2017 IEEE Second International Conference on Data Science in Cyberspace

Range Queries on Two Column Data

Ce Yang, Weiming Zhang and Nenghai Yu
CAS Key Laboratory of Electro-magnetic Space Information
University of Science and Technology of China
yangce@mail.ustc.edu.cn, zhangwm@ustc.edu.cn, ynh@ustc.edu.cn

***Abstract*—Order-revealing encryption (ORE) is a kind of encryption designed to support searches on ciphertexts. ORE enables efficient range query on ciphertexts, and it has been used in systems aimed at practical use. However, ORE has weaker security than conventional cryptography. To assess the security of ORE, researchers proposed concepts such as indistinguishability and one-wayness.**

Our work discusses the security of ORE when multiple columns are encrypted with ORE. This paper addresses two issues. First, we show an attacker can use quantile attack to distinguish two plaintext distributions with background information. Simulations show the attack succeed with high probability. Second, we propose a scheme to resist the quantile attack by adding dummy data. The proposed scheme calculates the number and position of dummy data based on the plaintext distribution and expected security level. We conduct experiments on a real dataset to show the performance of proposed scheme.

I. Introduction

Recently, cloud service has been widely used by institutions and individuals. With more and more privacy data being uploaded to the cloud, security has become a serious problem.

To protect privacy, encryption is adopted. However, conventional encryption, which is designed to protect the security of data on insecure channel, will cause problem to the use of data in the scene of cloud service. Thus, searchable encryption is proposed to solve this problem. Order-Preserving Encryption (OPE), and its extension Order-Revealing Encryption (ORE) are encryptions supporting order comparison of ciphertexts.

An ORE scheme is an encryption Enc with a function Cmp such that anyone can tell the order of two plaintexts x_1, x_2 by computing function Cmp on the corresponding ciphertexts $Enc_k(x_1), Enc_k(x_2)$ without knowing the secret key k. An OPE is an ORE if the function Cmp is the order comparison of ciphertext, i.e. $x_1 < x_2 \iff Enc_k(x_1) < Enc_k(x_2)$.

ORE enables efficient range query on ciphertext. To perform a range query $[x_1, x_2]$ on a column encrypted by ORE, the user can encrypt the endpoint x_1, x_2 and send the ciphertext. Then the server finds all rows with values in the interval $[Enc_k(x_1), Enc_k(x_2)]$. ORE does not change the architecture of the server, thus, range query can be performed on ciphertext as fast as the plaintexts. Because of its efficiency, ORE is adopted by systems aimed at practical use, such as CryptDB[1], CipherCloud[2], Google Encrypted BigQuery[3].

Though its efficiency, the security of ORE needs to be considered carefully, because ORE leaks the order of plaintext. To assess the security of ORE, researchers proposed different definitions of indistinguishability and one-wayness. The strongest security notion of deterministic ORE is indistinguishability under ordered-chosen plaintext attack (IND-OCPA) [4], which means an adversary can learn nothing other than the order of plaintext. The ORE based on multilinear map [5] and OPE based on B-tree [6] are proved to be IND-OCPA secure, however, they require complex computation or interactive protocol. To improve the efficiency, Lewi et al. [7] proposed an ORE scheme relying only on symmetric primitives. A deterministic ORE is an ORE scheme mapping the same plaintext to the same ciphertext. To protect the frequency information leaked by deterministic ORE, Wang et al. [8] proposed one-to-many OPE, which is a probabilistic encryption mapping a plaintext to a random value in a interval. Roche et al. [9] and Kerschbaum et al. [10] proposed the concept of frequency-hiding to measure the difference between one-to-many and deterministic ORE.

Above works analyse the theoretical security of ORE. Naveed et al. [11] made an empirical analysis of ORE security. They proposed sort attack and cumulative attack. Sort attack sorts the ciphertext and maps them to plaintext domain according to the order, and it is effective when data is dense, which means that almost every plaintext symbol appears in the plaintext sequence. Cumulative attack combines the cumulative distribution function and frequency information of plaintext to match the plaintexts to ciphertexts.

All these analysis applies to ORE on single-column dataset. To the best of our knowledge, Durak et al. [12] is the first to consider the case that ORE is applied to two or more encrypted columns. They study the information leakage for IND-OCPA secure ORE by 2-D sort attack. They show that information may be leaked even when the data in the column is sparse in its domain, and when all values are unique (and without any training data), and an IND-OCPA secure ORE is used.

In this paper, we consider the security of ORE on multi-column dataset. Though Durak et al. studied the security of ORE on multi-column dataset, their 2-D sort attack is merely applying the sort attack on each column separately and does not exploit inter-column correlation at all. In this paper, we consider the attack model of known-background attack, where the adversary needs to select the plaintext distribution from two or more possible distributions. We present the quantile attack, which exploit the statistics of ciphertexts to distinguish different distributions. Then we suggest a scheme which adds dummy data to resist quantile attack, and we show the performance of proposed scheme by experiments on real data.

