Journal of Network and Computer Applications ■ (■■■■) ■■■–■■■

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

# New order preserving encryption model for outsourced databases in cloud environments

Zheli Liu [a], Xiaofeng Chen [b], Jun Yang [a], Chunfu Jia [a], Ilsun You [c,*]

[a] College of Computer and Control Engineering, Nankai University, China
[b] State Key Laboratory of Integrated Service Networks, Xidian University, China
[c] School of Information Science, Korean Bible University, South Korea

ARTICLE INFO

*Article history:*
Received 18 October 2013
Received in revised form
7 June 2014
Accepted 7 July 2014

*Keywords:*
Order preserving encryption
Outsourced database
Privacy protection
Cloud computing
Ciphertext-only attack

ABSTRACT

The order of the plaintext remains in the ciphertext, so order-preserving encryption (OPE) scheme is under threat if the adversary is allowed to query for many times. To hide the order in the ciphertext, the only ideal-security OPE scheme (Popa et al., 2013) requires the database server to maintain extra information and realize comparison or range query by user defined functions (UDFs). However, order operations will no longer be performed directly on the ciphertext. It will affect the efficiency and make this scheme to be not suitable for some cases.

In this paper, we aim at constructing efficient and programmable OPE scheme for outsourced databases. Firstly, we introduce the system model of outsourced database where OPE scheme will be used, show that ciphertext-only attack is basic and practical security goal. Secondly, we discuss the statistical attack for OPE schemes, point out how to hide data distribution and data frequency is important when designing OPE schemes. Thirdly, we propose a new simple OPE model, which uses message space expansion and nonlinear space split to hide data distribution and frequency and further analyze its security against two kinds of attack in details. Finally, we discuss implementation details including how to use our OPE scheme in the database applications. And we also evaluate its performance through the experiment. The security analysis and performance evaluation show that our OPE scheme is secure enough and more efficient.



## 1. Introduction

Order-preserving encryption (OPE) is a common encryption scheme which ensures that the order of plaintexts remains in the ciphertexts. It is appealing because systems can perform order operations on ciphertexts in the same way as on plaintexts: for example, a database server can build an index, perform SQL range queries, and sort encrypted data, all in the same way as for plaintext data. This property results in good performance and requires minimal changes to existing software, making it easier to adopt.

In the cloud computing and big data environments, OPE will be more useful, because: (1) outsourced database has attracted much attention recently due to the emergence of cloud computing, however, how to protect the outsourced data storing in the untrusted cloud server becomes a serious problem. Since order-preserving, OPE allows untrusted server to perform database operations, such as comparison and range query over encrypted data, without decrypting them; (2) in the big data environment, a fruitful direction for future research in data mining will be the development of cryptology techniques that incorporate privacy concerns. For most of the data mining algorithms usually rely on the order of data, OPE will be also the ideal tool when to protect data privacy using the cryptographic techniques and ensure the right results can be mined.

The ideal security goal for an OPE scheme, IND-OCPA (Boldyreva et al., 2009), is to reveal no additional information about the plaintext values besides their order (which is the minimum requirement for the order-preserving property). Until now, the only ideal-security OPE scheme is mutable order-preserving encoding (mOPE) scheme (Popa et al., 2013), which is proposed by Popa et al. in 2013, where the ciphertexts reveal nothing except for the order of the plaintext values. The mOPE works by building a balanced search tree containing all of the plaintext values encrypted by the application in the database side, and it requires the encryption protocol to be interactive and for a small number of ciphertexts of already-encrypted values to change as new plaintext values are encrypted (e.g., it is straightforward to update a few ciphertexts stored in a database), and these operations in database side can be implemented by user define functions (UDFs).

It has been a problem of OPE that how to improve security but ensure the function and the efficiency. Although mOPE has ideal

* Corresponding author.
*E-mail address:* ilsunu@gmail.com (I. You).

http://dx.doi.org/10.1016/j.jnca.2014.07.001
1084-8045/

security, but the interaction and tree balancing will affect its efficiency, besides, UDFs and the maintained balance tree make it be not suitable for the cases in which: (1) user has no permission to create UDFs in the database, for example, some small companies deploy their web applications into the rented web server using the rented database; (2) the application requires the direct order comparison on the ciphertext, for example, the OPE is used to achieve privacy-preserving data publishing for special data mining task. Another scheme (Boldyreva et al., 2009) has provable security guarantees: the encryption is equivalent to a random mapping that preserves order, however, the experiment in Popa et al. (2011) shows that it has a poor efficiency and its execution time of encryption is 9 ms. Except them, some other OPE schemes (Kadhem et al., 2010; Seungmin et al., 2009; Yum et al., 2012) have been proposed, however, they all leak more information than just the order of values. Thus, it is always necessary to propose an efficient OPE scheme with the practical security.

