and Privacy Workshops

Preventive Inference Control in Data-centric Business Models

Rafael Accorsi and Günter Müller
University of Freiburg, Germany
{accorsi,mueller}@iig.uni-freiburg.de

***Abstract*—Inference control is a modern topic in data usage management, especially in the context of data-centric business models. However, it is generally not well understood how protection mechanisms could be designed to protect the users. The contributions of this paper are threefold: firstly, it describes the inference problem and relate it to protection mechanisms; secondly, it reports on a simple mechanism to provide a-priori inference protection; thirdly, it discusses on the drawbacks of such a mechanism, as well as on the acceptance it had on a preliminary, controlled field study. In particular, the study shows that, contrary to our expectations, participants prefer an a-posteriori approach based upon audits to detect whether inferences happened.**

I. Introduction

Privacy protection is essential in social networks [1], E-commerce [2] and novel service-oriented and Cloud computing architectures [3], which collect information related to a user to offer individualized services [4]. This paper refers to these services as "data-centric business models". Users decide which data they opt to disclose and to kept private. Privacy settings and policy languages, such as EPAL, OSL, P3P or XACML, formalize these preferences for processing and enforcement.

However, even if policies fully capture the preferences of users and are effectively enforced by monitors, illegitimate access to classified data can still occur by means of *inferences*. Inference stands for the derivation of attributes based upon observed events. Figure 1 depicts this problem. The set of legitimately released data D (so-called *core*) allows for the derivation of additional data D' (i.e. *inferential closure*), thereby violating the privacy policy of a user. As an example of such an *inference*, if a user reveals his full address, further correlations may allow one to obtain his telephone number. Such relationships are captured by *inference rules*. Inferences pose a serious threat to users' privacy [5], [6], yet the state of the art does not classify these threats as well as does not offer no tool support for inference control.

This paper first analyses the inference problem in its various dimensions and correlates them with protection mechanisms which could be used to tackle the problem. Spefically, the paper presents an approach for a-priori inference control during the privacy policy specification. The approach is based on resolution, a reasoning technique for automated theorem proving for sentences in propositional and first-order logic [7]. Assuming a universe of data items $\mathcal{D}$, inferences are modeled as rules of the kind $A \rightarrow d_i$, where $A \subseteq \mathcal{D}$ and $d_i \in \mathcal{D}$ is a particular data item inferable from A. Resolution employs refutation to show, for each $d_i \in \mathcal{D} \setminus D$, whether $D \models d_i$. Every d_i for which this holds poses a specific *inference threat* allowed by the policy. The paper further reports on a controlled, preliminary field study in which users could apply the technique to compute the inferences based on their policies based on an interview. The goal being to elicit their expectations on a tool for inference control and the tool at hand. While the vast majority of users felt somewhat comfortable with the tool, respondents stated that a tool for a-posteriori inference control would have been more appropriate and "trustworthy". In sum, this paper thus makes the following contributions:

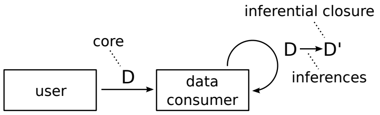


Fig. 1. Schematic view of the inference problem.

- It analyzes the inference problem and establishes it relationship to data usage management.
- It introduces a mechanism for inference control based on resolution to assist users in the specification and refinement phases of complex privacy policies.
- It reports on the feedback obtained from a field study with the mechanism.

The overarching goal of our approach is, on the one hand, to make users aware of inference threats and the risk they pose. On the other hand, we aim to design mechanisms for users to control inferences and, in the long run, enhance the trust on service providers. With regard to the particular mechanism we provide in this paper, users can use it to refine their policies to reduce inferences. To further allow for a qualitative distinction of threats according to their threat level, the policy language allows users to label data item with a sensitivity, which is employed by the inference engine to classify the threats. This allows the representation of inference threats in a *unity circle*, a visualization cue for inferences in databases [9]. The center of the circle stands for the core D and the inferable data items are points in it. The distance represents the threat level of an inference: the closer it is lies to the center, the higher the threat.