978-1-5386-1600-0/17 $31.00 © 2017 IEEE
DOI 10.1109/DSC.2017.49

II. Preliminaries

A. Order-Revealing Encryption

An ORE is a tuple of three algorithms (Key, Enc, Cmp). Algorithm Key takes a security parameter λ as input and outputs a secret key k. Algorithm Enc is encryption algorithm. Algorithm Cmp takes two ciphertexts as input and outputs a bit $b \in \{0, 1\}$. The order-revealing feature requires

$$\begin{aligned} m_1 < m_2 &\iff Cmp_k(Enc_k(m_1), Enc_k(m_2)) = 0, \\ m_1 \geq m_2 &\iff Cmp_k(Enc_k(m_1), Enc_k(m_2)) = 1. \end{aligned} \tag{1}$$

An OPE is an ORE where the algorithm Cmp is the standard order comparison algorithm, i.e.

$$\begin{aligned} m_1 < m_2 &\iff Enc_k(m_1) < Enc_k(m_2), \\ m_1 \geq m_2 &\iff Enc_k(m_1) \geq Enc_k(m_2). \end{aligned} \tag{2}$$

In this paper, we study ORE on multiple column data, where ORE is applied on each column independently with different secret-keys. For a plaintext $m = (w, x)$, we encrypt it to ciphertext $c = Enc_k(m) = (Enc_{k_w}(w), Enc_{k_x}(x))$. A two column data consisting of multiple plaintexts is denoted as $(w), (x)$, and the ciphertext is denoted as $(y), (z)$, where $(y_i, z_i) = Enc_k(w_i, x_i)$. For convenience, we mainly discuss OPE, however, the main conclusion also holds on ORE, because our analysis uses only the order relationship.

B. Attack Model

We present the attack model as known-background attack here.

1) The system generates secret key from security parameter.
2) The adversary chooses a function f.
3) The system generates two plaintext distributions P_1, P_2, and calculates background information $I_i = f(P_i)$.
4) The system randomly picks $b \in \{1, 2\}$, gets a sample $S = (w), (x)$ with size n from P_b as the plaintext, encrypts the plaintext to ciphertext $(y), (z)$, and sends $I_1, I_2, (y), (z)$ to the adversary.
5) The adversary makes a guess b' of b.

The advantage of the adversary is $P(b' = 1|b = 1) - P(b' = 1|b = 2)$. In the known-background attack, plaintext is a sample from P_b, and the length of plaintext is the sample size n. $I_i = f(P_i)$ is the background information. If f is the identity function, then the adversary has full information of the plaintext distribution. In most case, the adversary will not have the entire plaintext but an estimation of the plaintext distribution. For example, the adversary may know only the Pearson correlation coefficient of a uniformly distributed data. In the next section, we will present quantile attack, which only needs a statistics of the distribution.

III. Quantile Attack

In this section, we present the quantile attack, which breaks the encryption with the help of a statistics, which is called as quantile indicator in following paper. We will apply median attack on OPE firstly, then discuss it on more complex situation.

A. Median Attack

In this subsection, we discuss the information leakage caused by the correlation between different ORE encrypted columns.

Every plaintext (w_1, x_1) splits the plaintext space to 4 parts based on the order relationship, and the ciphertext space is also split to 4 parts by corresponding ciphertext (y_1, z_1). Consider another plaintext (w_2, x_2) and corresponding ciphertexts (y_2, z_2), because OPE preserves order, we have $y_2 < y_1$ and $z_2 < z_1$ if $w_2 < w_1$ and $x_2 < x_1$, and vice versa. Denote $T_{W,X}(w_1, x_1)$ as the number of plaintext of which the value on each column is smaller than w_1 and x_1 separately, i.e.

$$T_{W,X}(w_1, x_1) = |\{(w, x)|w < w_1 \wedge x < x_1\}|, \tag{3}$$

and $T_{Y,Z}(y_1, z_1)$ for ciphertext, we have

$$T_{W,X}(w_1, x_1) = T_{Y,Z}(y_1, z_1). \tag{4}$$

The median indicator $r(w, x)$ is the ratio of $T_{W,X}(w_1, x_1)$ to the number of plaintexts when w_1 and x_1 are the medians of w and x respectively, i.e.

$$r(w, x) = \frac{T_{W,X}(w_m, x_m)}{n}, \tag{5}$$

where n is the number of plaintexts, w_m is the median of w, and x_m is the median of x. Ciphertext median indicator $r(y, z)$ is defined similarly. Thus, OPE preserves the median indicator, and we can distinguish two plaintexts if they have different median indicator. The median indicator can be easily extended to quantile indicator, if we replace the median in (5) with quantile.

The details of median attack and its extension quantile attack will be discussed in following subsections, here we discuss the application scope of median attack.

