2017 IEEE 33rd International Conference on Data Engineering

Trading Data in Good Faith: Integrating Truthfulness and Privacy Preservation in Data Markets

Chaoyue Niu, Zhenzhe Zheng, Fan Wu, Xiaofeng Gao, and Guihai Chen
Shanghai Key Laboratory of Scalable Computing and Systems
Department of Computer Science and Engineering
Shanghai Jiao Tong University, China
Email: {rvincency, zhengzhenzhe220}@gmail.com; {fwu, gao-xf, gchen}@cs.sjtu.edu.cn

***Abstract*—As a significant business paradigm, many online information platforms have emerged to satisfy society's needs for person-specific data, where a service provider collects raw data from data contributors, and then offers value-added data services to data consumers. However, in the data trading layer, the data consumers face a pressing problem, *i.e.*, how to verify whether the service provider has truthfully collected and processed data? Furthermore, the data contributors are usually unwilling to reveal their sensitive personal data and real identities to the data consumers. In this paper, we propose TPDM, which efficiently integrates Truthfulness and Privacy preservation in Data Markets. TPDM is structured internally in an Encrypt-then-Sign fashion, using somewhat homomorphic encryption and identity-based signature. It simultaneously facilitates batch verification, data processing, and outcome verification, while maintaining identity preservation and data confidentiality. We also instantiate TPDM with a profile-matching service, and extensively evaluate its performance on Yahoo! Music ratings dataset. Our evaluation results show that TPDM achieves several desirable properties, while incurring low computation and communication overheads when supporting a large-scale data market.**

I. Introduction

In the era of big data, society has developed an insatiable appetite for sharing personal data. Realizing the potential of personal data's economic value in decision making and user experience enhancement, several open information platforms have emerged to enable person-specific data to be exchanged on the Internet [6], [11], [13], [14]. For example, Gnip, which is Twitter's enterprise API platform, collects social media data from Twitter users, mines deep insights into customized audiences, and provides data analysis solutions to more than 95% of the Fortune 500 [11].

However, there exists a critical security problem in these market-based platforms, *i.e.*, it is difficult to guarantee the truthfulness in terms of data collection and data processing, especially when privacies of the data contributors are needed to be preserved. Let's examine the role of a pollster in the presidential election as follows. As a reliable source of intelligence, the Gallup Poll [10] uses impeccable data to assist presidential candidates in identifying and monitoring economic and behavioral indicators. In this scenario, simultaneously ensuring truthfulness and preserving privacy require the Gallup Poll to convince the presidential candidates that those indicators are derived from live interviews without leaking any interviewer's real identity (*e.g.*, social security number) or the content of her interview. If raw data sets for drawing these indicators are mixed with even a small number of bogus or synthetic samples, it will exert bad influence on the final election result.

Ensuring truthfulness and protecting the privacies of data contributors are both important to the long term healthy development of data markets. On one hand, the ultimate goal of the service provider in a data market is to maximize her profit. Therefore, in order to minimize the expenditure for data acquisition, an opportunistic way for the service provider is to mingle some bogus or synthetic data into the raw data sets (called "partial data collection attack"). Yet, to reduce operation cost, a strategic service provider may return a fake result without processing the data from designated sources, or provide data services based on a subset of the whole raw data set (called "no/partial data processing attack"). However, if such speculative and illegal behaviors cannot be identified and prohibited, it will cause heavy losses to the data consumers, and thus destabilize the data market. On the other hand, while unleashing the power of personal data, it is the bottom line of every business to respect the privacies of data contributors. The debacle, which follows AOL's public release of "anonymized" search records of its customers, highlights the potential risk to individuals in sharing personal data with private companies [2]. Therefore, the content of raw data should not be disclosed to data consumers to guarantee data confidentiality, even if the real identities of the data contributors are hidden.

To integrate truthfulness and privacy preservation in a practical data market, there are three major challenges. The first and the thorniest design challenge is that verifying the truthfulness of data collection and preserving the privacy seem to be contradictory objectives. Specifically, the non-repudiation property in classical digital signature schemes implies the truthfulness of data collection in our model. However, the verification requires the knowledge of raw data, and can easily leak a data contributor's real identity [5].