Related Work. The problem of computing and controlling inferences has been extensively investigated in the setting of statistical databases, e.g. in [10], [11]. Later, inferences were considered for multidimensional general-purpose databases [12], [13] where different users with different clearances share the same data and the situation where the combination of lower level data enables the inference of higher level data has to be prevented. Up to now, inference control in data-centric business models has been only sparsely considered and there is to be

DOI 10.1109/SPW.2013.25

best of our knowledge no tool support to assist users. In the military setting, [14] proposes the use of resolution to check for inferability. However, the concept of user policy is not considered. In [15], the author proposes an approach based on logic programming to check whether a program allows inferences that violates privacy policies. Finally, An et al. [5] model the knowledge of an "inference-savvy" adversary using Bayesian networks. Here, only security levels are considered and not the particular policies of users.

Structure. §II introduces the inference problem and the corresponding protection mechanisms. §III presents an a-priori approach to controlling inferences and §IV reports on a preliminary field study carried out with this tool.

II. Inference Threat and Protection Mechanisms

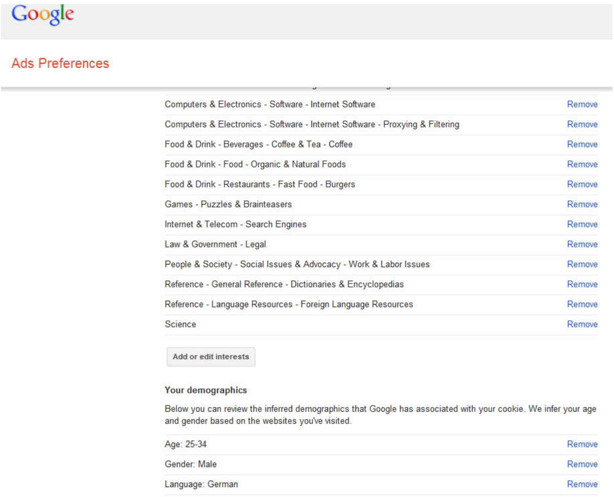
This section reports on the inference threat and on possible protection mechanisms for inference control.

A. Types of Data in Data-centric Business Models

Data-centric business models call on user profiles to (1) improve user experience through personalization and individualized content and (2) classify users into advertising target groups. These classifications are created by use of different types of user data. According to Schneier [16], the following types of data can be distinguished:

- *Service data*: Data a user discloses to get access to a service (e.g. email address or credit-card information).
- *Disclosed data*: Data a user actively discloses in the service (e.g. a picture upload or post).
- *Entrusted data*: Data a user actively discloses in the service. As opposed to disclosed data, the user cannot delete or modify entrusted data once disclosed (e.g. data in a message to another user of the service).
- *Incidental data*: Data *other* users disclose about a user (e.g. a user's contact information synchronized by other users with their online address books).
- *Behavioral data*: Data about a user's behavior (e.g. a user's clickstream or "likes" on the service's website).
- *Technical data*: Data disclosed passively by the user via the devices or software used to access a service (e.g. a user's IP address or browser "agent").
- *Inferred data*: Data about a user that is derived from all other data types (e.g. a user's interests inferred from the user's behavior).

Service data are data a user discloses in order to register for and get access to a service, such as e.g. an email address or credit-card information. *Disclosed data* and *entrusted data* are actively and knowingly disclosed by the user. *Technical data* is disclosed passively by the user and not necessarily with the user's knowledge. The latter three types of data can be protected by state of the art privacy protection technologies. However, users only have very limited means to control disclosure of *incidental data* and *behavioral data*.



Computers & Electronics - Software - Internet Software	Remove
Computers & Electronics - Software - Internet Software - Proxying & Filtering	Remove
Food & Drink - Beverages - Coffee & Tea - Coffee	Remove
Food & Drink - Food - Organic & Natural Foods	Remove
Food & Drink - Restaurants - Fast Food - Burgers	Remove
Games - Puzzles & Brainteasers	Remove
Internet & Telecom - Search Engines	Remove
Law & Government - Legal	Remove
People & Society - Social Issues & Advocacy - Work & Labor Issues	Remove
Reference - General Reference - Dictionaries & Encyclopedias	Remove
Reference - Language Resources - Foreign Language Resources	Remove
Science	Remove

Your demographics

Below you can review the inferred demographics that Google has associated with your cookie. We infer your age and gender based on the websites you've visited.

