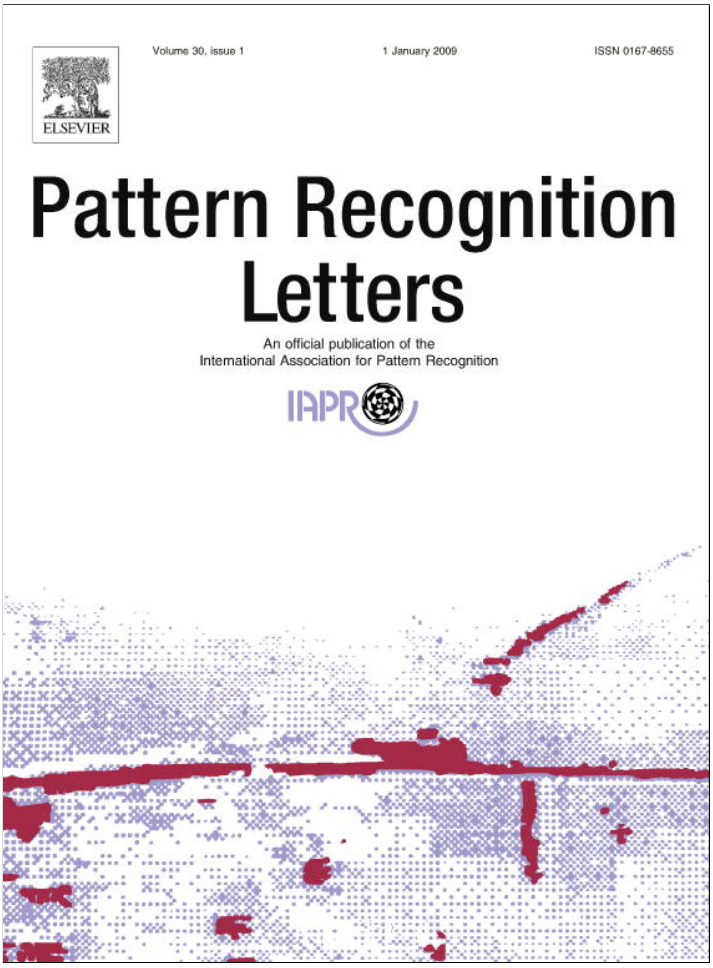

Volume 30, issue 1

1 January 2009

ISSN 0167-8655

ELSEVIER

Pattern Recognition Letters

An official publication of the
International Association for Pattern Recognition

IAPR

Pattern Recognition Letters 30 (2009) 80–86

Contents lists available at ScienceDirect

Pattern Recognition Letters

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

An efficient protocol for private and accurate mining of support counts

Fan Wu [a,*], Jiqiang Liu [b], Sheng Zhong [a]

[a] *Department of Computer Science and Engineering, State University of New York at Buffalo, Buffalo, NY 14260, USA*
[b] *Department of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China*

ARTICLE INFO

Article history:
Received 11 October 2006
Received in revised form 8 September 2008
Available online 20 September 2008

Communicated by W. Pedrycz

Keywords:
Support count
Bilinear mapping
Identity-based encryption

ABSTRACT

In recent years, a large number of data mining tools were developed, which may reveal costumers' privacy if proper protection measure is not taken. On the other hand, customers are becoming increasingly concerned about privacy. They are reluctant to provide personal information unless privacy-preserving techniques are used. In this paper, we propose a privacy-preserving protocol for mining support counts, which maintains high accuracy and strong privacy while achieving very good efficiency. Compared with existing works with similar privacy and accuracy guarantees, our solution is much more efficient. We use identity-based cryptography, which has an additional advantage that no public key certificate is needed. Further, our evaluation results show that the protocol is very efficient and practical.

1. Introduction

Recently, large amounts of consumer data were tracked by automated systems on the Internet. These data contain a lot of personal information. In many cases, data mining may be misused to reveal customers' privacy if proper measure is not taken. This results in that many customers are unwilling to provide personal information unless the privacy of sensitive information is guaranteed. So privacy-preserving data mining (Lindell and Pinkas, 2000; Agrawal and Srikant, 2000) has become an important problem.

Generally, there are two categories of approach for privacy-preserving data mining: perturbation-based approaches and cryptography-based approaches. Existing approaches based on perturbation (Agrawal and Srikant, 2000; Agrawal and Aggarwal, 2001; Du and Zhan, 2003; Evfimievski et al., 2003, 2002; Rizvi and Haritsa, 2002) are efficient, but suffer from the tradeoff between privacy and accuracy. Approaches based on cryptography (Lindell and Pinkas, 2000; Jagannathan et al., 2006; Jagannathan and Wright, 2005; Kantarcioglu and Clifton, 2002; Kardes et al., 2005; Vaidya and Clifton, 2002, 2003, 2004; Wright and Yang, 2004; Yang et al., 2005a,b) can provide strong privacy and high accuracy, but induce heavy computational workload. To make the mining practical, an approach that is fully private, fully accurate and sufficiently efficient is highly needed.

In this paper, we particularly consider the problem of mining *support count* in a fully distributed scenario. In this scenario, each customer maintains a piece of data, and a data miner wishes to count the number of pieces of data that support some pattern. We design an efficient protocol for this problem that preserves strong privacy without losing any accuracy. Our protocol is based on identity-based cryptography. Here, identity-based cryptography is a well-accepted key authentication technique that allows any party to generate a public key from the unique identity (e.g., a user's ID or email address). Consequently, the public keys in our protocol can be distributed before establishing of private keys. This saves the overhead for certificating the public key and makes our protocol widely applicable in many situations.

In summary, our contributions are as follows:

- We propose a protocol for support count that preserves strong privacy without losing any accuracy.
- There is no need for public key certificate in our protocol, due to advantage of identity-based cryptography.
- Our protocol requires only one round of interaction between the miner and each user, and does not need communication channels between the users.
- We did extensive evaluations. The evaluation results show that the protocol is very efficient and practical. Using our protocol, the processing time for the miner to survey 10,000 users is only 420 ms.

