Poster Paper

Privacy Protection through Query Rewriting in Smart Environments *

Hannes Grunert
Database Research Group
University of Rostock
18051 Rostock, Germany
hg(at)uni-rostock.de

Andreas Heuer
Database Research Group
University of Rostock
18051 Rostock, Germany
ah(at)uni-rostock.de

ABSTRACT

By the events in the past years, the integration of data protection mechanisms into information systems becomes a central research problem again. In this poster, we show how query rewriting can be used to maintain privacy of users in smart (or assistive) environments. We developed a privacy respecting query processing and a vertical fragmentation of queries, processing maximal parts of the query as close to the sources of the data (e.g. sensors) as possible.

Categories and Subject Descriptors

H.2.4 [**Database Management**]: Systems—*Query Processing*; K.4.1 [**Computer and Society**]: Public Policy Issues—*Privacy*

General Terms

Privacy Enhancing Technologies, Database Systems

1. PRIVACY

Smart Metering, Internet surveillance, motion profiles, biometric databases, data retention: In the digital world steadily more and more information about ourselves and our environment is collected. Besides "classical" personal information, such as the name, age or gender, a plurality of sensors records our activities and inclinations. Active and passive RFID tags, cameras, microphones, but also sensors on light switches and power sockets capture the current situation in the ubiquitous environments, up to 100 times per second.

Especially smart environments such as assistive systems using activity and intention recognition [4] are a possible cause of privacy violations, especially if the query realizing the recognition analysis is performed on a cloud server.

To reduce privacy violations, it is necessary

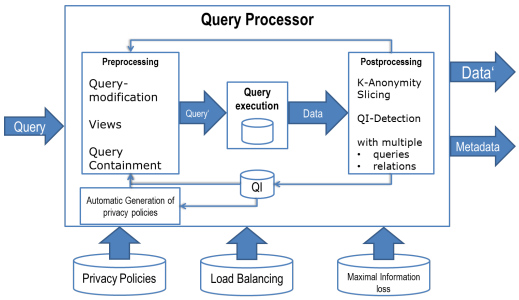

Figure 1: The concept of the privacy-aware query-processor.

- to decrease collected personal information, i.e. to apply the principle of data avoidance (except where data are required),
- to process data with personal references as less as possible or — at least — as close to the local data sources (sensors) as possible and
- to anonymize, pseudonymize and delete personal data, unless it is used for further processing and necessary to realize the aim of the assistive environment.

Data minimization and data avoidance are therefore prescribed, indispensable requirements for the design of smart systems. This requirement can be achieved in databases by transforming both queries and query results, as well as using views [3].

2. PRIVACY-AWARE QUERY PROCESSING

The PArADISE[1]-approach aims to withdraw the burden of respecting privacy constraints from the assistive systems by adding privacy protection mechanisms to those systems storing and analyzing the data: database systems on different levels. PArADISE combines performance aspects of big data analytics (by using massively parallel database technology [5]) and privacy protection. Our privacy-aware query-processor (see Figure 1) generates anonymized result sets. These data maintain a high degree of value for the initial query generated by the assistive system. On the opposite, additional knowledge can hardly be derived.

*A full version of this paper is available as a Technical Report at `www.ls-dbis.de/digbib/dbis-tr-cs-01-16.pdf`

[1]Privacy Aware Assistive Distributed Information System Environment

The preprocessor allows the analysis and the rewriting of database queries regarding user-defined privacy policies [2]. During the execution of the request, it is decided whether the request will be answered and anonymized directly on the current network peer, or is sent to lower nodes (vertical fragmentation, see below). The postprocessor executes the anonymization of the query results, taking into account various criteria of quality and privacy. For this, several data protection metrics and algorithms are provided. The module for the automatic generation of privacy settings produces and adapts existing user-defined privacy policies to new devices and changing requirements and queries.

3. QUERY REWRITING BY VERTICAL FRAGMENTATION

The smart environment or assistive system sends a query request Q to the database d integrating the entire sensor data recorded in our environment. The result of Q is needed to perform the activity and intention recognition. The data sources are sensors being located in appliances in apartments and buildings. Instead of shipping d to the cloud server sending the request, maximal parts of Q will be evaluated as close to the sensor as possible. As can be seen in Figure 2, instead of performing $Q(d)$ in the cloud, the maximal subquery Q_j will be shipped to the next lower node of the processing chain, in the case of the example a PC located in our apartment. While Q performs an iterative machine learning algorithm implemented in R and SQL, and Q_j being a complex SQL query with recursion, the lowest node in the processing chain (the sensors) can only compute some filter mechanisms (simple selections) and some simple aggregations over the last values generated (window function: average of last minute). Each of the nodes will ship the query result d_j to the node sending the request. After a final anonymization step A, the data "leaving our apartment" d' will only be a small subset of the original data d.

