A SEMANTICALLY SAFE ENCRYPTION ANALYSIS-BASED SKYLINE QUERY PROTOCOL FOR DATA ENCRYPTION

[1]Dr.K.Murali Babu, [2]K. Venkateswarao, [3]Junna Sirisha, [4]Beena Thinvesh
[1]Professor, [2,3]Assistant Professor, [4]Student, Dept. of Computer Science Engineering, Newton's Institute of Engineering, Macherla, Andhra Pradesh, India.

Abstract
To facilitate large scale data storage and query processing, it is cost-effective to outsource data and computation to cloud servers. Sensitive data, such as medical information, must be shielded from the cloud server and other unauthorized users owing to security and privacy issues. One strategy is to send encrypted data to the cloud server and have it only conduct queries on the encrypted data. Supporting multiple queries over encrypted data in a safe and effective manner that prevents the cloud server from learning about the data, query, and query result is still a difficult problem. The issue of safe skyline queries over encrypted data is examined in this work. The skyline query is particularly important for multi-criteria decision making but also presents significant challenges due to its complex computations. We propose a fully secure skyline query protocol on data encrypted using semantically-secure encryption. As a key subroutine, we presenta new secure dominance protocol, which can be also used as a building block for other queries.
Keywords – Sky line, Efficient, Encrypted Data

INTRODUCTION

There are more and more cloud services available because to the quick development of cloud computing. Each offers various service levels, costs, and access methods. Prior to employing cloud services, selecting the best ones requires some thought. Once a cloud customer chooses a cloud service, it is difficult and expensive to move to a different cloud service provider in the traditional cloud computing environment. Inter cloud has been suggested as a solution to this issue of vendor lock-in and to promote more collaborative cloud services. Utilizing the services of other clouds, cloud service providers can process user requests under the inter-cloud paradigm. The infrastructure of cloud service providers can be shared to increase overall resource usage.

Furthermore, applications can be migrated from one cloud service provider to another cloud service provider and workloads can be distributed among clouds for disaster recovery or multi-region application delivery. In this paper, we consider an Intercloud system based on the IEEE P2302 Draft Standard, which employs a three tier architecture, namely, root, exchanges and clouds. The root is a cluster of servers/clouds providing certification and naming services. The clouds provide cloud services to users and to each other. Like Internet exchanges, Intercloud exchanges mediate between the root and clouds. Each cloud should belong to at least one Intercloud exchange. The root, Intercloud exchanges and clouds can communicate with one another through Intercloud gateways by means of Extensible Markup Language (XML)-based messages.

The basic Intercloud system can also be extended to support a mobile Intercloud system. In this case, heterogeneous clouds can work collaboratively under a mobile environment so that data, applications and virtual mobile terminals can move across clouds through various handoff processes. In the Intercloud environment, cloud service selection can be made in an ad-hoc, dynamic and distributed manner. For instance, onecloud may want to select a number of reliable clouds to help run a time-consumingprogram. For mobile Intercloud, a mobileuser may want to select a cost-effective cloud service in a foreign city. This makes cloud service selection in an Intercloud environment more challenging.

The trustworthiness of cloud services is an important consideration for making cloud selection decision (i.e., knowing the expected performance of a cloud service).Currently, there has been little work done to study distributed trust evaluation for the Intercloud environment. This paper seeks to contribute to this important topic for the development of Intercloud. Trust in a service is generally concerned with a belief in whether

the service can be delivered satisfactorily, in accordance with certain trust attributes. In the Intercloud context, a cloud service provider (or user) typically trusts another cloud service provider based on certain trust attributes, such as servicereliability, quality of service and service efficiency. Before choosing/using a service, trust evaluation is often conducted based on the feedback of existing users (i.e., reputation based trust evaluation). Indeed,feedback provided by past cloud users is a good reference for trust evaluation.