Age: 25-34	Remove
Gender: Male	Remove
Language: German	Remove

Fig. 2. Sample categories in Google's Ads Preference Manager

B. Inferred Data and Privacy Threat

Inferred data is "data about you that is derived from all the other data" [16]. Figure 2 depicts the categories of interest and demographic information inferred by Google on the basis of a user's search and browsing behavior (http://www.google.com/settings/ads/onweb/. Inferred data such as depicted in Fig. 2 are created by use of inference rules. There are three general kinds reasoning underlying such inference rules: (a) *Deduction*, i.e. the process of reasoning from one or more general statements or events to reach a certain conclusion (inferred data); (b) *Abduction* is the opposite of deduction, i.e. generalizing statements based on observable events. For example, from a search query "Franz Beckenbauer" one abduct the fact that the user likes soccer; and (c) *Induction*, i.e. constructs or evaluates general propositions (inferred data) derived from specific examples. This kind of reasoning is particularly useful to create behavioral patterns of user interaction.

Inference rules based upon these reasoning styles require domain knowledge, statistical learning and automated techniques to elicit patterns from collected data. Note that inferred data is obtained not by analyzing a specific user's data alone, but by analysis of all data available to a data consumer, including the data of other users. Inferred information can be used to create more precise profiles, respectively categorizations of users into advertising target groups. It is exactly the abundance of data on the side of the data consumer and its capability of cross-analyze the collected data that makes inference a severe threat to the privacy of users in data-centric business models.

C. Possible Inference Control Mechanisms

Controlling inference and designing mechanisms for this purpose bears a number of challenges. The problems start, as mentioned above, with the fact that inference can be drawn on the grounds of other users' data. That is, even if a user do not disclose a particular piece of data (attribute), it could be inferred from other users considered "similar". Of course, such an inference might be wrong or just an approximation. However, our experience drawn from the field study shows

TABLE I. MECHANISMS FOR INFERENCE CONTROL.

Timepoint	a priori	runtime	a posteriori
Design principle	Information minimization	Monitor inference steps and data merging	Audit logs and dashboards
Mechanisms	Policy analysis to detect possible inferences at allow policy redesign	Trusted monitors capable of detecting complex inference steps and profile updates	Secure logging and data provenance to assert quality of data and analysis methods to detect inference

that, e.g., Google Ads Preferences (Fig. 2) rightly infers the age interval and gender of users.

Table I classifies the possible inference control mechanisms according to the timepoint they act and the design principle they follow. Orthogonal to these mechanisms one could also consider the reasoning approaches to concretely detect inferences (deduction, abduction and induction). Clearly, a priori and runtime methods could prevent inferences from being drawn, thereby being traditional privacy enhancing technologies; a posteriori methods would only be able to detect the fact that inferences have been drawn based upon user's data, which is a transparency enhancing technology.

III. A-PRIORI INFERENCE CONTROL

This section presents an approach based upon the "deductive" reasoning style for a priori inference control.

A. Overview and Building Blocks

The approach allows users to refine the policies with regard to the inference threats posed by a data consumer. The main assumption underlying the approach is that the inference rules are known by the inference engine. (These rules can be offered by a third-party service or made public by the data consumer – as a thrust to improve transparency.) This does not prevent the user from adding inference rules on her/his own discretion.

The diagram in Fig. 3 shows the refinement process. (This process is not bounded to the resolution-based engine proposed in this paper.) Assuming a shared vocabulary for data items (i.e. *domain*, see below), the user composes the privacy policy for a particular data consumer and passes this policy to the inference engine. With the policy and the privacy rules at hand, the engine computes the inferential closure of the input policy and evaluates the threats it poses according to the security level associated to the data items, visualizing these results as an unit circle. Given the resultant set of inferences, the user may further refine the policy and eventually agree on a policy.