The median attack is effective for ORE, which leaks the order and preserves the median indicator. Besides, the median attack can be applied to encryption scheme which does not preserves the order but leaks some query result with the median. For example, if the adversary knows results of two range queries, $(-\infty, w_m) \times (-\infty, +\infty)$ and $(-\infty, +\infty) \times (-\infty, x_m)$, where w_m and x_m are the median of the two plaintext columns, he can use the intersection of the two query results to calculate the median indicator.

If data has dimension larger than two, we can use the median attack to infer the information of each dimension. Consider a database of trading records consisting of three columns, $C1, C2, C3$, two of which are positions of the user consisting of attitude and latitude, and another column is the trading fee. An adversary has the encrypted database, and some knowledge of the location information of the customers. The first step of the adversary is to judge which two columns are locations. To do so, the adversary can use median attack to judge the similarity between the joint distribution of different rows and the background information. After that, he can combine the background information with each column to infer more information.

B. Median Attack on Normal Distribution

In this subsection, we discuss a simplified and typical case that the plaintext follows a two-dimensional normal distribution. We will calculate the theoretic value of median indicator r and conduct experiments to check it.

Consider a two column data $(w), (x)$ drawing from a two-dimensional normal distribution $P_{W,X}(w, x)$. When the data are encrypted to ciphertexts $(y), (z)$ by OPE, we study the relationship between the median indicator $r(y, z)$ and $P_{W,X}$.

A normal distribution is determined by mean and covariance matrices. Denote the mean of $P_{W,X}$ as

$$\mu = \begin{bmatrix} \mu_1 & \mu_2 \end{bmatrix}, \tag{6}$$

and the covariance matrix as

$$\sigma = \begin{bmatrix} \sigma_w^2 & \rho\sigma_w\sigma_x \\ \rho\sigma_w\sigma_x & \sigma_x^2 \end{bmatrix}, \tag{7}$$

where ρ is the Pearson correlation coefficient.

First, we assume that $\mu_w = 0$ and $\mu_x = 0$. The probability density function of 2-dimensional normal distribution in such situation is

$$p(w, x) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}} e^s, \tag{8}$$

where

$$s = -\frac{1}{2(1-\rho^2)}\left(\frac{1}{\sigma_w^2}w^2 - 2\frac{\rho}{\sigma_w\sigma_x}wx + \frac{1}{\sigma_x^2}x^2\right). \tag{9}$$

The median indicator r_m can be calculated as

$$r = \int_{w=w_m}^{\infty}\int_{x=x_m}^{\infty} p(x_1, x_2)\,\mathrm{d}x_1\,\mathrm{d}x_2, \tag{10}$$

where w_m and x_m are the median of w and x respectively. The accurate calculation of r is difficult, thus here we calculate an approximate for large n. When the sample size is large enough, the median of a normally distributed sample will equal to the sample mean, thus, $w_m = x_m = 0$.

Then we can use substitution to simplify the expression. Let $t_w = w/\sigma_w$ and $t_x = x/\sigma_x$, we have

$$r = \int_{t_1=0}^{\infty}\int_{t_2=0}^{\infty} \frac{1}{2\pi\sqrt{1-\rho}} e^{-\frac{1}{2(1-\rho^2)}(t_1^2 - 2\rho x_1 x_2 + t_2^2)}\,\mathrm{d}t_1\,\mathrm{d}t_2. \tag{11}$$

Let $t_1 = t\cos\theta$, $t_2 = t\sin\theta$, we have

$$r = \int_{t=0}^{\infty}\int_{\theta=0}^{\frac{\pi}{2}} \frac{1}{2\pi\sqrt{1-\rho}} e^{-\frac{1}{2(1-\rho^2)}(t^2 - \rho t^2 sin(2\theta))} t\,\mathrm{d}t\,\mathrm{d}\theta. \tag{12}$$

We first calculate t,

$$\begin{aligned} &\int_{t=0}^{\infty} \frac{1}{2\pi\sqrt{1-\rho}} e^{-\frac{1}{2(1-\rho^2)}(t^2 - \rho t^2 sin(2\theta))} t\,\mathrm{d}t \\ =&\int_{t^2=0}^{\infty} \frac{1}{4\pi\sqrt{1-\rho}} e^{-\frac{1-\rho sin(2\theta)}{2(1-\rho^2)}t^2}\,\mathrm{d}t^2 \\ =&\frac{1}{2\pi}\frac{\sqrt{1-\rho^2}}{1-\rho sin(2\theta)}. \end{aligned} \tag{13}$$

