Issues and Challenges for the Detection of Inference Attacks

Paul Lachat, Nadia Bennani, Veronika Rehn-Sonigo, Lionel Brunie, Harald Kosch

Issues and Challenges for the Detection of Inference Attacks: Multi-Database Context

Paul Lachat[1,2], Nadia Bennani[1], Veronika Rehn-Sonigo[3], Lionel Brunie[1], Harald Kosch[2]

1 LIRIS, INSA Lyon, Villeurbanne France.
2 DIMIS, University of Passau, Passau, Germany.
3 FEMTO-ST Institue, University of Bourgogne Franche-Comte, Besançon, France.

Abstract

Nowadays applications produce and manage data of individual among which some may be sensitive and must be protected. Moreover, with the advent of smart applications, sensor data are produced by IoT devices in a huge quantity and sent to servers in the vicinity to be stored and processed. Meanwhile, newly discovered inference channels involving sensor data gives insights on personal data and raises new threats on individuals privacy. They escape the vigilance of traditional inference detection systems devoted to protecting personal data stored locally in a database. In this paper, we motivate the need of a distributed inference detection system acting in a general multi-database context and we highlight the issues that such a system would face.

1 Introduction

The current ubiquity of personal data implies that collected data is exchanged with data collectors and shared with authorized entities. For the sake of privacy, databases are protected by access control (AC) mechanisms against direct unauthorized access to sensitive personal data. But, due to the inference problem [FJ02], dishonest users can indirectly access sensitive personal data by exploiting for instance the *dependency strategy* [WB10]. Therefore, additional databases can be protected by inference detection systems (InfDSs) against inference attacks [CC06]; [TFE10]; [GMB17]; [CM03]; [NA19]. However, personal data is rarely stored in a single database but rather spread over multiple databases managed by several distinct entities. In such a multi-database context (MD), using the *distributed dependency strategy* [WB10], a dishonest entity can leverage her authorized accesses to non-sensitive data stored in distinct databases, to infer sensitive data of individuals. By doing so, the attacker bypasses existing InfDSs proposed in the literature, which each protects a database. A first study of this problem for personal databases has been presented

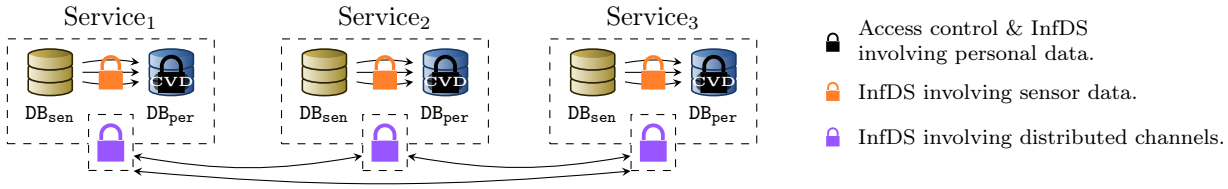

Figure 1: Three services have their own local sensitive personal data protected by an InfDS involving personal data and another involving sensor data. The services collaborate via a last InfDS to be protected against distributed inference attacks.

in a previous work [LRB20]. With the advent of smart devices, sensor data characterizing physical measurements are issued and collected as data streams. Such data is either directly (e.g., by means of wearable sensors [Ban+15]) or indirectly (e.g., measuring the power consumption of a house [EE14]) related to an individual. Several works [Krö19] have demonstrated ways to infer personal data using sensor data. Therefore, it is possible to infer sensitive personal data by involving non-sensitive personal data and sensor data in a MD context.