In this paper, we aim at proposing feasible, programmable and secure OPE scheme which is practical on the outsourced database or privacy-preserving data publishing (Fung et al., 2010). In particular, we assume that: (1) the database should support the direct order comparison on the ciphertext, i.e., the ciphertext should also be numerical data; (2) the new OPE scheme should have a good performance and lead to minimal change for existing software, and the ideal OPE ciphertext can be stored in the original field; (3) the basic security goal for outsourced database is against the ciphertext-only attack, besides, the security against chosen-plaintexts attack can also be achieved if we make some restriction of the database system due to the different scene.

## 2. System model

In this section, we will briefly discuss the system model for database applications based on the cloud storage, where OPE scheme will be applied, and further discuss its adversary model.

### 2.1. Basic model

As shown in Fig. 1, there are three different roles in the model, which are *owner*, *cloud database service provider* and *application server*.

- *Cloud database service provider*: It is the service provider, who provides the cloud storage service and allows paying customers to store their application data. It helps customers to reduce the management and maintenance cost, and avoids purchasing expensive hardware and database software. However, it must be untrusted and can be defined as “honest but curious”, i.e. it is interested in the users' private data.
- *Owner*: It is the data provider, who stores data to the rented cloud database.
- *Application server*: It is not the necessary role in our model, but database applications based on three layer architecture usually use it to process business operation. For database applications based on “client/server” model, the owner and application server will be the same role. So, in our model, we assume that *application server* is trusted as the *owner*, and we call them as “OPE client”.

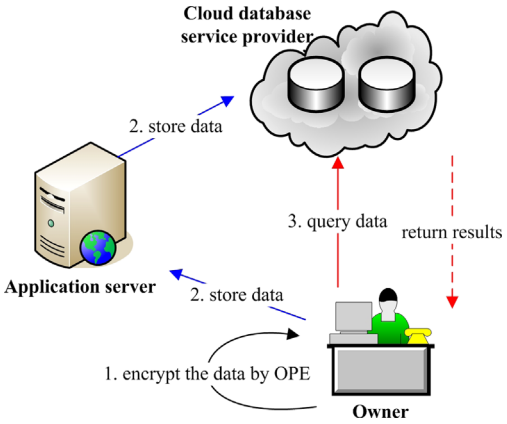


**Fig. 1.** System model for outsourced database.

There are also two data flows in the model, i.e., *storing data* and *querying data*.

- *Storing data*: To store data, data owner should firstly use OPE to encrypt the data which needs for preserving order in the OPE client (owner or application server), and then store the ciphertext to the cloud.
- *Querying data*: To perform a query, data owner should firstly use OPE to encrypt the keyword in the range query or exact query SQL sentence, and then send the new SQL query to the cloud. The cloud database can directly execute the SQL sentence and return the results to the OPE client.

**Notice**: In our system model, the OPE operations are happened in OPE client, but comparison or range query can be directly supported by the database server. And thus, the OPE scheme will be suitable for privacy-preserving data publishing.

### 2.2. Adversary model

If the OPE encryption executes in the *application server* side, we assume that sufficient access control or other effective methods are applied, to make sure the *application server* will not leak the key information.

We consider two types of attackers:

1. Attackers have access rights of database, such as DBA or cloud service provider of outsourced database. They can see encrypted data, database structure, but can only launch ciphertext-only attack. The security against such attackers is the basic and practical security notion.
2. Attackers have access rights of both application system and database, who can access SQL interpretation interface deployed in database applications, construct SQL sentences with plaintext, gain interpreted SQL sentences with encrypted data, view all fields and structure of database. They have more information to guess the encryption details. They can launch chosen plaintext or ciphertext attacks, in order to guess encryption key. The security against such attackers is the advanced security notion.

In the practical applications, the main threat is the first type attacker. In this case, curious attacker is easy to get the data storing in database, but he is difficult to get encryption key which can be protected by the cryptographic method. So that the security against the ciphertext-only attack is our basic and practical security goal.

## 3. Related works

In this section, we will make a summary on the related works, discuss statistical attack for OPE schemes and introduce two typical OPE schemes.

Please cite this article as: Liu Z, et al. New order preserving encryption model for outsourced databases in cloud environments. Journal of Network and Computer Applications (2014), http://dx.doi.org/10.1016/j.jnca.2014.07.001