Then r can be calculated, we have

$$r = R_m(\rho), \tag{14}$$

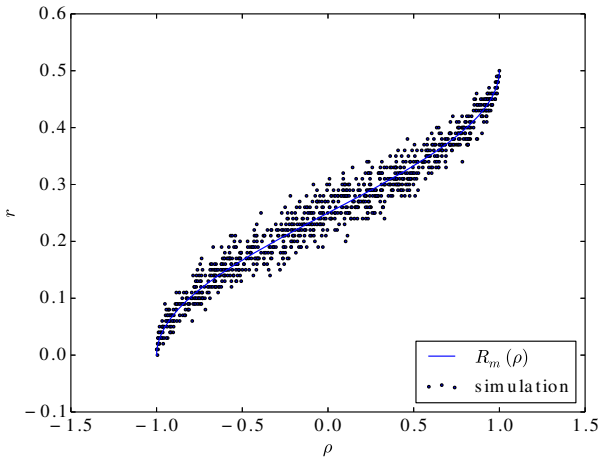

Fig. 1. The median indicator r for different ρ of normal distributions. Sample size $n = 100$.

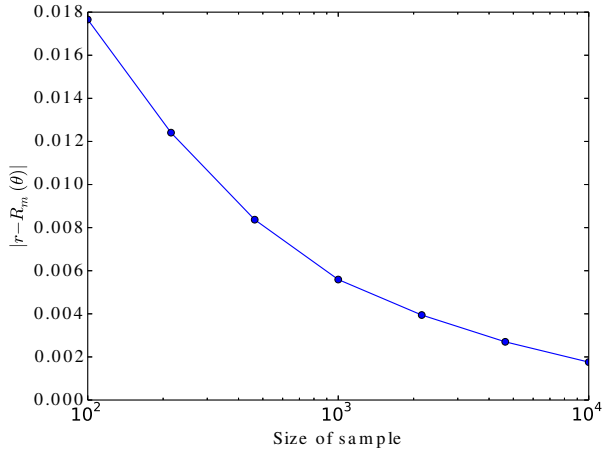

Fig. 2. Mean absolute error $|r - R_m(\theta)|$ of normal distribution.

where

$$R_m(\rho) = \int_{\theta=0}^{\frac{\pi}{2}} \frac{1}{2\pi}\frac{\sqrt{1-\rho^2}}{1-\rho sin(2\theta)}\,\mathrm{d}\theta. \tag{15}$$

This means that r is a function of ρ, and the value of r can be calculated numerically. When the mean is not 0, we have the same conclusion after similar deduction.

When the sample size is not large enough, the actual median indictor r of the sample may deviate from $R_m(\rho)$. We conduct experiments to show the actual relationship between r and ρ. The mean of normal distribution (m_1, m_2) is uniformly picked from $[0, 50] \times [0, 50]$, and the covariance of normal distribution $(\sigma_w, \sigma_x, \rho)$ is uniformly picked from $[0, 30] \times [0, 30] \times [-1, 1]$.

The results are shown in Figs. 1 and 2. Fig. 1 shows the actual r of different ρ when $n = 100$. Fig. 2 shows the mean absolute difference between r and $R_m(\rho)$ of different n. The actual r scatters around $R_m(\rho)$. With the increase of sample size n, the distribution of samples becomes closer to

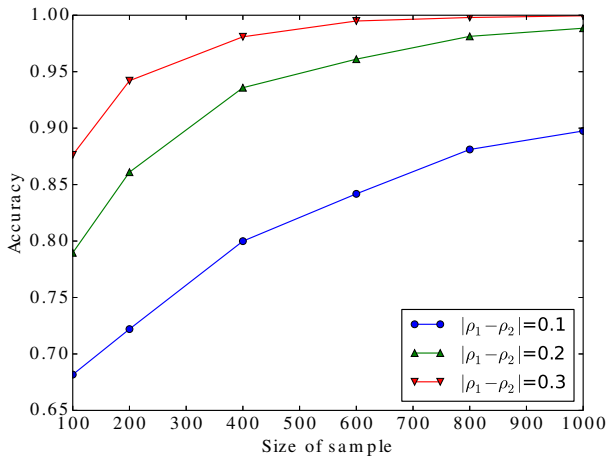

Fig. 3. The accuracy of median attack on normal distribution.

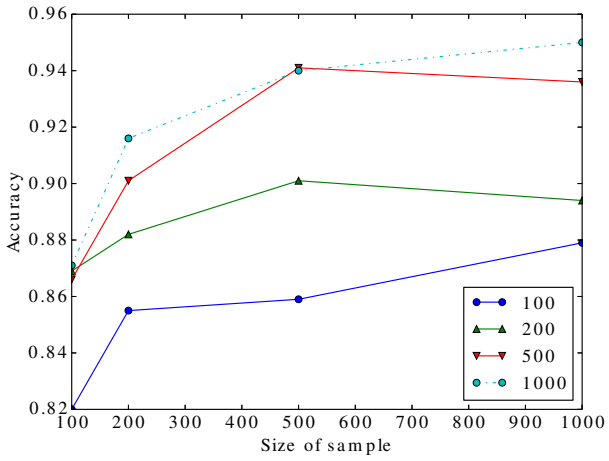

Fig. 4. The accuracy of median attack on mixture of multiple normal distributions. Different curve in the figure correspond to different background sample size.

the normal distribution, and the actual r becomes closer to $R_m(\rho)$.