We assume that the lower nodes will each have less query computing power than the higher nodes: while sensors are only performing simple filters / selections and aggregations, an appliance like a TV or a smart network music player will be able to perform simple (SQLsuperlight) database queries, an Android-based home media server even more complex SQL queries.

The whole query processing procedure

$$Q(d) := d_j = Q_j(\dots d'_i = A(d_i = Q_i(\dots (d_1 = Q_1(d)) \dots)$$

is transformed in a way, that the cloud server will perform a remainder query Q_δ on d' instead of performing Q on d, hence resulting in the privacy protecting query rewriting $Q(d) \longrightarrow Q_\delta(d')$. In other words, the query Q is fragmented in queries Q_j (that can be performed at a lower node) and a remainder query Q_δ that can only performed at the more powerful node of our vertical fragmentation of query processors.

Since recognizing the maximal SQL queries in an R machine learning algorithm is undecidable in general, we try to detect some larger "SQLable" patterns in the activity and intention recognition procedures described in [4]. Other aspects of our privacy-aware query-processor are described in, e.g. [1].

By rewriting the query Q into Q_j and Q_δ and only performing Q_δ outside our "privacy protected" appliance ensemble in our apartment, we hope to automatically prevent the service provider of our assistive system to use our personal data in a way we did not consider to be possible when starting to use his smart service. A remaining open problem is to decide whether a privacy-violating query $Q_\downarrow$ can be performed even on d' instead of d. In this case, we have to extend the anonymization step A already performed. This open problem results in a query containment problem of $Q, Q_\downarrow$ and Q_j that will be part of our further research.

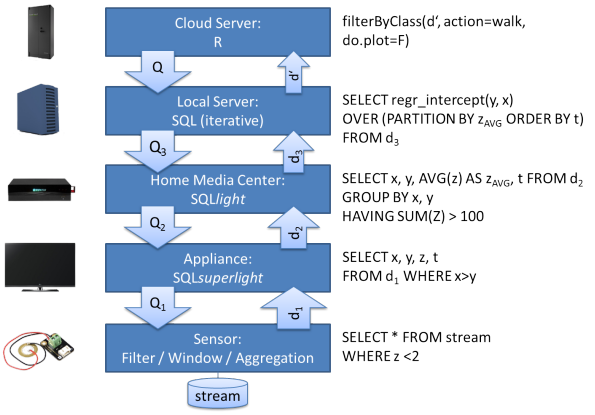

Figure 2: Vertical Query Fragmentation: Query and query result transformation on different peers.

4. ACKNOWLEDGMENTS

Hannes Grunert is funded by the German Research Foundation (DFG), Graduate School 1424 (Multimodal Smart Appliance Ensembles for Mobile Applications - MuSAMA). The authors gratefully acknowledge the constructive comments of the anonymous referees.

5. REFERENCES

[1] H. Grunert. Distributed denial of privacy. In *INFORMATIK 2014: Big Data Komplexität meistern*, pages 2299–2304. Springer, 2014.

[2] H. Grunert and A. Heuer. Generating privacy constraints for assistive environments. In *Proceedings of the 8th International Conference on PErvasive Technologies Related to Assistive Environments, PETRA 2015*. ACM, 2015.

[3] A. Heuer and A. Lubinski. Data reduction - an adaptation technique for mobile environments. In *Interactive Applications of Mobile Computing (IMC'98)*, 1998.

[4] F. Krüger, M. Nyolt, K. Yordanova, A. Hein, and T. Kirste. Computational State Space Models for Activity and Intention Recognition. A Feasibility Study. *PLOS ONE*, Nov. 2014. 9(11): e109381. doi:10.1371/journal.pone.0109381.

[5] D. Marten and A. Heuer. A framework for self-managing database support and parallel computing for assistive systems. In *Proceedings of the 8th International Conference on PErvasive Technologies Related to Assistive Environments, PETRA 2015*. ACM, 2015.