Extending SQL to Support Privacy Policies

Kambiz Ghazinour, Sampson Pun, Maryam Majedi, Amir H. Chinæi, and Ken Barker

Abstract Increasing concerns over Internet applications that violate user privacy by exploiting (back-end) database vulnerabilities must be addressed to protect both customer privacy and to ensure corporate strategic assets remain trustworthy. This chapter describes an extension onto database catalogues and Structured Query Language (SQL) for supporting privacy in Internet applications, such as in social networks, e-health, e-government, etc. The idea is to introduce new predicates to SQL commands to capture common privacy requirements, such as purpose, visibility, generalization, and retention for both mandatory and discretionary access control policies. The contribution is that corporations, when creating the underlying databases, will be able to define what their mandatory privacy policies are with which all application users have to comply. Furthermore, each application user, when providing their own data, will be able to define their own privacy policies with which other users have to comply. The extension is supported with underlying catalogues and algorithms. The experiments demonstrate a very reasonable overhead for the extension. The result is a low-cost mechanism to create new systems that are privacy aware and also to transform legacy databases to their privacy-preserving equivalents. Although the examples are from social networks, one can apply the results to data security and user privacy of other enterprises as well.

Kambiz Ghazinour, Sampson Pun, Maryam Majedi and Ken Barker,
University of Calgary,
Department of Computer Science, Calgary, AB, Canada
e-mail: {kghazino,pszypun,mmajedi,kbarker}@ucalgary.ca

Amir H. Chinæi,
University of Puerto Rico (Mayagüez campus), Department of Electrical and Computer Engineering, Mayagüez, PR, USA.
e-mail: ahchinaei@ece.uprm.edu
1 Introduction

The Structured Query Language (SQL) is at the core of millions of database applications, many of which were implemented during the past decade. This chapter introduces an important extension onto SQL, which gives increased expressive power to its security model so it can support today’s privacy requirements. The extension is simple to understand, easy to implement, and inexpensive to maintain.

Before proceeding to next sections, this work is motivated with running examples from the area of online social networks. (Social networks are good examples of applications that require user-centricity and are very vulnerable to privacy violations.) Even though current social networks, such as Facebook, Flickr and My-Space, publish a privacy policy indicating the usage of personal information within their network, they do not provide a system-level or user-centric protection to enforce the policies. Facebook and Flickr have chosen to join a third party auditing firm to resolve disputes once they occur, and MySpace relies on its internal policy to enforce such rules. Currently, other social networks do not provide more effective privacy enforcement either. This chapter proposes a user-centric enforcement of privacy policies applicable to a variety of web applications, such as social networks, e-health, e-government, etc. The proposal extends the nearly ubiquitous SQL language but the core ideas are readily extendible to other modern database languages, such as XQuery. The balance of this section illustrates the extension’s requirements.

1.1 Requirements for Extension

Similar to Rosenthal and Sciore’s contributions [10], three criteria during the extension are considered: modularity, compatibility, and simplicity. Modularity allows system vendors to provide a straightforward implementation of their systems to support the additional privacy provided by this increment. Furthermore, it helps system administrators and end-users to gradually adopt the new features at their discretion. Backwards compatibility is critical when transforming legacy systems to their privacy-preserving equivalents. It also provides an easier implementation because it is built on existing technology. Simplicity, which is partially ensured by modularity and compatibility, is a key characteristic for the extension to be accepted by the market. These three criteria together make the advances simple to understand, easy to implement, and inexpensive to maintain.