When we know ρ_1 and ρ_2, the problem of guessing b can be solved by calculating the maximum likelihood probability. Here we use a simple method to distinguish different samples. Given two normal distributions P_1, P_2 with Pearson correlation coefficient ρ_1, ρ_2 respectively and a sample S_b from distribution P_b, the adversary can calculate $r_1 = R_\theta(\rho_1)$, $r_2 = R_\theta(\rho_2)$, and the statistics r of sample S_b. Then the adversary can guess $b = 1$ if $|r_1 - r| < |r_2 - r|$, or $b = 2$ if $|r_1 - r| > |r_2 - r|$. The full median attack on normally distributed data is shown in Algorithm 1.

Algorithm 1 Median Attack on Normal Distribution

```
INPUT: ρ1, ρ2, data S = (w), (x).
OUTPUT: Guess b
r1 ← Rm(ρ1)
r2 ← Rm(ρ2)
Calculate the median indicator r = r(w, x).
if |r − r1| < |r − r2| then
    b ← 1
else
    b ← 2
end if
return b
```

We conduct simulations to show the attack accuracy, and the result is shown in Fig. 3. The precision increase with the sample size n and the difference between ρ_1 and ρ_2. The difference between ρ represents the dissimilarity, thus two distributions with large $|\rho_1 - \rho_2|$ is easy to distinguish.

C. Mixture of Normal Distributions

In this subsection, we discuss distribution which is more complex than normal distribution and hard to calculate theoretically. We assume the plaintext distribution is a mixture of multiple normal distributions, and then we develop a judgement criterion.

When the plaintext follows a mixture of multiple normal distributions, the Pearson correlation coefficient ρ is difficult to calculate. Here we use median indicator r of another sample as background knowledge. More specifically, the background knowledge I_i is the median indicator of S_i, which is sampled from P_i with size n'.

We conduct experiments to show the performance. The plaintext distribution is a mixture of 2 to 10 normal distributions. The mean (μ_w, μ_x) is uniformly distributed on $[-50, 50] \times [-50, 50]$, and the covariance $(\sigma_w, \sigma_x, \rho)$ is uniformly distributed on $[-30, 30] \times [-30, 30] \times [-1, 1]$. We show the accuracy when $n, n' \in \{100, 200, 500, 1000\}$.

Fig. 4 shows the simulation results. Different from the case of single normal, the accuracy decreases when the sample size n increases from 500 to 1000 for $n' = 200$ and $n' = 500$. This is caused by the deviation of background sample from the actual distribution.

D. Quantile Attack

In this subsection, we extend the median attack to quantile attack. If the adversary cannot get the median indicator or median indicator of the two distributions is too close, the adversary can use quantile attack instead.

Here we discuss the attack scenario defined in previous subsection, where the plaintext follows a mixture of normal distributions. With two background samples S_1, S_2, the adversary calculate (i, j) quantile indicator $r(i, j)$ of q-quantiles as $r(i, j) = \frac{1}{n}|\{(w, x)|w < w_q \wedge x < x_q\}|$, where w_q and x_q are i-th and j-th q-quantiles of (w) and (x), respectively. When $q > 2$, the adversary can choose w_q and x_q such that the attack has highest accuracy. As discussed in the scenario of median attack on normal distribution, it is reasonable to assume the larger the difference between r_1 and r_2, the higher

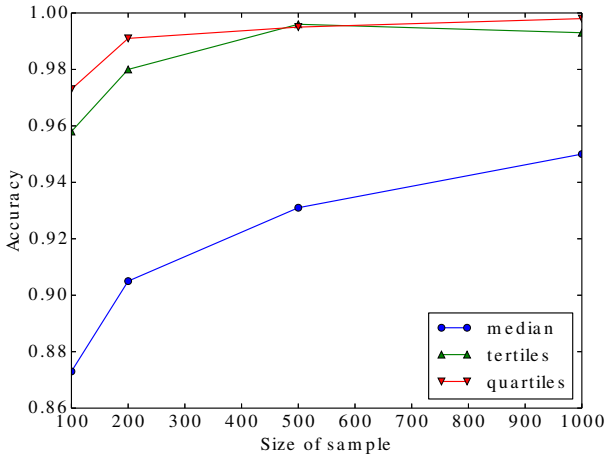

Fig. 5. The accuracy of quantile attack on mixture of multiple normal distributions.