The rest of this paper is organized as follows: in Section 2 we present related work. In Section 3, we introduce some preliminaries. In Section 4, we present our identity-based privacy-preserving mining protocol, prove its correctness, and analyze its privacy. In Section 5, we show the results of evaluation on our protocol. Finally, we conclude the paper and point out future work in Section 6.

[*] Corresponding author. Tel.: +1 716 864 4117; fax: +1 716 645 3464.
E-mail addresses: fwu2@cse.buffalo.edu (F. Wu), jl278@cse.buffalo.edu (J. Liu), szhong@cse.buffalo.edu (S. Zhong).

0167-8655/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.patrec.2008.09.004

2. Related work

To solve privacy-preserving data mining problem, there are two main categories of approach: perturbation-based approaches and cryptography-based approaches.

Approaches (Agrawal and Srikant, 2000; Agrawal and Aggarwal, 2001; Du and Zhan, 2003; Evfimievski et al., 2003, 2002; Rizvi and Haritsa, 2002) are based on perturbation of each customer's data. Generally, these approaches are efficient, but rely on a tradeoff between privacy and accuracy: the more accurate result the miner gets, the less privacy can each customer preserves, and vice versa. Since perturbation-based approaches cannot achieve high accuracy and strong privacy simultaneously, their range of usage is limited. It has been shown by Kargupta et al. (2003) that arbitrary data perturbation does not preserve privacy as we expected. Huang et al. (2005) further studied why and how correlations affect privacy and identified other potential factors that can influence privacy.

In contrast, the cryptography-based approaches, proposed by Lindell and Pinkas (2000), Jagannathan et al. (2006), Jagannathan and Wright (2005), Kantarcioglu and Clifton (2002), Kardes et al. (2005), Vaidya and Clifton (2002, 2003, 2004), Wright and Yang (2004), Yang et al. (2005a), and Yang et al. (2005b), provide mining solutions with strong privacy and high accuracy. But these approaches are not as efficient as the perturbation-based approaches. The problem solved here can be referred to as secure multiparty computation (SMC). Some of the cryptography-based solutions rely on expensive protocols for general purpose SMC (Goldreich et al., 1987; Yao, 1986), while others design their own special-purpose protocols. However, as far as we know, all solutions applicable to our support count problem do not have sufficient efficiency to be really practical.

A class of closely related works (Agrawal and Srikant, 2000; Agrawal and Aggarwal, 2001; Evfimievski et al., 2003; Gilburd and Wolff, 2004) studied the foundations for measurement of the effectiveness of privacy preserving data mining algorithms. Confidentiality models in statistical databases are given by Dinur and Nissim (2003) and Dwork and Nissim (2004), respectively.

Unlike previous approaches, Fu et al. (2005) presented a semi-join based approach, without using cryptography, to address privacy-preserving frequent pattern mining in a star schema with two-dimension sites. Their result is interesting but their model and assumptions are significantly different from ours.

Yet another piece of related work is cryptographic randomized response techniques proposed by Ambainis et al. (2004), which guarantees that respondents reply based on the desired probability distributions, under the premise that privacy for the respondents are guaranteed statistically. However, their approaches still has to be tradeoff between privacy and accuracy. In contrast, our protocol guarantees strong privacy and high accuracy at the same time.

3. Technical preliminaries

In this paper, we consider the support count problem in a distributed user–miner scenario and propose an efficient protocol which preserves strong privacy in the semi-honest model. The privacy of our protocol is based on a variant of the decisional Diffie–Hellman assumption(DDH) called decisional bilinear Diffie–Hellman (decisional BDH) assumption. The protocol is based on bilinear maps between groups. The Weil pairing on ecliptic curves is an example of such a mapping.

In this section, we first present the model of our support count problem and define the required privacy. Then we recall the definition of admissible bilinear map and decisional BDH assumption, on which the privacy of our protocol is based. Finally, we briefly review the Boneh–Franklin identity-based encryption scheme (Boneh and Franklin, 2001), on which our protocol is based.

3.1. Problem definition

In our model, we assume that a miner wants to mine a support count on a database or a series of transactions; the database or transactions are distributed among n users $\{U_1, U_2, \dots, U_n\}$ and each user holds a piece of data (e.g., a record of a database or a transaction); each user U_i outputs a binary value b_i (either 1 or 0) indicating whether the data it holds matches the pattern or not.

The objective is letting the miner compute the sum $b = \sum b_i$ without learning anything about users' data. We also require that:

- There is only one round of communication between each user and the miner.
- Users do not communicate with each other.

3.2. Definition of privacy

In the data-mining setting, an adversary can always alter its input to get some information from other party's database. Although this attack can be mitigated by adding noise to data (e.g., perturbation), the outcome is still not as good as we expected (Kargupta et al., 2003). Here we consider the case in semi-honest model, in which the adversary correctly follows the protocol specification, but attempts to learn additional information by analyzing the transcript of messages received during the execution. Semi-honest model is a well-accepted model used in many previous works, such as Lindell and Pinkas(2000), Vaidya and Clifton(2002, 2003, 2004), Wright and Yang (2004), Yang et al.(2005a,b), and Fu et al. (2005). It was shown by Goldreich (2001) that given a multi-party protocol that is secure in the semi-honest model, a protocol that is secure in the malicious model, in which the adversary can arbitrarily deviate from the protocol specification, can be constructed.

We use a simplified form of the standard definition of security in the static semi-honest model due to Goldreich (2001).