To illustrate these privacy threats, let us consider the following motivating example where a recommendation service aims to enable its customers to stay healthy. To do so, the service collects and stores in a database $\texttt{DB}_{\texttt{per}}$ the following information: *Age*, *Sex*, *PAL* (Physical Activity Level) and *CVD* (Cardiovascular Disease) status. In parallel, it collects and stores in a database $\texttt{DB}_{\texttt{sen}}$ the sensor data generated by wearable sensors of customers which it uses to compute their activities [Ban+15] in order to propose them the most suitable exercises. To protect sensitive personal data in $\texttt{DB}_{\texttt{per}}$, i.e., the *CVD* in our case, the service deploys both an AC mechanism and an InfDS on the $\texttt{DB}_{\texttt{per}}$. We assume that authorized entities (e.g., an insurance employee) can access data from both $\texttt{DB}_{\texttt{per}}$ (except the *CVD*) and $\texttt{DB}_{\texttt{sen}}$, in order to be able to assess the health risk of each customer and adapt their insurance policies. In such a setting, an authorized but dishonest entity could infer the *CVD* status of a customer c with a high enough percentage of confidence, i.e., $\mathrm{P_c}(\mathrm{CVD}) \geq 50\%$, by leveraging a first background knowledge describing the probability of having a *CVD* based on *Age*, *Sex* and *PAL* [Kub+17]. Assuming that the InfDS prevents the exploitation of this inference channel, the attacker queries only the *Age* and *Sex* information from $\texttt{DB}_{\texttt{per}}$, which allows her to calculate $\mathrm{P_c}(\mathrm{CVD} \mid \mathrm{Age} = 45, \mathrm{Sex} = \mathrm{male}) = 49.4\%$. A second background knowledge [Ban+15] that describes the ability to infer activities performed by a customer based on the data coming from wearable sensors. She is then able to infer the *PAL* of the targeted customer by querying its sensor data from $\texttt{DB}_{\texttt{sen}}$ which enables her to update her knowledge $\mathrm{P_c}(\mathrm{CVD} \mid \mathrm{Age} = 45, \mathrm{Sex} = \mathrm{male}, \mathrm{PAL} = \mathrm{poor}) = 52.7\%$. Therefore, mining the *PAL* from the sensor data allows the attacker to infer the *CVD* without being detected by the InfDS which only keeps track of queries issued to $\texttt{DB}_{\texttt{per}}$.

In a more general context, depicted by Figure 1, where multiple services

are assumed to own at least a $\texttt{DB}_{\texttt{per}}$ and a $\texttt{DB}_{\texttt{sen}}$, besides InfDSs devoted to protecting data from inference attacks within personal databases (represented by the black locks on Figure 1), there is a need to:

(i) Detect attacks which leverage both non-sensitive personal data in $\texttt{DB}_{\texttt{per}}$ and knowledge mined from sensor data stored in $\texttt{DB}_{\texttt{sen}}$ (represented by the orange locks on Figure 1).

(ii) Prevent inference attacks occurring when a user acquires knowledge querying an external personal or sensor database (represented by the purple locks on Figure 1).

To the best of our knowledge, current InfDSs proposed in the state of the art only protect personal databases against local inference attacks and do not fulfill requirements (i) and (ii). In this paper, we analyze the requirements of an inference detection system able to detect inference attacks inside a personal database, inference attacks involving mined data from sensor databases and inference attacks involving external databases. We classify the identified issues into three categories:

- The modeling of inference channels related to sensor data and user knowledge gained by querying sensor databases.
- The optimization of the detection to cope with the huge amount of user knowledge, created by the continuous generation of sensor data.
- The detection of attacks distributed over multiple databases, while preserving the privacy of both individuals and services.

The remainder of this article is organized as follows: Section 2 formulates the issues and their respective challenges that must be dealt with in order to tackle the new inference attacks; Section 3 reviews the related works and describes the limits of each approach with respect to the issues; a conclusion and future research directions are provided in Section 4.

2 Issues & Challenges

In order to detect at query-time inference attacks involving sensor data, one should be able to reason on the existing dependencies that an attacker can leverage, both from databases containing sensor data and/or personal data. Hence, it requires capturing the knowledge gained by the users via queries in a representation which allows such reasoning. Moreover, the MD implies that the detection system must cope with heterogeneous databases owned by distinct services. In the following, we have categorized the issues into three main groups: the first one is related to the requirements a model must meet to represent the knowledge a user acquires after querying both personal and sensor data, based on data dependencies; the second one presents optimization issues that must be solved to detect inferences at query-time, in an acceptable time; the last group describes what an InfDS should achieve to process queries in the MD context.

Modeling issues Detecting inference attacks involving sensor data leverages some modeling challenges that we will detail in this section. To be able to exploit an inference channel implying raw data, the attacker

needs to acquire enough knowledge (e.g., for example a certain amount of consecutive measures). Consequently, a suitable InfDS should be able to model this new knowledge in a form that allows it to discover the attack. This raises several issues:

First, sensor data representation in the knowledge; data is of stream nature which means very big. It is observed that the data itself (i.e., measures) isn't so important for the detection but rather the amount of data acquired, the frequency of data and the time this data has been generated by the sensors. Thus, modeling the user knowledge about sensor data requires metadata representation (i.e., time window, frequency, and so on).