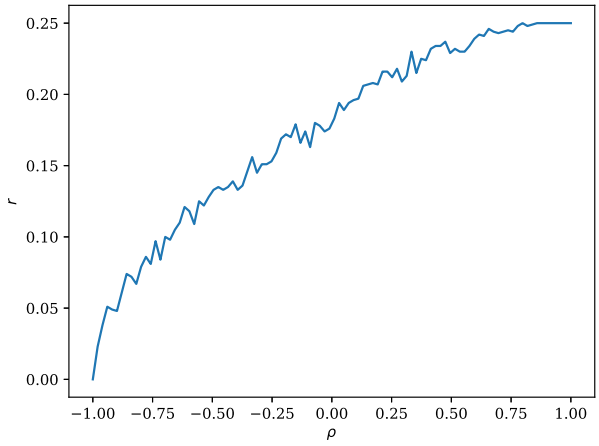

Fig. 6. The relationship between Pearson correlation coefficient ρ and quantile indicator r of normal distributed samples.

the accuracy. Thus, the strategy of the adversary is to choose w_q and x_q with maximum $|r_1 - r_2|$. The full algorithm is shown in Algorithm 2.

Algorithm 2 Quantile Attack

```
INPUT: Quantile indicators r_1, r_2, data S = (w), (x).
OUTPUT: Guess b
Calculate the quantile indicator r of data S.
Find j, k such that |r_1(j,k) − r_2(j,k)| gets maximum.
if |r(j,k) − r_1(j,k)| < |r(j,k) − r_2(j,k)| then
    b ← 1
else
    b ← 2
end if
return b
```

Fig. 5 shows the performance of quantile attack. Obviously, using 3- and 4-quantiles makes a remarkable improvement in the attack accuracy, while the accuracy of 3- and 4-quantile attack keep close. This implies that the improvement happens on distributions with close median indicator

IV. Multi-Dimensional Encryption With Dummy Data

A. Quantile Correlation Coefficient

As shown in preceding section, the quantile attack is effective for ORE. To resist quantile attack, the encryption scheme needs to protect quantile indicator. Because quantile indicator is a statistics of the distribution, an encryption should alter the distribution to ensure security. Here we propose a concept, the quantile correlation coefficient, to measure the security of an encryption under quantile attack.

Quantile indicator can be protected in two means. First, if the ciphertext quantile indicator is independent of the plaintext, the adversary can not infer information of plaintext using quantile attack. Second, the quantile attack has low accuracy when the quantile indicators of two plaintext distributions are close, thus an encryption maps plaintexts to ciphertexts with similar distribution can resist quantile attack effectively.

Thus, we can use the quantile correlation coefficient h_c, which is correlation between the plaintext and ciphertext quantile indicators, to measure the security of an encryption under quantile attack. The quantile correlation coefficient h_c of q-quantile is calculated as follow: First, we generate a group of samples S_i with different quantile indicators and encrypt them to ciphertexts S_i'. Second, we calculate the quantile indicator r_i of plaintext S_i and r_i' of S_i'. The quantile correlation h_c can then be calculated as

$$h_c = \frac{\sum_i (r_i - r_i)(r_i' - r_i')}{\sum_i (r_i - r_i)^2}. \tag{16}$$

When h_c is close to 0, the encryption is secure. Quantile attack is effective for encryptions with h_c close to 1 or -1.

The ideal plaintext distributions depends on actual situation. Here we use normal distributions with the same means and different Pearson correlation coefficients.

Fig. 6 shows the relationship between Pearson correlation coefficient ρ and quantile indicator r of normal distributed samples. The size of plaintext sample is 100, and r is the $(1, 3)$ quantile indicator of 4-quantiles. As shown in the figure, though r is not linear with ρ, the quantile indicator r scatters in the full possible range when ρ is uniformly distributed on $[-1, 1]$. The jump in the curve is caused by the deviation of samples from the background distributions. However, because we use the quantile indicator of plaintext sample rather than ρ in the calculation of h_c, the deviation is not important.

B. Multi-Dimensional Encryption With Dummy Data

Here we use dummy data to improve security under quantile attack. The key problem of the algorithm is arranging dummy data. Because quantile attack is based on quantiles, we can add dummy data into some grids of a q_r by q_r grid divided by q_r-quantiles to resist quantile attack.

h_c	$r(1,1)$	r(1, 3)	$r(2,2)$	r(3, 2)
$k=1$	-0.0085	0.0424	0.4775	0.6197
$k=2$	-0.2043	-0.2120	0.2587	0.1591
$k=3$	-0.2470	-0.2247	0.1286	0.0322
$k=4$	-0.2555	-0.2712	0.0464	0.0057

TABLE I
THE QUANTILE CORRELATION COEFFICIENT OF ORE WITH DUMMY.

h_c	$r(1,1)$	r(1, 3)	$r(2,2)$	r(3, 2)
$k=1$	0.1838	0.1333	0.5086	0.6027
$k=2$	0.1003	0.0723	0.2826	0.2518
$k=3$	-0.0785	0.0445	0.0737	0.1837
$k=4$	-0.0614	-0.0054	0.1063	0.0784

TABLE II
THE QUANTILE CORRELATION COEFFICIENT OF ORE WITH DUMMY.