Definition 1 (*Computational indistinguishability*). Two ensembles, $X \overset{\text{def}}{=} \{X_w\}_{w \in S}$ and $Y \overset{\text{def}}{=} \{Y_w\}_{w \in S}$ where $S \subseteq \{0,1\}^*$, are computational indistinguishable, denoted $X \overset{c}{\equiv} Y$ if the following holds:

For every polynomial time circuit family, $\{C_n\}_{n \in \mathbb{N}}$, every possible polynomial $p(\cdot)$, every sufficient large n, and every $w \in S \bigcap \{0,1\}^*$,

$$|Pr[C_n(X_w) = 1] - Pr[C_n(Y_w) = 1]| < \frac{1}{p(n)} \qquad (1)$$

Specifically, in our definition of privacy, we consider the possibility that some users conclude with the miner to derive the private information of honest users. We require that no more private information other than the sum can the miner get from the honest users even with help of corrupted users. Observe that, users cannot get any more information than the miner in our model, so we do not have to consider the case that users share information with each other.

Definition 2. Assume that each user U_i has a private key k and a public key *Pub*. A protocol for the above defined mining problem protects each customer's privacy against the miner and t corrupted users in the semi-honest model if, $\forall I \subseteq \{1, \dots, n\}$ such that $|I| = t$, there exists a probabilistic polynomial-time algorithm $\mathscr{M}$ such that

$$\{\mathscr{M}(b, [b_i, k]_{i \in I}, Pub)\} \overset{c}{\equiv} \{\text{view}_{\text{miner},\{U_i\}_{i \in I}}([b_i, k]_{i=1}^{n})\} \qquad (2)$$

Here, $\mathscr{M}$ is also called a simulator that can generate an ensemble computational indistinguishable from the miner and the corrupted users' view using only public keys, miner's knowledge and corrupted users' knowledge.

3.3. Admissible bilinear map

Now we briefly review the bilinear maps (Joux, 2000; Miller, 2004).

Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two groups of order q for some large prime q. The bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ required by our protocol must has the following properties:

(i) *Bilinear:* We say that a map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ is bilinear if $\hat{e}(aP, bQ) = \hat{e}(P,Q)^{ab}$ for all $P, Q \in \mathbb{G}_1$ and all $a, b \in \mathbb{Z}$.
(ii) *Non-degenerate:* It does not map all pairs in $\mathbb{G}_1 \times \mathbb{G}_1$ to elements in $\mathbb{G}_2$. We note that since $\mathbb{G}_1$ and $\mathbb{G}_2$ are groups of prime order, if P is a generator of $\mathbb{G}_1$ then $\hat{e}(P,P)$ is a generator of $\mathbb{G}_2$.
(iii) *Computable:* There exists an efficient algorithm to compute $\hat{e}(P,Q)$ for any $P, Q \in \mathbb{G}_1$.

A bilinear map that satisfies all the three properties above is said to be an admissible bilinear map.

3.4. Decisional bilinear Diffie–Hellman assumption

As we stated before, the privacy of our protocol is based on decisional BDH assumption. In this section, we review the standard decisional BDH assumption.

Definition 3 (*Bilinear Diffie–Hellman problem*). Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two cyclic groups of order q for some large prime q. Let $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ be an admissible bilinear map and P be a generator of $\mathbb{G}_1$. The BDH problem in $\langle \mathbb{G}_1, \mathbb{G}_2, \hat{e} \rangle$ is defined as follows: given $\langle P, aP, bP, cP \rangle$ for some $a, b, c \in \mathbb{Z}_q^*$, compute $W = \hat{e}(P,P)^{abc} \in \mathbb{G}_2$. An algorithm $\mathscr{A}$ has advantage $\epsilon > 0$ in solving BDH problem in $\langle P, aP, bP, cP \rangle$ if

$$Pr[\mathscr{A}(P, aP, bP, cP) = \hat{e}(P,P)^{abc}] \geqslant \epsilon \tag{3}$$

where the probability is over the random choice of a, b, c in $\mathbb{Z}_q^*$, the random choice of $P \in \mathbb{G}_1^*$, and the random bits of $\mathscr{A}$.

Similarly, we have the decisional BDH problem.

Definition 4 (*Decisional bilinear Diffie–Hellman problem*). Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two cyclic groups of order q for some large prime q. Let $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ be an admissible bilinear map and P be a generator of $\mathbb{G}_1$. The decisional BDH problem in $\langle \mathbb{G}_1, \mathbb{G}_2, \hat{e} \rangle$ is defined as follows: given $\langle P, aP, bP, cP \rangle$ for some $a, b, c \in \mathbb{Z}_q^*$, an algorithm $\mathscr{B}$ that outputs $\delta \in \{0,1\}$ has advantage $\epsilon > 0$ in solving decisional BDH problem in $\langle P, aP, bP, cP \rangle$ if

$$\begin{aligned} &|Pr[\mathscr{B}(P, aP, bP, cP, \hat{e}(P,P)^{abc}) = 0] - \\ &Pr[\mathscr{B}(P, aP, bP, cP, T) = 0]| \geqslant \epsilon \end{aligned} \tag{4}$$

where the probability is over the random choice of a, b, c in $\mathbb{Z}_q^*$, the random choice of $P \in \mathbb{G}_1^*$, the random choice of $T \in \mathbb{G}_2^*$, and the random bits of $\mathscr{B}$.

Definition 5 (*Decisional bilinear Diffie–Hellman assumption*). We say that the decisional (t, ϵ)-BDH assumption holds in $\langle \mathbb{G}_1, \mathbb{G}_2, \hat{e} \rangle$ if no t-time algorithm has advantage at least ϵ in solving the decisional BDH problem in $\langle \mathbb{G}_1, \mathbb{G}_2, \hat{e} \rangle$.

Occasionally, we drop the t and ϵ and refer to the decisional BDH assumption in $\langle \mathbb{G}_1, \mathbb{G}_2, \hat{e} \rangle$.*Hardness of decisional BDH:* So far, there is no proof of hardness of decisional BDH problem. However, no algorithm is known to be able to solve this problem either. We refer to a survey (Joux, 2002) for a detailed analysis of BDH problem.