Based on this feedback or rating, a cloud user can evaluate how likely (e.g., a probability) that a cloud service will be performed as expected. However, thecredibility of feedback is often difficult to guarantee as cloud users often avoid leaving honest comments, especially negative ones. The main reason for this behavior is the unequal status between cloud service providers and cloud users (e.g., a cloud service provider can easily remove negative comments about its services). This problem becomes more serious in the Intercloud environment. As there is more and more mutual co-operation, a cloud user or his/her business could be another type of cloud service provider in future business transactions. This possible mutual relationship makes the privacy requirement even more important in the Intercloud scenario.

If feedback information cannot be made private, cloud users may only give positive feedback, as they want to maintain a good relationship or are fearful of retaliation. Hence, it is important to develop an effective and flexible trust evaluationprotocol with privacy protection for Intercloud.

BACKGROUND WORK

The skyline computation problem was first studied in computational geometry field where they focused on worst-case time complexity. In previous proposed output- sensitive algorithms achieving O(nlogk) in worst-case where k is the number of skyline points which is far less than n in general. Since the introduction of the skyline operator by B¨orzs¨onyi et al., skyline has been extensively studied in the database field. Kossmann et al. studied the progressive algorithm for skyline queries.

Different variants of the skyline problem have been studied.

Secure query processing on encrypted data. Fully homomorphic encryption schemes enable arbitrary computations on encrypted data. Even though it is shown that we can build such encryption schemes with polynomial time, they remain far from practical even with the state of the art implementations. Many techniques have been proposed to support specific queries or computations on encrypted data withvarying degrees of security guarantee andefficiency. We are not aware of any formal work on secure skyline queries over encrypted data with semantic security. Botheet al. and their demo version illustrated an approach about skyline queries on so-called “encrypted” data without any formal security guarantee. Another work studied theverification of skyline query result returned by an untrusted cloud server.

PROPOSED WORK

In this section, we present a set of secure sub-protocols for computing basic functions on encrypted data that will be used to construct our secure skyline query protocol. All protocols assume a two-party setting, namely, C1 with encrypted input and C2 with the private key sk as shown in Figure 3.The goal is for C1 to obtain an encrypted result of a function on the input without disclosing the original input to either C1 or C2. We note that this is different from the traditional two-party secure computationsetting with techniques such as garbled circuits where each party holds a private input and they wish to compute a function ofthe inputs. For each function, we describe the input and output, present our proposed protocol or provide a reference if existing solutions is available. Due to limited space, we omit the security proof which can be derived by the simulation and composition theorem in a straightforward way.

We first propose a basic secure skyline protocol and show why such a simplesolution is not secure. Then we propose a fully secure skyline protocol.

The general idea of Algorithm 1 is to first map the data points to the new space with the query point as origin (Lines 1-3). Given the new data points, it computes the sum of all attributes for each tuple S (ti) (Line 6) and chooses the tuple tmin with smallest S (ti) as a skyline because no other tuples can dominate it. It then deletes those tuples dominated by tmin. The algorithm repeats this process for the remaining tuples until an empty dataset T is reached.

Algorithm 1: Skyline Computation.

```
input : A dataset P and a query q.
output: Skyline of P.
for i = 1 to n do
    for j = 1 to m do
        t_i[j] = |p_i[j] − q[j]|;
while the dataset T is not empty do
    for i = 1 to size of dataset T do
        S(t_i) = Σ_{j=1}^{m} t_i[j];
        choose the tuple t_min with smallest S(t_i) as a skyline;
        add corresponding tuple p_min to the skyline pool;
        delete those tuples dominated by t_min from T;
        delete tuple t_min from T;
return skyline pool;
```

As mentioned in Algorithm 1, given a skyline query q, it is equivalent to compute the skyline in a transformed space with the query point q as the origin and the absolute distances to q as mapping functions. Hence we first show a preprocessing step inAlgorithm 2 which maps the dataset to the new space.