The approach comprises the following building blocks. A *domain* consists of a countable set of data items $\mathcal{D} = \{d_1, d_2, ...\}$. The *sensitivity* associated to data objects is denoted as $d_i.s \in [0;1]$. The domain is used to define the set of data items used as a vocabulary for policy definitions within a system. A *privacy policy* consists of a set of rules, where each rule grants access to a set of data items for a specific security level. Security levels are assumed to form a lattice, where $L_i \prec L_j$ denotes the fact that L_j is more sensitive than L_i. Formally, each *policy rule* $r_i = (O, L)$ of a privacy policy $P_{\mathcal{D}} = \{r_1, r_2, \ldots, r_n\}$ based upon a domain $\mathcal{D}$ consists of a set of a data items $r_i.O \subseteq \mathcal{D}$ and a classification level $r_i.L$. $|P_{\mathcal{D}}|$ denotes the number of rules (cardinality) of a policy. A default-deny configuration is assumed: attributes not explicitly disclosed are considered to remain private.

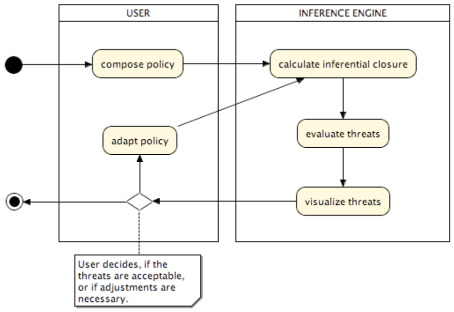

USER
INFERENCE ENGINE
compose policy
calculate inferential closure
adapt policy
evaluate threats
visualize threats
User decides, if the threats are acceptable, or if adjustments are necessary.

Fig. 3. Policy refinement process.

Policies are used to extract cores for specific security levels to then apply inference control approaches. The inference engine subsumes at least a formalization of inference rules, an algorithm to compute the inferential closure. An *inference rule* describes a semantic relationship between data items and is denoted by $A \rightarrow d_i$, where $A \subseteq \mathcal{D}$ is a possibly unitary set of data items and $d_i \in \mathcal{D}$ is a data item. The data required for inference (at the left hand side of the arrow) is called *support* and the inferable data item *derivate*. Inference rules with a unitary support are called *binary inference rules*, otherwise *combined inference rules*. The following provides an example of privacy policy and some applicable inference rules.

Example Let the security levels of a social network be organized as a lattice $L_{SN} = \mathit{All} \prec \mathit{FriedsofFriends} \prec \mathit{Friends}$. The domain $\mathcal{D}$ contains five data items as follows $\mathcal{D} = \{\mathit{name}, \mathit{address}, \mathit{birthday}, \mathit{telnr}, \mathit{email}\}$. The policy $P_{\mathcal{D}} = \{r_1, r_2\}$ consists of two rules: (1) $r_1 = (\{\mathit{name}, \mathit{birthday}\}, \mathit{All})$, and (2) $r_2 = (\{\mathit{email}, \mathit{address}\}, \mathit{Friends})$. The rule r_1 denotes that the data items *name* and *birthday* can be seen by all other users, whereas the data items *email* and *address* only by friends. The default-deny setting prohibits access to *telnr*. $\dashv$

Given a set of initial data items (*core*), the *inferential closure* includes all the derivate that can be inferred by (possibly iterated) applications of inference rules. Here, one can distinguish between *simple inferability*, which means that all the data items in the support must be in the core, and *extended inferability*, in which case iterated applications of inference rules are possible. (In particular, extended inferability allows derived non-core data items to serve as support for a rule.) Correspondingly, the *simple inferential closure* includes all derivates computed by simple inferability, whereas the *extended inferential closure* encompasses all the data items computed by the extended inferability.