3.5. Boneh–Franklin identity-based encryption scheme

Our technique on support count uses Boneh–Franklin identity-based encryption scheme (Boneh and Franklin, 2001). Their scheme is a tuple of four algorithms $\langle Setup, Extract, Encrypt, Decrypt \rangle$ as follows:

- The *Setup* algorithm creates a tuple of system parameters $\langle q, \mathbb{G}_1, \mathbb{G}_2, \hat{e}, P, P_{Pub}, H_1 \rangle$ and a master-key s, where
  - $\mathbb{G}_1$ and $\mathbb{G}_2$ are two groups of order q for some large prime q.
  - $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ is an admissible bilinear map.
  - P is a random generator in $\mathbb{G}_1$.
  - s is randomly picked from $\mathbb{Z}_q^*$.
  - P_{Pub} is set as $P_{Pub} = sP$.
  - H_1 is a cryptographic hash function defined as $H_1 : \{0,1\}^* \rightarrow \mathbb{G}_1^*$.
- The *Extract* algorithm takes as input the master secret-key and a given $ID \in \{0,1\}^*$, and returns the corresponding private key d_{ID}.
- The *Encrypt* algorithm takes as input the public parameters, an identity $ID \in \{0,1\}^*$, and a message M, then outputs a ciphertext $C = Encrypt(ID, M)$.
- The *Decrypt* algorithm takes as input an identity ID, an associated private key d_{ID}, and a ciphertext C, then outputs a message $M = Decrypt_{d_{ID}}(ID, C)$.

4. Identity-based privacy-preserving support count protocol

In this section, we go to the detail of our support count protocol, prove its correctness and analyze its privacy.

4.1. Protocol

In our protocol, every user U_i has two identities $ID_{i,1} \in \{0,1\}^*$ and $ID_{i,2} \in \{0,1\}^*$, and knows the system parameters $\langle q, \mathbb{G}_1, \mathbb{G}_2, \hat{e}, P, P_{Pub}, H_1 \rangle$ as defined in Boneh–Franklin IBE scheme. Additionally, we introduce another cryptographic hash function $H_2 : \{0,1\}^* \rightarrow \mathbb{Z}_q^*$.

U_i gets her private key $\langle x_i, y_i \rangle$ from the *Extract* algorithm. Here

$$x_i = sP_{ID_i}, \text{ where } P_{ID_i} = H_1(ID_{i,1}),$$
$$y_i = sQ_{ID_i}, \text{ where } Q_{ID_i} = H_1(ID_{i,2}),$$

where s is the master key.

We define

$$X = \sum_{i=1}^{n} r_i P_{ID_i},$$
$$Y = \sum_{i=1}^{n} r_i Q_{ID_i},$$

where $r_i = H_2(ID_{i,1} \| Session\ ID)$. Here, $\|$ is a concatenation operation. We note that the protocol requires that every user knows other users' identities.

Recall that the objective of the miner is to compute the sum $b = \sum b_i$, where b_i is U_i's binary output, without learning anything about each user's input. We define our privacy-preserving support count protocol as follows:

User:
Each user U_i does the following:

(i) computes $m_i = \hat{e}(b_i P, P) \cdot \hat{e}(X, y_i)^{r_i}$,
(ii) computes $n_i = \hat{e}(Y, x_i)^{r_i}$,
(iii) and sends $C_i = \langle m_i, n_i \rangle$ to miner.

Miner:

The miner first computes $Z = \prod_{i=1}^{n} \frac{m_i}{n_i}$. Then it tests b, whose value is from 1 to n. If $\hat{e}(P,P)^b = Z$ then outputs b. Otherwise, it fails to find such a b, and outputs failure.

4.2. Correctness

We prove the correctness of our protocol in this section.

Theorem 6. *The above protocol for support count correctly computes the sum of all users' outputs.*

Proof. Now, we show that the miner can figure out the desired sum b by following the above protocol.

$$\begin{aligned}
Z &= \prod_{i=1}^{n} \frac{m_i}{n_i} \\
&= \prod_{i=1}^{n} \frac{\hat{e}(b_iP,P)\cdot \hat{e}(X,y_i)^{r_i}}{\hat{e}(Y,x_i)^{r_i}} \\
&= \prod_{i=1}^{n} \hat{e}(b_iP,P)\cdot \prod_{i=1}^{n} \frac{\hat{e}(X,r_iy_i)}{\hat{e}(Y,r_ix_i)} \\
&= \frac{\hat{e}(X,\sum_{i=1}^{n} r_iy_i)}{\hat{e}(Y,\sum_{i=1}^{n} r_ix_i)}\cdot \prod_{i=1}^{n} \hat{e}(P,P)^{b_i} \\
&= \frac{\hat{e}(X,sY)}{\hat{e}(Y,sX)}\cdot \hat{e}(P,P)^{\sum_{i=1}^{n} b_i} \\
&= \frac{\hat{e}(X,Y)^s}{\hat{e}(Y,X)^s}\cdot \hat{e}(P,P)^{\sum_{i=1}^{n} b_i} \\
&= \frac{\hat{e}(X,Y)^s}{\hat{e}(X,Y)^s}\cdot \hat{e}(P,P)^{\sum_{i=1}^{n} b_i} \\
&= \hat{e}(P,P)^{\sum_{i=1}^{n} b_i}
\end{aligned}$$

Given $\hat{e}(P,P)^b = Z$, thus $\hat{e}(P,P)^b = \hat{e}(P,P)^{\sum_{i=1}^{n} b_i}$. It is follows that $b = \sum_{i=1}^{n} b_i$.
□

4.3. Privacy analysis

In this section, we prove the privacy guarantee of our protocol using a simplified form of a public key encryption scheme, called BasicPub, which is proposed by Boneh and Franklin (2001). The simplified BasicPub is described as follows:

Setup:
The algorithm works as follows:

(i) Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two cyclic groups of order q, here q is a large prime. Let $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ be an admissible bilinear map. Choose a random generator $P \in \mathbb{G}_1$.

(ii) Pick a random $s \in \mathbb{Z}_q^*$ and set $P_{Pub} = sP$. Pick a random $Q_{ID} \in \mathbb{G}_1^*$.