More specifically, for a dataset S, first we find the q_r-quantiles on each column, and divide the data to a q_r by q_r grid by q_r-quantiles. Denote the center and side length of the (i,j) grid as $(w_{c,i}, x_{c,j})$ and $(w_{l,i}, x_{l,j})$, respectively. Then we calculate the number of data in each grid and add dummy data to k grids with least data. Data added to each grid follows a normal distribution with mean and variance randomly distributed in the grid, and the number of dummy data is the difference between the numbers of current grid and grid with most data. For example, if a grid locates at $[w_1, w_2] \times [x_1, x_2]$, and the dummy data follows normal distribution with mean (μ_w, μ_x) and covariance $(\sigma_w, \sigma_x, \rho)$, then (μ_w, μ_x) follows a uniform distribution on $[w_1, w_2] \times [x_1, x_2]$ and $(\sigma_w, \sigma_x, \rho)$ follows a uniform distribution on $[0, w_2 - w_1] \times [0, x_2 - x_1] \times [-1, 1]$.

The parameter q_r and k are assigned by the user. We calculate the quantile correlation coefficient with $q = 4$ and $q_r = 4$. The result is shown in Table I. Though adding dummy data lowers the quantile correlation coefficient, 0.27 is still a relatively high correlation coefficient. When the adversary knows the parameters of the system, he can add dummy data to background samples with the same parameters and use quantile attack to infer information.

We improve the algorithm in two aspects. The number of dummy data is replaced by a random number, and the grid is recalculated after each group of dummy data is added. The quantile correlation coefficient with $q = 4$ and $q_r = 4$ of the improved algorithm is shown in Table II.

We show the performance of proposed method by simulations.

First, we show the performance when the adversary does not take dummy data into account and use quantile attack on the ciphertext directly.

Fig. 7 shows the performance of proposed method with different parameter k when $q_r = 4$. Compared with Fig. 5, adding dummy data improves the ability to resist quantile attack, and the accuracy of the attack decreases with k. The 4-quantile attack fails when 4 group of dummy data are added.

If the adversary knows that ciphertext contains dummy data and he knows the parameters i, j, q, he can add dummy data to background samples with the same methods before using quantile attack. The performance is shown in Fig. 8. Though

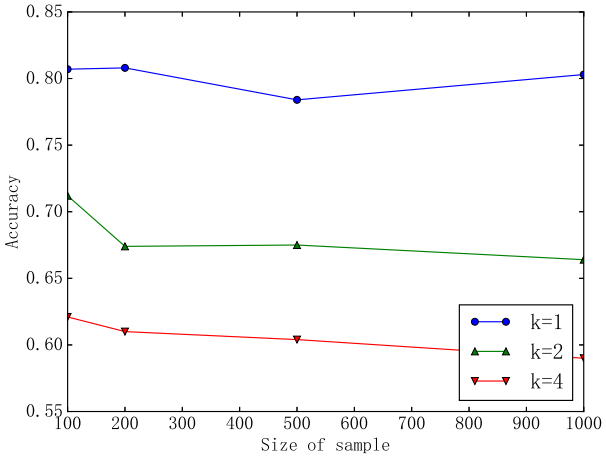

Fig. 7. The accuracy of 4-quantile attack after adding dummy data. $q_r = 4$ and $k = 1, 2, 4$.

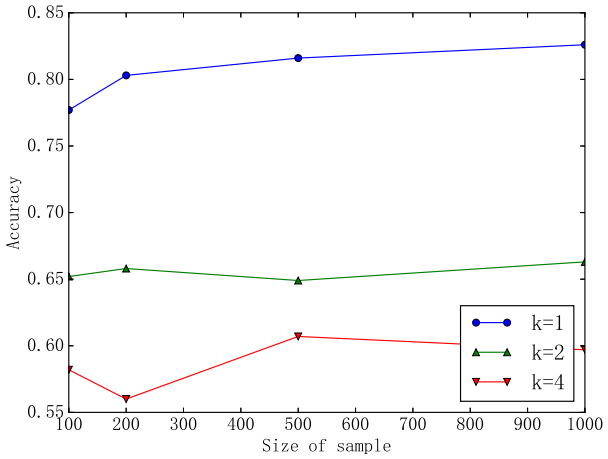

Fig. 8. The accuracy of 4-quantile attack after adding dummy data. $q_r = 4$ and $k = 1, 2, 4$.

the adversary knows the key parameters, adding dummy data still lower the accuracy of attack effectively.

C. Reality dataset

In this subsection we consider dataset from the real world. We use map data of Beijing and Chengdu[1]. The map data of Beijing consists of 412,810 points, and Chengdu 85,658 points. Each point has a longitude and latitude.