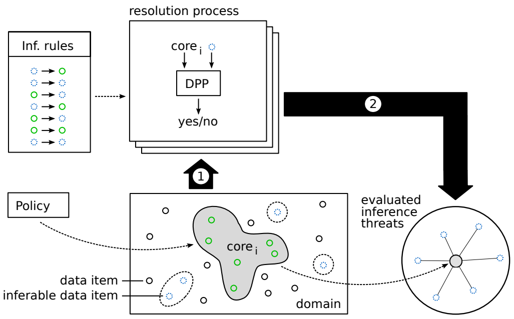


Fig. 4. Inference engine of the resolution-based approach.

B. *Resolution-based Engine for Inference Control*

This section presents a resolution-based inference engine to compute the extended inferential closure using combined inference rules. Fig. 4 depicts the steps necessary to do so. First, the core is extracted from the corresponding privacy policy and passed to the resolution process, which tests for each non-core element whether it is derivable from the core. The inference threats are evaluated and shown to the user. The following defines these steps in a formal manner.

Turning to the engine, assuming a finite, nonempty domain $\mathcal{D} = \{d_1, \ldots, d_n\}$, the structure of an inference rule is:

Def. 1: $A \rightarrow y$ is an inference rule where the support $A \subseteq \mathcal{D}$ is a set of data items and $y \in \mathcal{D}$ is the derivate. $\dashv$

Propositional logic is used to express the data items and inference rules. The set of propositional variables is given by the data items in $\mathcal{D}$. With propositional logic, inference rules can be rewritten as disjunctions following the equivalence $(a_1 \wedge \cdots \wedge a_n) \rightarrow y \equiv (\neg a_1 \vee \cdots \vee \neg a_n \vee y)$. The core contains all the elements released by a policy having at least the security label label L. Formally:

Def. 2: A core of security level L relative to a policy P is defined as $Core_L(P) := \bigcup_{i \in |P|} \{o \in r_i.O \mid r_i.L \prec L\}$. $\dashv$

To extract a core from a policy, the policy P is examined rule by rule. If $r_i.L \prec L$, then all data items in $r_i.O$ are recognized as core elements. Once $Core_L$ is determined, the question is which elements are inferable by known inference rules that do not belong to $Core_L$. Using logic, inferability is modeled with deduction and allows the computation of the core extent, i.e. the set of all derivate of a core given a set of inference rules. The following defines these sets.

Def. 3: The extent of a core $Core_L(P_\mathcal{D})$ for a domain $\mathcal{D}$ and policy $P_\mathcal{D}$ is the set of all inferable data items from core elements $d \in Core_L(P_\mathcal{D})$ and (iterated) inference rules r. The set of elements not in the core is defined as $E := \{d \in \mathcal{D} \setminus Core_L(P_\mathcal{D})\}$. Let $\mathcal{R}$ be the set of inference rules,

$$\begin{aligned} &Ext_{Core_k(P_D)} := \\ &\quad \{d \in E \mid \; [\exists A \subseteq Core_L(P) : (A \rightarrow d) \in \mathcal{R}] \vee \\ &\qquad [\exists d_1, \ldots, d_j \in E \exists\; A_0 \subseteq Core_L(P) \\ &\qquad \exists\; A_1, \ldots, A_j (A_i \subseteq Core_L(P) \cup \{d_1, \ldots, d_i\}) : \\ &\qquad \{(A_0 \rightarrow d_1), (A_1 \rightarrow d_2), \ldots, (A_j \rightarrow d)\} \subseteq \mathcal{R}]\}. \end{aligned}$$

The inferential closure C of $Core_L(P_\mathcal{D})$ relative to a domain $\mathcal{D}$ and a policy $P_\mathcal{D}$ is the set of all inferable data items, i.e, $C_{Core_L(P_\mathcal{D})} := Core_L(P_\mathcal{D}) \cup Ext_{Core_L(P_\mathcal{D})}$. $\dashv$