The message space is $\mathcal{M} = \mathbb{G}_2$. The ciphertext space is $\mathcal{C} = \mathbb{G}_1^* \times \mathbb{G}_2^*$. The public key is $\langle q, \mathbb{G}_1, \mathbb{G}_2, \hat{e}, P, P_{Pub}, Q_{ID}\rangle$. The private key is $d_{ID} = sQ_{ID} \in \mathbb{G}_1^*$.

Encrypt:
To encrypt $M \in \mathcal{M}$, choose a random $r \in \mathbb{Z}_q^*$, and set the ciphertext to be

$$C = \langle rP, M \cdot \hat{e}(Q_{ID}, P_{Pub})^r\rangle$$

Decrypt:
Let $C = \langle U, V\rangle$ be a ciphertext encrypted using the public key $\langle q, \mathbb{G}_1, \mathbb{G}_2, \hat{e}, P, P_{Pub}, Q_{ID}\rangle$. To decrypt C using the private key $d_{ID} \in \mathbb{G}_1^*$ compute:

$$V \cdot \hat{e}(d_{ID}, U)^{-1} = M$$

This completes our simplified BasicPub. Before going to the proof of privacy of our protocol, we first show that the simplified BasicPub is a semantically secure encryption scheme (IND-CPA) if the decisional BDH assumption holds in $\langle \mathbb{G}_1, \mathbb{G}_2, \hat{e}\rangle$.

Lemma 7. *The simplified BasicPub stated above is a semantically secure encryption scheme (IND-CPA) assuming that decisional BDH assumption holds in* $\langle \mathbb{G}_1, \mathbb{G}_2, \hat{e}\rangle$.

Proof. Suppose $\mathcal{A}$ has advantage ϵ in attacking the simplified BasicPub. Then we construct an algorithm $\mathcal{B}$ that can solve the decisional BDH problem in $\langle \mathbb{G}_1, \mathbb{G}_2, \hat{e}\rangle$. Algorithm $\mathcal{B}$ is given as input the decisional BDH parameters $\langle q, \mathbb{G}_1, \mathbb{G}_2, \hat{e}\rangle$ and a random instance $\langle P, aP, bP, cP, T\rangle = \langle P, P_1, P_2, P_3, T\rangle$ that is either sampled from $\mathcal{P}_{\mathrm{BDH}}$ (where $T = \hat{e}(P,P)^{abc}$) or from $\mathcal{R}_{\mathrm{BDH}}$ (where T is uniform and independent in $\mathbb{G}_2$). Algorithm $\mathcal{B}$'s goal is to output 1 if $T = \hat{e}(P,P)^{abc}$ or 0 otherwise, by interacting with $\mathcal{A}$ as the following game:

Setup:
Algorithm $\mathcal{B}$ creates the simplified BasicPub public key $K_{Pub} = \langle q, \mathbb{G}_1, \mathbb{G}_2, \hat{e}, P, P_{Pub}, Q_{ID}\rangle$ by setting $P_{Pub} = P_1$ and $Q_{ID} = P_2$, and gives it to $\mathcal{A}$. Observe that the private key associated to K_{Pub} is $d_{ID} = aQ_{ID} = abP$, which is unknown.

Phase 1:
$\mathcal{A}$ issues q_s private key queries. To respond to these queries, algorithm $\mathcal{B}$ maintains a list of tuples $\langle X_i, Y_i\rangle$. The list is empty in the beginning. To respond to query X_i, algorithm $\mathcal{B}$ does as follows:

(i) If the query X_i already exists on the list in a tuple $\langle X_i, Y_i\rangle$, it responds with Y_i.

(ii) Otherwise, $\mathcal{B}$ picks a random element $Y_i \in \mathbb{G}_2^*$, add the tuple $\langle X_i, Y_i\rangle$ to the list, and responds with Y_i.

Challenge:
When $\mathcal{A}$ decides that Phase 1 is over, it outputs two messages $M_0, M_1 \in \mathbb{G}_2$ on which it wishes to be challenged. Algorithm $\mathcal{B}$ picks a random bit $\delta \in \{0,1\}$, defines C to be the ciphertext $C = \langle P_3, M_\delta \cdot T\rangle$, and gives C as the challenge to $\mathcal{A}$. Observe that, if $T = \hat{e}(P,P)^{abc}$ then C is a valid encryption of M_δ under the public key K_{Pub}. On the other hand, when T is uniform and independent in $\mathbb{G}_2$, then C is independent of δ in $\mathcal{A}$'s view.

Phase 2:
$\mathcal{A}$ continues to issue queries not issued in Phase 1. Algorithm $\mathcal{B}$ responds the same way as before.

Guess:
$\mathcal{A}$ outputs a guess $\delta' \in \{0,1\}$. Then algorithm $\mathcal{B}$ outputs 1 meaning $T = \hat{e}(P,P)^{abc}$ if $\delta = \delta'$, or 0 meaning $T \neq \hat{e}(P,P)^{abc}$ otherwise.

When the input instance is sampled from $\mathcal{P}_{\mathrm{BDH}}$ (where $T = \hat{e}(P,P)^{abc}$), $\mathcal{A}$'s view is identical to the view in a real attack game. Then $\mathcal{A}$ satisfies $|Pr[\delta = \delta'] - 1/2| > \epsilon$. On the other hand, when the sampled instance is from $\mathcal{R}_{\mathrm{BDH}}$ (where T is uniform and independent in $\mathbb{G}_2$), $\mathcal{A}$ satisfies $Pr[\delta = \delta'] = 1/2$. So we have

$$\begin{aligned}
&|Pr[\mathcal{B}(P, aP, bP, cP, \hat{e}(P,P)^{abc}) = 0] - \\
&Pr[\mathcal{B}(P, aP, bP, cP, T) = 0]| \geqslant |(\tfrac{1}{2} \pm \epsilon) - \tfrac{1}{2}| = \epsilon
\end{aligned}$$

where the probability is over the random choice of a, b, c in $\mathbb{Z}_q^*$, the random choice of $P \in \mathbb{G}_1^*$, the random choice of $T \in \mathbb{G}_2^*$, and the random bits of $\mathscr{B}$. This completes the proof. □

Our protocol takes advantage of the homomorphic property of bilinear: $\hat{e}(P_1, Q) \cdot \hat{e}(P_2, Q) = \hat{e}(P_1 + P_2, Q)$.