The second modeling challenge concerns metadata diversity due to the diversity of algorithms behind the inference channels and the data they capture and exploit. This requires a generic way to represent metadata and the capacity of the system to easily integrate new metadata representation, according to the advent of new inference channels with new metadata requirements.

Lastly, as shown through the motivating example, inference channels based on sensor data can be leveraged to acquire personal data (e.g., the PAL). Therefore, the new inference detection system should be able to reason thanks to a unique model integrating both personal and sensor data knowledge.

Efficiency issues In the next decade, with the advent of applications which manage sensor data of millions of users, services are expected to collect huge amount of data. In addition, the query rate on that data will be very high, which will impact the size of the knowledge managed by an InfDS supporting sensor data. In the meantime the query time should remain as short as possible. The main challenge is how to make the detection time the lowest to keep the query time acceptable.

Behind this challenge, several questions lurk: How would one organize the knowledge storage so that it would be efficient to retrieve? How could the detection algorithms be optimized to target only data required at each step? Is it interesting to consider a distributed inference detection system close to the sensors which generate the data, by exploiting new paradigms (e.g., Edge and/or Fog Computing paradigms)? Is it necessary to systematically launch the detection online, keeping in mind that only few queries will result in an attack situation? How could we categorize users so that only suspicious users/queries are controlled online?

Inference detection attacks issues in the MD context As explained above, detecting an inference attack in the MD context requires the InfDS to work on a data model that represents data dependencies both inside a database and between attributes of distinct databases.

The first faced issue is how to achieve a semantic matching between attributes of distinct databases. Indeed, the same information could be coded differently in two distinct databases (e.g., in one database the first and last name of an individual are stored as a single attribute while in the second database the first name and the last name are represented as

two distinct attributes). Furthermore, in the case where the databases are managed by distinct services, the inference detection task must be delegated to an external entity. This generates an additional issue, the external inference detection system needs to have access to (a) the databases schema to be able to build a data model that captures all the data dependencies among attributes, which implies disclosing the database structures and attributes. It also requires that (b) the user's knowledge is managed at the external inference detection system level, i.e., the purple locks in Figure 1. Detecting inferences in this case leads to disclosing data at the instance level. For example, we consider a simple situation where $\texttt{DB}^1_{per}$ and $\texttt{DB}^2_{per}$ are owned respectively by services $\texttt{S}^1$ and $\texttt{S}^2$ and we assume that $\texttt{DB}^1_{per}$ and $\texttt{DB}^2_{per}$ contain both the same content. Then, if a user u queries Age and Sex from $\texttt{DB}^1_{per}$, thanks to the external inference detection system, the PAL queried from $\texttt{DB}^2_{per}$ is denied as it would allow the CVD to be identified. However, the sensitive data is then disclosed to the detection system itself.

An alternative way to enforce detection without disclosing database schemas and instance information would be to enforce the detection at the side of each service. In fact, the local inference detection systems should check the inference condition locally and collaborate to update the other InfDSs about the knowledge the user should gain, in order to avoid answering queries in case of inference attacks. The challenge, in this case, is how to adapt privacy-preserving data exchange protocols (e.g., secure Multi-Party protocols) to the type of exchanged data in order to reach the distributed inference detection objective. Besides, the privacy it offers, this solution avoids the huge data exchange due to the database schema exchange and user knowledge at the instance level.

3 Related work

In this section, we review the related work in the field of inference detection solutions and raise their limitations to overcome the issues listed above. Most of the proposed solutions target single personal databases against inference attacks. We can distinguish systems that detect inference attacks at run-time from those which detect and remove inference channels by design.

In the first category, Chen et al. [CC06], Guarnieri et al. [GMB17], and Chang et al. [CM03] propose to tackle inference attacks by building models which represent probabilistic inference channels. While Chen et al. propose a mechanism reasoning on the probabilistic dependency among attributes in a single centralized database, Chang et al. propose a similar mechanism for a distributed personal database. Guarnieri et al. propose a system where one module acts as a policy decision point and another checks inference attempts. Moreover, the solution of Guarnieri et al. works under the assumption that only closed queries are issued to the database. All these solutions address inference attack detection considering no updates on the databases which is not suitable to the continuous sensor data generation. Both solutions of Chen et al. and Chang et al. do not offer any possibilities to represent dependencies between sensor data and personal

data, with the associated temporal and/or spatial constraints.