Algorithm 1 IsDeducible($\mathit{\Gamma}$, d)

```
Require: $\mathit{\Gamma}$ is a clauseset
$\mathit{\Gamma}' := \mathit{\Gamma} \wedge \neg d$
$\{Y_1, ..., Y_n\} \leftarrow var(\mathit{\Gamma}')$
$i \leftarrow 1$
while $i \leq n$ do
    (A) $\mathit{\Gamma}' := \mathit{\Gamma}' \setminus \{K \in \mathit{\Gamma}' | \; \exists\lambda : \lambda, \bar{\lambda} \in K\}$
    (B) $\mathit{\Gamma}' := \mathit{\Gamma}' \cup \{K | \; \exists K_1, K_2 : K_1 \sqcup \{Y_i\}, K_2 \sqcup \{\neg Y_i\} \in \mathit{\Gamma}' \wedge K = K_1 \cup K_2\}$
    (C) $\mathit{\Gamma}' := \mathit{\Gamma}' \setminus \{K \in \mathit{\Gamma}' | \; Y_i \in var(K)\}$
    $i \rightarrow i + 1$
end while
if $\mathit{\Gamma}' = \emptyset$ then
    return false                                    ▷ $\mathit{\Gamma}'$ is satisfiable
else
    return true                                     ▷ $\mathit{\Gamma}'$ is not satisfiable
end if
```

Given these definitions, the following defines algorithms to determine the inferential closure and evaluate the inference threats posed by a policy and a set of inference rules.

C. *Computing the Inferential Closure*

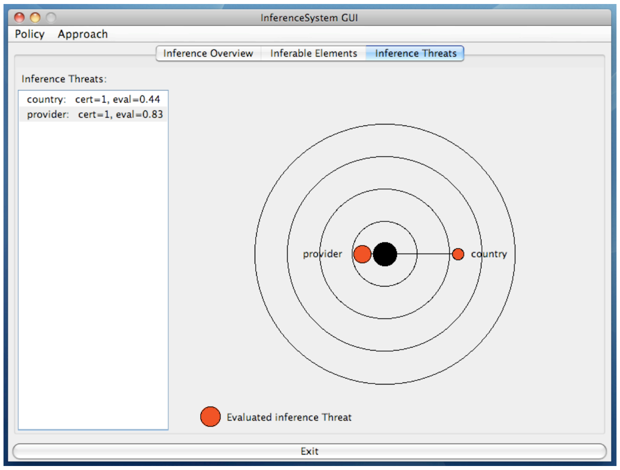
Given a core, the theory Γ (basis for resolution) consists of the core elements and the inference rules. Prior to the actual resolution, the theory has to be rewritten as clauses. This preprocessing step encompasses two rules: (1) for each core element $d \in Core_L(P_\mathcal{D})$, $\{d\} \in \Gamma$; and (2) for each inference rule $r = (a_1 \wedge \cdots \wedge a_n) \rightarrow y$, $\{\neg a_1 \vee \cdots \vee \neg a_n \vee y\} \in \Gamma$.

The inferential closure of a given core is determined by finding resolution proofs for the deductibility of each data item that does not belong to the core. To this end, we employ the David-Putnam Procedure (DPP) on the resultant theory Γ. The algorithm is given in Alg. 1.

The function $var(K)$ respectively $var((K))$ returns the set of variables in the literals of a clause K or a clauseset (K). Every run of the Alg. 1 considers the inferability of one non-core variable. In Step (A) the algorithm removes clauses that represent tautologies from the clauseset Γ' (formulas that evaluate to true for every possible variable assignment). In Step (B), it determines new clauses along the actual considered variable by applying the resolution rule and in Step (C), it removes all clauses containing this variable. At the end, the structure of the remaining clauseset Γ' implies the deductibility of d. If $\emptyset$ is derived, then Γ' is satisfiable, i.e. the data item cannot be inferred; otherwise, if it is $\{\emptyset\}$, then Γ' is not satisfiable and thus $\Gamma \models d$ is a tautology, indicating that the data item can be inferred from Γ.