Yet, another challenge comes from data processing, which makes verifying the truthfulness of data collection even harder. In data markets, some of the service providers process data for value-added data services rather than directly offering raw data [1]. This differs from most conventional data sharing scenarios, *e.g.*, data publishing. However, the data service may no longer be semantically consistent with the raw data [9], which makes the data consumer hard to believe the truthfulness of data collection. Moreover, although data provenance [12] helps to determine the derivation history of a data processing result, it cannot guarantee the truthfulness of data collection.

The last but not least design challenge is the efficiency requirement of data markets, especially for data acquisition. For example, 25 billion data collection activities take place on Gnip every day [11]. Meanwhile, the service provider needs to verify data authentication and data integrity. However, the sequential verification method in classical signature schemes may fail to satisfy the stringent time requirement of data markets. Furthermore, the maintenance of digital certificates under the traditional Public Key Infrastructure (PKI) also incurs significant communication overhead.

In this paper, by jointly considering above three challenges, we propose TPDM. TPDM first exploits somewhat homomorphic encryption to construct a ciphertext space[1], which enables the service provider to launch data services and the data

This work was supported in part by the State Key Development Program for Basic Research of China (973 project 2014CB340303), in part by China NSF grant 61672348, 61672353, 61422208, and 61472252, in part by Shanghai Science and Technology fund 15220721300, and in part by the Scientific Research Foundation for the Returned Overseas Chinese Scholars. The work of Z. Zheng was supported by a Google Ph.D. Fellowship and a Microsoft Asia Ph.D. Fellowship. The opinions, findings, conclusions, and recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the funding agencies or the government.

F. Wu is the corresponding author.

[1]The ciphertext space construction by exploiting conventional symmetric/asymmetric encryption is vulnerable to no/partial data processing attack.

2375-026X/17 $31.00 © 2017 IEEE
DOI 10.1109/ICDE.2017.80

consumers to perform outcome verification, while maintaining data confidentiality. In contrast to classical digital signature schemes, which are operated over plaintexts, our new identity-based signature scheme is conducted in the ciphertext space. Furthermore, each data contributor's signature is derived from her real identity, and is unforgeable against the service provider or other external attackers. This appealing property can convince data consumers that the service provider has truthfully collected data. To reduce the latency caused by verifying a bulk of signatures, we propose a two-layer batch verification scheme. At last, TPDM realizes identity preservation and recoverability by carefully adopting ElGamal encryption.

II. Design of TPDM

In this section, we propose TPDM.

A. System Model

We consider a general system model for data markets [1]. The model has a data acquisition layer and a data trading layer. There are four major kinds of entities: data contributors, a service provider, data consumers, and a registration center.

In the data acquisition layer, the service provider procures massive raw data from the data contributors, such as social network users, mobile smart devices, smart meters, and so on. For the sake of security, each registered data contributor is equipped with a tamper-proof device. The tamper-proof device can be implemented in the form of either specific hardware [16] or software [5]. It prevents any adversary from extracting the information stored in the device, including cryptographic keys, codes, and data.

We consider that the service provider tends to offer value-added data services to data consumers rather than directly revealing sensitive raw data, *e.g.*, social network analyses, personalized recommendations, and aggregate statistics.

The registration center maintains an online database of registrations, and assigns each registered data contributor an identity and a password to activate her tamper-proof device. In addition, the registration center maintains an official website, called certificated bulletin board [4], on which the legitimate system participants can publish information.

B. Fine-grained Profile Matching

In this work, from a practical standpoint, we elaborate on a classic data service in social networking, *i.e.*, fine-grained profile matching. Unlike the interactive scenario in [18], our centralized data market breaks the limit of neighborhood finding. In particular, a data consumer's friending strategy can be derived from a large scale of data contributions.

During the initial phase of profile matching, the service provider, *e.g.*, Twitter or OkCupid, defines a public attribute set consisting of β attributes $\mathbb{A} = \{A_1, A_2, \cdots, A_\beta\}$, where A_i corresponds to a personal interest such as movie, sports, cooking, and so on. Then, to create a fine-grained personal profile, a data contributor o_i, *e.g.*, a Twitter or OkCupid user, selects an integer $u_{ij} \in [0, \theta]$ to indicate her level of interest in $A_j \in \mathbb{A}$, and thus forms her profile vector $\vec{U}_i = (u_{i1}, u_{i2}, \cdots, u_{i\beta})$. Subsequently, o_i submits her profile vector $\vec{U}_i$ to the service provider for matching process.