Algorithm 2: Preprocessing.

```
input : C1 has E_pk(P), C2 has sk, and the client has q.
output: C1 obtains the new encrypted dataset E_pk(T).
Client:
send (E_pk(−q[1]), ..., E_pk(−q[m])) to C1;
C1:
for i = 1 to n do
    for j = 1 to m do
        E_pk(temp_i[j]) = E_pk(p_i[j] − q[j]) =
        E_pk(p_i[j]) × E_pk(−q[j]) mod N^2;
C1 and C2:
use SM protocol to compute E_pk(T) = (E_pk(t_1), ..., E_pk(t_n))
only known by C1, where
E_pk(t_i) = (E_pk(t_i[1]), ..., E_pk(t_i[m])) and
E_pk(t_i[j]) = E_pk(temp_i[j]) × E_pk(temp_i[j]);
```

Fully Secure Skyline Protocol

The basic protocol clearly reveals severalinformation to C1 and C2 as follows.

When selecting the skyline tuple with minimum attribute sum, C1 and C2 know which tuples are skyline points,which violates our result privacy requirement.

When eliminating dominated tuples, C1 and C2 know the dominance relationshipamong tuples with respect to the query tuple q, which violates our data pattern privacy requirement.

To address this leakage, we propose a fully secure protocol in above Algorithm. The step to compute minimum attribute sum and return the results to the client are the same as the basic protocol. We focus on the following steps that are designed to address the disclosures of the basic protocol.

RESULT ANALYSIS

The overall run time complexity depends on the number of points (n), the number of skyline points (k), the number of decomposed bits (l) which is determined by the domain of the attribute values, and the number of dimensions (m). Astraightforward way to enhance the performance is to partition the input dataset into sub datasets and then we can use a divide-and-conquer approach to avoid unnecessary computations.

Furthermore, the partitioning also allows active parallelism. The basic idea of datapartitioning is to divide

the dataset into a set of initial partitions, compute the skyline in each partition, and then continuously merge the skyline result of the partitions into new partitions and compute their skyline, until all partitions are merged into the final result. This can be implemented with either a single thread (sequentially) or multiple threads (in parallel). We describe our data partitioning scheme. Given an input dataset, the number of partitions s is specified as one parameter.

CONCLUSION

According to the semi-honest concept, this research presented a totally secure skyscraper protocol on encrypted data utilizing two non-colluding cloud servers. By preventing the cloud server from knowing anything about the data, including indirect data patterns, queries, or query results, it maintains semantic security. Additionally, neither the client nor the data owner is required to take part in the calculation. We also provided a secure dominance mechanism that is applicable to both skyline and other types of queries. To further lessen the computational strain, we also presented two optimizations: data splitting and lazy merging. Finally, we described the protocol's application and illustrated its viability and effectiveness.

REFERENCES

[1] F. Baldimtsi and O. Ohrimenko. Sorting and searching behind the curtain. In FC 2015, pages 127–146, 2015.

[2] A. Beimel. Secret-sharing schemes: a survey. In International Conference on Coding and Cryptology, pages 11–46. Springer, 2011.

[3] J. L. Bentley. Multidimensional divide- and-conquer. Commun. ACM, 23(4):214–229, 1980.

[4] J. L. Bentley, H. T. Kung, M. Schkolnick, and C. D. Thompson. On the average number of maxima in a set of vectors and applications. J. ACM, 25(4):536–543, 1978.

[5] S. B¨orzs¨onyi, D. Kossmann, and K.Stocker. The skyline operator. In ICDE 2001.

[6] S. Bothe, A. Cuzzocrea, P. Karras, and A. Vlachou. Skyline query processing over encrypted data: An attribute-order- preserving-free approach. InPSBD@CIKM, pages 37–43, 2014.

[7] S. Bothe, P. Karras, and A. Vlachou.eskyline: Processing skyline queries overencrypted data. PVLDB, 6(12):1338– 1341, 2013.

[8] C. Y. Chan, H. V. Jagadish, K.-L. Tan,
A. K. H. Tung, and Z. Zhang. Finding k- dominant skylines in high dimensional space. In SIGMOD Conference, pages503–514, 2006.