D. *Visualizing Inference Threats*

The resolution-based approach considers absolute inference, i.e. inferable data items are either completely inferred or not at all. To distinguish the severity of threats, we employ the sensitivity associated with each of the data items in the domain, as depicted in Fig. 5. The red/gray dot stand for the data items in the inferential closure of the core (center of the circle). The distance between the these dots and the center indicates the threat level of an inference (calculated for an inferable data item d as $1 - d.s$): the closer a dot is placed to the core, the higher its threat. To enhance the usability, user-defined levels can be added (concentric circles within the unit circle). In a refinement process as is outlined in Fig. 3, the



country: cert=1, eval=0.44
provider: cert=1, eval=0.83

Fig. 5. Unit circle depicting inference threats.

user can now make changes on his policy and check whether these modifications lead to a reduction of the threats.

The prototypical implementation of the approach is available. Users can either compose their own policy or upload existing policy specifications in XACML (v2.0) and P3P. The inference rules can be uploaded into the tool or retrieved from a server. The process of deriving and showing the inferences is "push-button" and the refinement steps are iterative, i.e. users return to the editing window and can then compare the whether the new policy is more restrictive or permissive in terms of inference. The overall tool design and presentation follows the user-centered security engineering approach (UCSEC) [17].

IV. User Experience and Acceptance

In order to understand how users would judge such a tool, we conducted a preliminary field study based upon a simple questionnaire within the lecture "Security in Business Processes" offered in Freiburg in the Winter Term 2012/13. Towards its end, the course approaches the privacy issues inherent to the use of business processes in large enterprises. In this context, the inference threat posed to privacy, mechanisms to support inference and the countermeasures to protect users were described and discussed. The pool of students in amounts to 27 graduate students enrolled in the Master Program "Applied Computer Science" offered by the University of Freiburg, Germany. The exercises on inference control included of the following questions:

1) Were you aware of inferences before the lecture?
2) Were you aware of the Google "Ads Preference"?
3) Do inferences threat your privacy? Justify.
4) Would you use the tool for a-priori inference control?
5) Having the choice, which privacy protection mechanism would you choose to control inferences? Why?

Of 27 respondents, only 2 were aware of inferences in data-centric services (7 percent) and none of the respondents were aware or had visited the Google "Ads Preference" page before. Over 80 percent (22 respondents) consider inference as an imminent threat to their privacy. The key argument is that inference leads to detailed profiling. The remaining five respondents consider inference a legitimate means to improve business and service quality and, hence, in-line with the privacy legislation. Interestingly, 90 percent of the respondents would make use of a privacy-enhancing tool for inference control. That is, some of the respondents would control inference even if they do not consider it harmful to their privacy. Around 25 percent of the respondents (7 individuals) would use a preventive method for inference control; 20 respondents would prefer an approach based upon dashboards to audit the system. The main argument to favor dashboards was the fact that they allows a more complete and "trustworthy" view of the inferred data.

The results for Questions (1)–(4) are not surprising. Users are generally unaware of inferences, a phenomena we could also observe in other similar interviews. We consider surprising that respondents would have felt more confident with a dashboard to detect inferences rather than a tool to prevent inferences. Extrapolating these results, this could indicate that users are more permissive in publishing their data, while they at the same time desire a mechanism to know what happens to data. That is, a move from strict "access control" (incarnated as minimal data release) to "usage control" after the fact.

Due to the relatively small number of respondents and their homogeneity, the study cannot be considered representative. It is, however, a snap-shot which in many ways corroborates the results of recent studies, e.g. [18], [19]. Further, the overall study design, which is rudimentary. This includes the type of questions we posed (predominantly open, largely disconnected questions with free text justifications). In addition, respondents could have been somewhat biased by the contents of the lecture, even though the lecture was neutral, equally considering the pros and cons of inferences. Finally, respondents could only experience one kind of inference control. For fair judgment, respondents must also test with other tools.

V. Summary and Future Work

Inference is a very useful concept for several applications and services, and there is a need to understand its various facets, control possibilities and implications to users' privacy. This paper tackled the problem of inference control in data-centric business models. It merely touches the tip of an iceberg, though, and several relevant issues still need to be approached. Besides a more thorough study of users' perception of inference threats and control, future work (1) generalizes the definition of inference to also refer to statistical data associated to a group of users; (2) provides a detailed classification of inference control mechanisms (and their underlying assumptions and techniques); (3) investigates the relationship between inferences and "risk" (the likelihood that a piece of data or attribute is going to be disclosed) [21]; and (4) devises dashboard techniques to discover inferences [20].

