ABSTRACT

With the development of cloud computing and big data, data privacy protection has become an urgent problem to solve. Data encryption is the most effective way to protect privacy; however, it will change the data format and result in: 1. database structure and application software will be changed; 2. structured query language (SQL) operations cannot work properly, especially in SQL-based fuzzy query. As a result, it is necessary to provide an SQL-based fuzzy query mechanism over encrypted databases, including traditional databases and cloud outsourced databases. This paper establishes a secure database system using format-preserving encryption (FPE) as the underlying primitive to protect the data privacy while not change the database structure. It further proposes a new SQL-based fuzzy query mechanism supporting directly query over encrypted data, which is constructed by FPE and universal hash function (UHF). The security of the proposed mechanism is analyzed as well. In the end, it makes extensive experiments on the system to demonstrate its practical performance.

Keywords: Data Privacy, Format-Preserving Encryption (FPE), Fuzzy Query, Outsourced Database, Structured Query Language (SQL)

1. INTRODUCTION

Recently, privacy of outsourced database has attracted more and more attentions. Encryption is the most effective way for data privacy protection, but it raises at least two challenges: 1. data encryption may change the data type and length, resulting in that it is hard to impose encryption mechanism over existing database. For example, a phone number, encrypted using the AES encryption algorithm, not only does not resemble a phone number but will even not
contain any numbers at all. A database field typically defined to hold an eleven-character phone number would not be able to store the AES-encrypted version of data; 2. data encryption may lead to hardly performing structured query language (SQL) operations over encrypted data.

For the first challenge, a novel primitive namely format-preserving encryption (FPE) making sure that ciphertext has the same format as plaintext was proposed. In other words, using this technique, the data encryption will not change the data type and length, so ciphertext can be stored in the original database. Since it was proposed, several solutions to FPE have been investigated. In 2002, Black (2002) formalized the FPE problem and proposed three basic methods to implement such cipher. After 2008, some FPE schemes (Morris, 2009; Bellare, 2010; Liu, 2010; Li, 2012) were proposed to provide format-preserving encryption for different domains like integer, datetime, etc.

For the second challenge, several cryptographic tools were developed to provide solutions for operating directly on encrypted data, such as order-preserving encryption (Agrawal, 2004; Boldyreva, 2009; Popa 2013) for encryption while preserving orders of ciphertext, homomorphic encryption (Van Dijk, 2010) for allowing function computations directly imposed on ciphertexts, searchable encryption for searching keywords over encrypted data (Song, 2000; Curtmola, 2006; Li, 2010) and query processing over encrypted databases (Kwok 2002; Wang, 2005; Yang, 2006; Ashrafi, 2007; Amanatidis, 2007; Evdokimov, 2007; Taniar, 2008). Although these techniques have been developed, fuzzy query over encrypted data is still a challenge.

In this paper, motivated by the fact that sensitive information (e.g., name, account, password, e-mail and address) typically existing in the form of character string, is usually required to be queried, we take the focus on how to perform SQL query (especially fuzzy query due to the practical utility) over encrypted character string. Actually, there are some related works concerning on this point, which are listed in Table 1. We argue that these works are impractical for SQL-based fuzzy query for the following reasons:

- To apply encryption techniques to achieve the goal of query over existing database application, one important issue is not to change the database structure. For changing the data type and length of the plaintext, encryption will lead to a large cost on adapting the applications. Many of existing works (e.g., Hacigumus (2002), Yang (2006) and Popa (2013)) cannot satisfy this requirement;
- Though some methods (Wang, 2005; Amanatidis, 2007; Evdokimov, 2007) do not change the database structure, Amanatidis (2007) and Evdokimov (2007) only provide solutions to exact matching, and Wang (2005) supports partial fuzzy query, i.e., it is only able to locate range for results;
- Searchable encryption schemes (Song, 2000; Curtmola, 2006; Li, 2010; Wang,

<table>
<thead>
<tr>
<th>Method</th>
<th>Require Index</th>
<th>Exact Query</th>
<th>Fuzzy Query</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hacigumus (2002)</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Amanatidis (2007)</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Evdokimov (2007)</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Wang (2005)</td>
<td>No</td>
<td>Yes</td>
<td>Partial</td>
</tr>
<tr>
<td>Yang (2006)</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Popa (2013)</td>
<td>Yes</td>
<td>Yes</td>
<td>Partial</td>
</tr>
</tbody>
</table>
Related Content

Visual Mobility Analysis using T-Warehouse
[www.igi-global.com/article/visual-mobility-analysis-using-warehouse/49638?camid=4v1a](www.igi-global.com/article/visual-mobility-analysis-using-warehouse/49638?camid=4v1a)

Fuzzy Miner: Extracting Fuzzy Rules from Numerical Patterns
[www.igi-global.com/article/fuzzy-miner-extracting-fuzzy-rules/1748?camid=4v1a](www.igi-global.com/article/fuzzy-miner-extracting-fuzzy-rules/1748?camid=4v1a)
The Application of Data Mining for Drought Monitoring and Prediction
Tsegaye Tadesse, Brian Wardlow and Michael J. Hayes (2009). Data Mining Applications for Empowering Knowledge Societies (pp. 278-289).
www.igi-global.com/chapter/application-data-mining-drought-monitoring/7557?camid=4v1a

A Parallel Implementation Scheme of Relational Tables Based on Multidimensional Extendible Array
www.igi-global.com/article/parallel-implementation-scheme-relational-tables/1775?camid=4v1a