We use 4-quantile attack on the dataset. The background dataset is consist of 1000 points uniformly picked from the plaintext. The plaintext is protected with dummy data and encrypted with ORE. The result is shown in Fig. 9. When $k = 0$, no dummy data is added to the plaintext, and the quantile attack has highest accuracy. When $k = 4$, the accuracy is close to 0.5, which implies that an adversary

[1]https://www.openstreetmap.org/

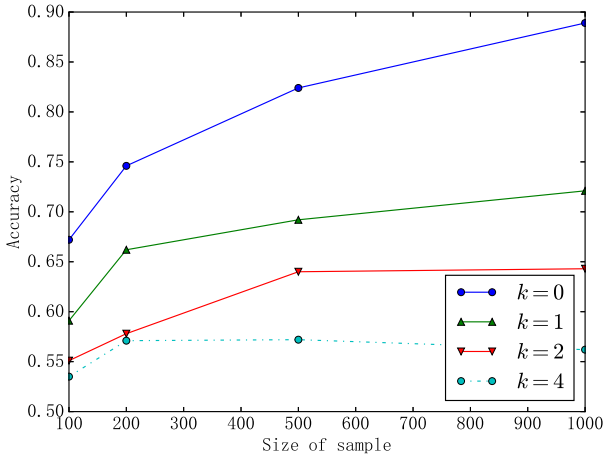

Fig. 9. The accuracy of 4-quantile attack on real dataset with dummy data.

cannot distinguish two plaintexts effectively using quantile attack.

V. Conclusion

In the paper, we analysed the security of ORE on two column data. We proposed quantile attack, which utilize the correlation between two columns to extract information. The experiments on different plaintext distribution and real data shows that the quantile attack is effective for ORE. Then we suggested a scheme which add dummy data to plaintext based on the plaintext distribution, which made a remarkable improvement to the security of ORE against quantile attack.

Acknowledgment

This work was supported in part by the Natural Science Foundation of China under Grant U1636201 and 61572452.

References

[1] R. A. Popa, C. M. S. Redfield, N. Zeldovich, and H. Balakrishnan, "Cryptdb: Protecting confidentiality with encrypted query processing," in *Proceedings of the Twenty-Third ACM Symposium on Operating Systems Principles*, ser. SOSP '11. New York, NY, USA: ACM, 2011, pp. 85–100.

[2] CipherCloud, "Tokenization for cloud data," http://www.ciphercloud.com/tokenization-cloud-data.aspx.

[3] Google, "The encrypted bigquery client," https://github.com/google/encrypted-bigquery-client.

[4] A. Boldyreva, N. Chenette, Y. Lee, and A. ONeill, "Order-preserving symmetric encryption," in *Advances in Cryptology - EUROCRYPT 2009*, ser. Lecture Notes in Computer Science, A. Joux, Ed. Springer Berlin Heidelberg, 2009, vol. 5479, pp. 224–241.

[5] D. Boneh, K. Lewi, M. Raykova, A. Sahai, M. Zhandry, and J. Zimmerman, "Semantically secure order-revealing encryption: Multi-input functional encryption without obfuscation," in *Advances in Cryptology-EUROCRYPT 2015*. Springer, 2015, pp. 563–594.

[6] R. Popa, F. Li, and N. Zeldovich, "An ideal-security protocol for order-preserving encoding," in *Security and Privacy (SP), 2013 IEEE Symposium on*, May 2013, pp. 463–477.

[7] K. Lewi and D. J. Wu, "Order-revealing encryption: New constructions, applications, and lower bounds," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '16. New York, NY, USA: ACM, 2016, pp. 1167–1178. [Online]. Available: http://doi.acm.org/10.1145/2976749.2978376

[8] C. Wang, N. Cao, K. Ren, and W. Lou, "Enabling secure and efficient ranked keyword search over outsourced cloud data," *IEEE Trans. Parallel Distrib. Syst.*, vol. 23, no. 8, pp. 1467–1479, Aug. 2012.

[9] D. S. Roche, D. Apon, S. G. Choi, and A. Yerukhimovich, "Pope: Partial order preserving encoding," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '16. New York, NY, USA: ACM, 2016, pp. 1131–1142. [Online]. Available: http://doi.acm.org/10.1145/2976749.2978345

[10] F. Kerschbaum, "Frequency-hiding order-preserving encryption," in *Proceedings of the 22Nd ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '15. New York, NY, USA: ACM, 2015, pp. 656–667. [Online]. Available: http://doi.acm.org/10.1145/2810103.2813629

[11] M. Naveed, S. Kamara, and C. V. Wright, "Inference attacks on property-preserving encrypted databases," in *Proceedings of the 22Nd ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '15. New York, NY, USA: ACM, 2015, pp. 644–655.

[12] F. B. Durak, T. M. DuBuisson, and D. Cash, "What else is revealed by order-revealing encryption?" in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '16. New York, NY, USA: ACM, 2016, pp. 1155–1166. [Online]. Available: http://doi.acm.org/10.1145/2976749.2978379