Next theorem shows that our protocol protects users' privacy as long as simplified BasicPub is semantically secure (IND-CPA), even if up to $n-2$ users corrupt with the miner.

Theorem 8. *If all keys are distributed properly before the protocol starts, our protocol will protect honest users' privacy against the miner and up to* $n-2$ *corrupted users.*

Proof. It is sufficient to consider the case of up to $n-2$ users collude with the miner in our scenario. Without loss of generality, we assume that $I = \{3, 4, \ldots, n\}$. As previously stated, we also construct a simulator $\mathscr{M}$, which will finish in polynomial time, to generate an ensemble using only the public keys, the miner's knowledge and the corrupted users' knowledge. So far, we can state that the miner and the corrupted users jointly learn nothing beyond b.

Note that the simplified BasicPub is semantically secure, and each cipher-text of the simplified BasicPub can be simulated (Goldreich, 2001). Since our protocol is based on the simplified BasicPub, we can combine our simulator $\mathscr{M}$ with a polynomial-time simulator for simplified BasicPub cipher-texts. This completes our simulator.

Next, we show the algorithm that computes the view of the miner and the corrupted users. It takes the following four encryptions as input:

$$(u_{11}, v_{11}) = (r_1 Q_{ID_1}, \hat{e}(b_1 P, P) \cdot \hat{e}(r_1 P_{ID_1}, y_1)^{r_1})$$

$$(u_{12}, v_{12}) = (r_2 P_{ID_2}, \hat{e}(x_2, r_1 Q_{ID_1})^{r_2})$$

$$(u_{21}, v_{21}) = (r_1 P_{ID_1}, \hat{e}(b_2 P, P) \cdot \hat{e}(x_1, r_2 Q_{ID_2})^{r_1})$$

$$(u_{22}, v_{22}) = (r_2 Q_{ID_2}, \hat{e}(r_2 P_{ID_2}, y_2)^{r_2})$$

Then it computes m_1,m_2, n_1, and n_2 as follows:

$$m'_1 = v_{11} v_{12} \hat{e}\left(\sum_{i \in I} r_i x_i, Q_{ID_1}\right)^{r_1}$$

$$m'_2 = v_{21} v_{22} \hat{e}\left(\sum_{i \in I} r_i x_i, Q_{ID_2}\right)^{r_2}$$

$$n'_1 = v_{11} v_{21} \hat{e}\left(\sum_{i \in I} r_i y_i, P_{ID_1}\right)^{r_1} \bigg/ \hat{e}\left(\left(b - \sum_{i \in I} b_i\right) P, P\right)$$

$$n'_2 = v_{12} v_{22} \hat{e}\left(\sum_{i \in I} r_i y_i, P_{ID_2}\right)^{r_2}$$

Next we show that the combined simulator's output is indistinguishable from the adversary's view. To do this, we actually show that, if our simulator $\mathscr{M}$ has the four encryptions m_1, n_1, m_2, n_2 (rather than the simulated values for these encryptions generated by the simulator of BasicPub), then the output of $\mathscr{M}$ is identical to the adversary's view. Consequently, when we combine the two simulators (i.e., when we replace the four encryptions with the output of the simulator for BasicPub), the output of the combined simulator is indistinguishable from the adversary's view (because the output of the simulator for BasicPub is indistinguishable from the four encryptions).

Below are our derivations for $m'_1 = m_1$ and $n'_1 = n_1$. The derivation for $m'_2 = m_2$ and $n'_2 = n_2$ are similar.

$$\begin{aligned}
m'_1 &= v_{11} v_{12} \hat{e}\left(\sum_{i \in I} r_i x_i, Q_{ID_1}\right)^{r_1} \\
&= \hat{e}(b_1 P, P) \cdot \hat{e}(r_1 P_{ID_1}, y_1)^{r_1} \cdot \hat{e}(x_2, r_1 Q_{ID_1})^{r_2} \cdot \hat{e}\left(\sum_{i \in I} r_i x_i, Q_{ID_1}\right)^{r_1} \\
&= \hat{e}(b_1 P, P) \cdot \hat{e}(r_1 P_{ID_1}, sQ_{ID_1}))^{r_1} \cdot \\
&\quad \hat{e}(r_2 s P_{ID_2}, r_1 Q_{ID_1}) \cdot \hat{e}\left(\sum_{i \in I} r_i s P_{ID_i}, Q_{ID_1}\right)^{r_1} \\
&= \hat{e}(b_1 P, P) \cdot \hat{e}(r_1 P_{ID_1}, sQ_{ID_1}))^{r_1} \cdot \\
&\quad \hat{e}(r_2 P_{ID_2}, sQ_{ID_1})^{r_1} \cdot \hat{e}\left(\sum_{i \in I} r_i P_{ID_i}, sQ_{ID_1}\right)^{r_1} \\
&= \hat{e}(b_1 P, P) \cdot \hat{e}(r_1 P_{ID_1}, y_1))^{r_1} \cdot \hat{e}(r_2 P_{ID_2}, y_1)^{r_1} \cdot \hat{e}\left(\sum_{i \in I} r_i P_{ID_i}, y_1\right)^{r_1} \\
&= \hat{e}(b_1 P, P) \cdot \hat{e}\left(\sum_{i=1}^{n} r_i P_{ID_i}, y_1\right)^{r_1} \\
&= \hat{e}(b_1 P, P) \cdot \hat{e}(X, y_1)^{r_1} \\
&= m_1.
\end{aligned}$$