References

[1] T. H. Ngoc, I. Echizen, K. Komei, and H. Yoshiura, "New approach to quantification of privacy on social network sites," in *IEEE Conf. on Advanced Information Networking and Applications*. IEEE, 2010, pp. 556–564.

[2] G. Udo, "Privacy and security in concerns as major barriers for e-commerce: A survey study," *Inf. Mngt. Comp. Sec.*, vol. 9, no. 4, pp. 165–174, October 2001.

[3] R. Chow, P. Golle, M. Jakobsson, E. Shi, J. Staddon, R. Masuoka, and J. Molina, “Controlling data in the cloud: Outsourcing computation without outsourcing control,” in *ACM Workshop on Cloud Computing Security*. ACM, 2009, pp. 85–90.

[4] S. Sackmann, J. Strüker, and R. Accorsi, “Personalization in privacy-aware highly dynamic systems,” *Comm. ACM*, vol. 49, no. 9, pp. 32–38, 2006.

[5] X. An, D. Jutla, and N. Cercone, “Dynamic inference control in privacy preference enforcement,” in *Conf. Privacy, Security and Trust*. ACM, 2006, pp. 1–10.

[6] A. Antón, E. Bertino, N. Li, and T. Yu, “A roadmap for comprehensive online privacy policy management,” *Comm. ACM*, vol. 50, no. 7, pp. 109–116, 2007.

[7] J. A. Robinson, “A machine-oriented logic based on the resolution principle,” *J. ACM*, vol. 12, no. 1, pp. 23–41, 1965.

[8] M. Davis and H. Putnam, “A computing procedure for quantification theory,” *J. ACM*, vol. 7, no. 3, pp. 201–215, 1960.

[9] D. Denning and M. Morgenstern, “Military database technology study: AI techniques for security and reliability,” SRI, Tech. Rep., 1986.

[10] I. P. Fellegi, “On the question of statistical confidentiality,” *J. Amer. Stats. Assoc.*, vol. 67, pp. 7–10, 1972.

[11] M. Hansen, “Insuring confidentiality of individual records in data storage and retrieval for statistical purposes,” in *Joint Computer Conference*. ACM, 1971, pp. 579–585.

[12] D. E. Denning, “A preliminary note on the inference problem in multilevel database management systems,” in *National Computer Security Center Invitational Workshop on Database Security*, 1986.

[13] M. Morgenstern, “Controlling logical inference in multilevel database systems,” in *IEEE Symp. Security & Privacy*. IEEE, 1988, pp. 245–255.

[14] N. Rowe, “Inference-security analysis using resolution theorem-proving,” in *Conf. Data Engineering*. IEEE, 1989, pp. 410–416.

[15] R. Chandramouli, “Privacy protection of enterprise information through inference analysis,” in *IEEE POLICY*. IEEE, 2005, pp. 47–56.

[16] B. Schneier, “A taxonomy of social networking data,” *IEEE S/P*, vol. 8, no. 4, p. 88, 2010.

[17] U. Jendricke and D. G. tom Markotten, “Benutzbare Sicherheit durch Identitätsmanagement,” *Datenschutz und Datensicherheit*, vol. 27, no. 5, 2003.

[18] S. Spiekermann, J. Korunovska, and C. Bauer, “Psychology of ownership and asset defense: Why people value their personal information beyond privacy,” in *Conf. Information Systems*. AIS, 2012.

[19] N. Christin, “Traveling the silk road: A measurement analysis of a large anonymous online marketplace,” *CoRR*, vol. abs/1207.7139, 2012.

[20] R. Accorsi and S. Stocker, “On the exploitation of process mining for security audits: The compliance checking case,” in *ACM Symp. Applied Computing*. ACM, 2012, pp. 1709–1716.

[21] R. Accorsi, Y. Sato, and S. Kai, “Compliance monitor for early warning risk determination,” *Wirtschaftsinformatik* vol. 50, no. 5, pp. 375–382, 2008.