The system presented by Toland et al. [TFE10] models the functional dependencies within a database to compute the disclosed knowledge each time a query is issued. In their work, the functional dependencies are limited to logical dependencies. To the best of our knowledge, this work is the only one that considers database updates (i.e., tuple update, deletion or insertion), by storing the most recent updates in the query history log. This solution however focuses only on relational databases containing personal data and has not been designed to model sensor data.

Lachat et al. [LRB20] extend the work of Chen et al. in order to detect inference attacks exploiting multi-database inference channels using data linkage techniques. While their model assumes that the databases are static, contain personal data only, and cannot represent dependencies with temporal or contextual constraints, at the best of our knowledge, it is the only work which addresses the inference detection problem in the MD context.

In the second category, Noury et al. [NA19] propose an access control model which enforces value and temporal constraints in time series databases. In their context, an inference occurs when a new access control rule takes precedence on an older one, enabling dishonest users to infer sensitive values. Precedence conflicts are detected thanks to the static rules analysis. However, this model is not adapted to represent dependencies between time series data.

4 Conclusion

In this paper, we have motivated the need to prevent inference attacks in a multi-database context, extending the reasoning to sensor databases. Indeed, the pervasiveness of sensors provides attackers with new sources of data that, if exploited, could breach the privacy of individuals. We have highlighted the issues and challenges which must be tackled in order to propose a system detecting these new inference attacks. As a future work, we first plan to address the modeling of inference channels and user knowledge involving sensor data by using as a case study the MHEALTH dataset [Ban+15]. We then plan to propose an InfDS based on this first model, with a first set of optimizations to demonstrate the feasibility of our approach and to evaluate its performance.

References

[Ban+15] Oresti Banos et al. “Design, implementation and validation of a novel open framework for agile development of mobile health applications”. In: *BioMedical Engineering OnLine* 14.2 (2015), S6.

[CC06] Yu Chen and Wesley W. Chu. “Database Security Protection Via Inference Detection”. In: *Intelligence and Security Informatics*. Berlin, Heidelberg, 2006, pp. 452–458.

[CM03] LiWu Chang and Ira Moskowitz. “A study of inference problems in distributed databases”. In: *Research Directions in Data and Applications Security*. Springer, 2003, pp. 191–204.

[EE14] Günther Eibl and Dominik Engel. “Influence of data granularity on smart meter privacy”. In: *IEEE Transactions on Smart Grid* 6.2 (2014), pp. 930–939.

[FJ02] Csilla Farkas and Sushil Jajodia. “The inference problem: a survey”. In: *ACM SIGKDD Explorations Newsletter* 4.2 (2002), pp. 6–11.

[GMB17] Marco Guarnieri, Srdjan Marinovic, and David Basin. “Securing Databases from Probabilistic Inference”. In: *2017 IEEE 30th CSF Symposium*. 2017, pp. 343–359.

[Krö19] Jacob Kröger. “Unexpected Inferences from Sensor Data: A Hidden Privacy Threat in the Internet of Things”. In: *Internet of Things. Information Processing in an Increasingly Connected World*. Springer International Publishing, 2019, pp. 147–159.

[Kub+17] Yasuhiko Kubota et al. “Physical Activity and Lifetime Risk of Cardiovascular Disease and Cancer”. In: *Medicine and science in sports and exercise* 49.8 (2017), pp. 1599–1605.

[LRB20] Paul Lachat, Veronika Rehn-Sonigo, and Nadia Bennani. “Towards an Inference Detection System Against Multi-database Attacks”. In: *European Conference on Advances in Databases and Information Systems*. Springer. 2020, pp. 199–209.

[NA19] Amir Noury and Morteza Amini. “An access and inference control model for time series databases”. In: *Future Generation Computer Systems* 92 (2019), pp. 93–108.

[TFE10] Tyrone S. Toland, Csilla Farkas, and Caroline M. Eastman. “The inference problem: Maintaining maximal availability in the presence of database updates”. In: *Computers & Security* 29.1 (2010), pp. 88–103.

[WB10] Philip Woodall and Pearl Brereton. “A systematic literature review of inference strategies”. In: *International Journal of Information and Computer Security* 4.2 (2010), pp. 99–117.