$$\begin{aligned}
n'_1 &= v_{11} v_{21} \hat{e}\left(\sum_{i \in I} r_i y_i, P_{ID_1}\right)^{r_1} \bigg/ \hat{e}\left(\left(b - \sum_{i \in I} b_i\right) P, P\right) \\
&= \hat{e}(b_1 P, P) \cdot \hat{e}(r_1 P_{ID_1}, y_1)^{r_1} \cdot \hat{e}(b_2 P, P) \\
&\quad \cdot \hat{e}(\sum_{i \in I} r_i y_i, P_{ID_1})^{r_1} \bigg/ \hat{e}\left(\left(b - \sum_{i \in I} b_i\right) P, P\right) \\
&= \hat{e}(x_1, r_2 Q_{ID_2})^{r_1} \cdot \hat{e}(r_1 P_{ID_1}, sQ_{ID_1})^{r_1} \\
&\quad \cdot \hat{e}(r_2 Q_{ID_2}, x_1)^{r_1} \cdot \hat{e}\left(\sum_{i \in I} r_i s Q_{ID_i}, P_{ID_1}\right)^{r_1} \\
&= \hat{e}(r_1 Q_{ID_1}, sP_{ID_1})^{r_1} \cdot \hat{e}(r_2 Q_{ID_2}, x_1)^{r_1} \cdot \hat{e}\left(\sum_{i \in I} r_i Q_{ID_i}, sP_{ID_1}\right)^{r_1} \\
&= \hat{e}(r_1 Q_{ID_1}, x_1)^{r_1} \cdot \hat{e}(r_2 Q_{ID_2}, x_1)^{r_1} \cdot \hat{e}\left(\sum_{i \in I} r_i Q_{ID_i}, x_1\right)^{r_1} \\
&= \hat{e}\left(\sum_{i=1}^{n} r_i Q_{ID_i}, x_1\right)^{r_1} \\
&= \hat{e}(Y, x_1)^{r_1} \\
&= n_1. \quad \square
\end{aligned}$$

5. Evaluations

We implemented our privacy-preserving support count protocol using the PBC libraries (pairing-based cryptography library), which is a C library based on the GMP library that contains routines that aid the implementation of pairing-based crypto-systems, for the cryptographic operations. We ran a series of evaluations on a laptop with a 1.4 GHz processor and 768MB memory under RedHat Linux 9. In our evaluations, the length of each cryptographic key is set to 512 bits. We measure the computational time of the privacy-preserving support count protocol for different numbers of customers, from 2000 to 10,000.

We use the curve $y^2 = x^3 + x$ over the field F_q for a large prime q. It turns out $\#E(F_q) = q + 1$ and $\#E(F_{q^2}) = (q+1)^2$. We choose $q \equiv -1 \bmod 12$ so we can implement F_{q^2} as $F_q[i]$ (where $i = \text{sqrt}(-1)$).

In the setup phase, it takes only 27.0 ms to generate the private key for each user. Before each time of mining, the protocol

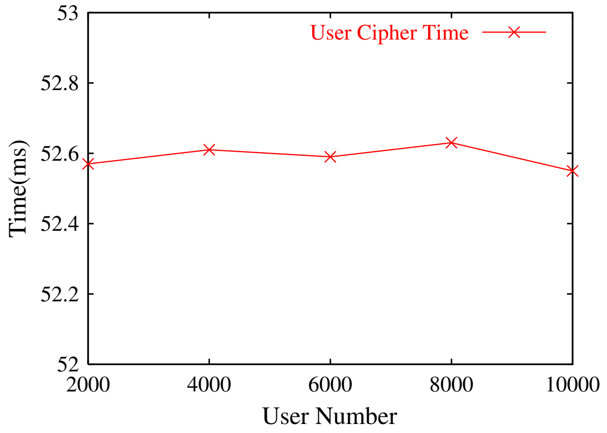

Fig. 1. Average time used to compute cipher-text, as a function of user number.

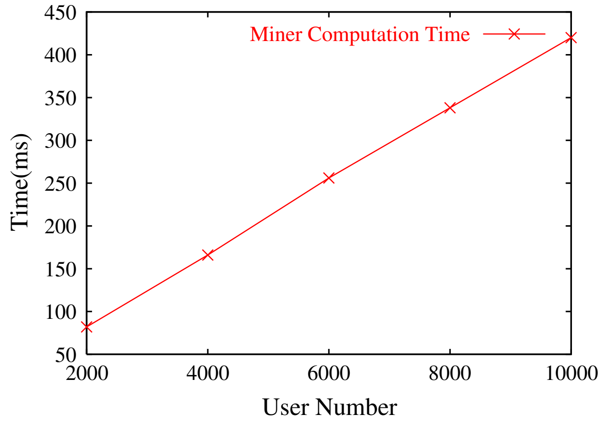

Fig. 2. Time used to decrypt cipher-text and to find d, as a function of user number.

parameters, X and Y, should be computed, which takes 54.9 s for 2000 users. This may look expensive. However, this computation can be performed before the protocol starts, since these two parameters are computed only based on user IDs and session IDs. So computing X and Y does not really slow down our mining protocol.

Fig. 1 demonstrates that using our privacy-preserving support count protocol, the length of average time for computing cipher-text in different cases is almost same, which is about 52.6 ms. This amount of time is mainly for two bilinear mappings. In our scenario, since b_i is 0 or 1, $\hat{e}(b_iP, P)$ has only two possible values corresponding to 0 and 1, respectively. So we can reduce a bilinear mapping, which is time consuming, by introducing a case operation. This saves one third of the computational time.

As shown in Fig. 2, the miner's computational time is roughly linear with the number of users. Although it is a little longer than the cipher time of a single user, it is still very efficient. The computational time is 420 ms, even in case of 10,000 users.

Our evaluations show that the support count protocol is capable for large scale privacy-preserving support count.

6. Conclusion and future work

In this paper, we propose a privacy-preserving support count protocol to solve the problem of surveying a large number of customers without revealing users' private input, in a fully distributed scenario. Our protocol preserves strong privacy without losing any accuracy. Further, we did extensive evaluations, and the results show that our protocol is very efficient.

This paper deals with the case of horizontally partitioned data over each user. One of the open problem is whether we can design such a efficient support count protocol suitable with the case of vertically partitioned data among different parties. Yet, another problem is whether we can combine our protocol with perturbation techniques to further improve efficiency without losing any accuracy or only losing acceptable accuracy. However, we leave these as issues for future work.

References

Agrawal, D., Aggarwal, C., 2001. On the design and quantification of privacy preserving data mining algorithms. In: Proceedings of the 20th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, ACM Press, pp. 247–255.

Agrawal, R., Srikant, R., 2000. Privacy-preserving data mining. In: Proceedings of the Sixth ACM SIGMOD International Conference on Management of Data, ACM Press, pp. 439–450.

Ambainis, A., Jakobsson, M., Lipmaa, H., 2004. Cryptographic randomized response techniques. In: Proceedings of the PKC 2004 International Workshop on Practice and Theory in Public Key Cryptography, Springer-Verlag, pp. 425–438.

Boneh, D., Franklin, M., 2001. Identity-based encryption from the weil pairing. In: Advances in Cryptology – CRYPTO, vol. 2139. LNCS, pp. 213–229.

Dinur, I., Nissim, K., 2003. Revealing information while preserving privacy. In: Proceedings of 22nd ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, pp. 202–210.

Du, W., Zhan, Z., 2003. Using randomized response techniques for privacy-preserving data mining. In: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data mining, ACM Press, pp. 505–510.

Dwork, C., Nissim, K., 2004. Privacy-preserving datamining on vertically partitioned databases. In: Advances in Cryptology – CRYPTO, vol. 3152. LNCS, pp. 528–544.

Evfimievski, A., Gehrke, J., Srikant, R., 2003. Limiting privacy breaches in privacy preserving data mining. In: Proceedings of the 22nd ACM SIGMOD-SIGACTSIGART Symposium on Principles of Database Systems, ACM Press, pp. 211–222.

Evfimievski, A., Srikant, R., Agrawal, R., Gehrke, J., 2002. Privacy preserving mining of association rules. In: Proceedings of Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM Press, pp. 217–228.

Fu, A., Wong, R., Wang, K., 2005. Privacy-preserving frequent pattern mining across private databases. In: Proceedings of the ICDM Workshop on Privacy and Security Aspects of Data Mining.

Gilburd, B., Schuster, A., Wolff, R., 2004. k-TTP: A new privacy model for large-scale distributed environments. In: Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, ACM Press, pp. 563–568.

Goldreich, O., 2001. Foundations of Cryptography. Basic Tools, vol. 1. Cambridge University Press.

Goldreich, O., Micali, S., Wigderson, A., 1987. How to play any mental game. In: Proceedings of the 19th Annual ACM Conference on Theory of Computing, ACM Press, pp. 218–229.

Huang, Z., Du, W., Chen, B., 2005. Deriving private information from randomized data. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, ACM Press, pp. 37–48.

Jagannathan, G., Pillaipakkamnatt, K., Wright, R.-N., 2006. A new privacy-preserving distributed k-clustering algorithm. In: Proceedings of the SDM SIAM International Conference on Data Mining.

Jagannathan, G., Wright, R.N., 2005. Privacy-preserving distributed k-means clustering over arbitrarily partitioned data. In: Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, ACM Press, pp. 593–599.

Joux, A., 2000. A one round protocol for tripartite Diffie–Hellman. In: Proceedings of the Fourth ANTS-IV International Symposium on Algorithmic Number Theory, pp. 385–394.

Joux, A., 2002. The weil and tate pairings as building blocks for public key cryptosystems. In: Proceedings of the Fifth ANTS-V International Symposium on Algorithmic Number Theory, Springer-Verlag, pp. 20–32.

Kantarcioglu, M., Clifton, C., 2002. Privacy-preserving distributed mining of association rules on horizontally partitioned data. In: Proceedings of the DMKD ACM SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery, pp. 24–31.

Kardes, O., Ryger, R.S., Wright, R.-N., Feigenbaum, J., 2005. Implementing privacy-preserving bayesian-net discovery for vertically partitioned data. In: Proceedings of the ICDM Workshop on Privacy and Security Aspects of Data Mining.

Kargupta, H., Datta, S., Wang, Q., Sivakumar, K., 2003. On the privacy preserving properties of random data perturbation techniques. In: Proceedings of the Third ICDM IEEE International Conference on Data Mining, pp. 99–106.

Lindell, Y., Pinkas, B., 2000. Privacy preserving data mining. In: Advances in Cryptology – Crypto2000, vol. 1880. LNCS, Springer-Verlag, pp. 36–53.

Miller, V.S., 2004. The weil pairing, and its efficient calculation. J. Cryptol. 17 (4), 235–261.

Rizvi, S., Haritsa, J., 2002. Maintaining data privacy in association rule mining. In: Proceedings of the 28th VLDB Conference on Very Large Data Base, pp. 682–693.

Vaidya, J., Clifton, C., 2002. Privacy preserving association rule mining in vertically partitioned data. In: Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 639–644.

Vaidya, J., Clifton, C., 2003. Privacy-preserving k-means clustering over vertically partitioned data. In: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 206–215.

Vaidya, J., Clifton, C., 2004. Privacy preserving naive bayes classifier for vertically partitioned data. In: Proceedings of the Fourth SIAM International Conference on Data Mining, pp. 522–526.

Wright, R., Yang, Z., 2004. Privacy-preserving bayesian network structure computation on distributed heterogeneous data. In: Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM Press, pp. 713–718.

Yang, Z., Zhong, S., Wright, R.-N., 2005a. Anonymity-preserving data collection. In: Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, ACM Press, pp. 334–343.

Yang, Z., Zhong, S., Wright, R.-N., 2005b. Privacy-preserving classification of customer data without loss of accuracy. In: Proceedings of the Fifth SIAM International Conference on Data Mining.

Yao, A., 1986. How to generate and exchange secrets. In: Proceedings of the 27th IEEE Symposium on Foundations of Computer Science, pp. 162–167.