To facilitate profile matching, the data consumer is also required to provide her profile vector $\vec{V} = (v_1, v_2, \cdots, v_\beta)$ and an acceptable similarity threshold δ, where δ is a non-negative integer. Without loss of generality, we assume that the service provider employs *Euclidean distance* $f(\cdot)$ to measure the similarity between the data contributor o_i and the data consumer, where $f(\vec{U}_i, \vec{V}) = \sqrt{\sum_{j=1}^{\beta} (u_{ij} - v_j)^2}$. To simplify construction, we covert the matching metric $f(\vec{U}_i, \vec{V}) < \delta$ to its squared form $f(\vec{U}_i, \vec{V})^2 = \sum_{j=1}^{\beta} (u_{ij} - v_j)^2 < \delta^2$.

Given the profile-matching scenario considered here, we utilize a somewhat homomorphic encryption scheme based on bilinear maps, called Boneh-Goh-Nissim (BGN) cryptosystem [4]. This is because we only require the oblivious evaluation of quadratic polynomials, *i.e.*, $\sum_{j=1}^{\beta} (u_{ij} - v_j)^2$. In particular, the BGN scheme supports any number of homomorphic additions after a single homomorphic multiplication.

C. Design Details

We now introduce TPDM in details. TPDM consists of 4 phases: initialization, signing key generation, data submission, and data processing and verifications.

Phase I: Initialization

We assume that the registration center sets up the system parameters at the beginning of data trading as follows:

• The registration center chooses three multiplicative cyclic groups $\mathbb{G}_1$, $\mathbb{G}_2$, and $\mathbb{G}_T$ with the same prime order q. Besides, g_1 is a generator of $\mathbb{G}_1$, and g_2 is a generator of $\mathbb{G}_2$. Moreover, these three cyclic groups compose an admissible pairing [3] $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$.

• The registration center randomly picks $s_1, s_2 \in \mathbb{Z}_q^*$ as her two master keys, and then computes

$$P_0 = {g_1}^{s_1}, P_1 = {g_2}^{s_1}, \text{ and } P_2 = {g_2}^{s_2}$$

as public keys. The two master keys s_1, s_2 are preloaded into each registered data contributor's tamper-proof device.

• The registration center sets up parameters for the BGN cryptosystem: a private key $\mathcal{SK}$, a public key $\mathcal{PK}$, an encryption scheme $E(\cdot)$, and a decryption scheme $D(\cdot)$.

• To activate the tamper-proof device, each registered data contributor o_i is assigned with a "real" identity $RID_i \in \mathbb{G}_1$ and a password PW_i. Here, RID_i uniquely identifies o_i, while PW_i is required in the access control stage.

• The system parameters

$$\{\hat{e}, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T,\ q,\ g_1,\ g_2,\ P_0,\ P_1,\ P_2,\ \mathcal{PK},\ E(\cdot)\}$$

are published on the certificated bulletin board.

Phase II: Signing Key Generation

To achieve anonymous authentication in the data market, the tamper-proof device is utilized to generate a pair of pseudo identity PID_i and secret key SK_i for o_i:

$$PID_i = \langle PID_i^1, PID_i^2 \rangle = \langle {g_1}^r, RID_i \odot {P_0}^r \rangle, \quad (1)$$

$$SK_i = \langle SK_i^1, SK_i^2 \rangle = \langle {PID_i^1}^{s_1}, H(PID_i^2)^{s_2} \rangle, \quad (2)$$

where r is a per-session random nonce, $\odot$ represents the Exclusive-OR (XOR) operation, and $H(\cdot)$ is a MapToPoint hash function [3], *i.e.*, $H(\cdot) : \{0,1\}^* \to \mathbb{G}_1$. We note that PID_i is actually an ElGamal encryption [8] of RID_i over the elliptic curves, while SK_i is generated accordingly by exploiting identity-based encryption (IBE) [3].

Phase III: Data Submission

Ahead of submission, each data contributor o_i encrypts her profile $\vec{U}_i$ with the BGN scheme, and gets the ciphertext vector

$$\vec{D}_i = \big(E(u_{ij}), E({u_{ij}}^2)\big)\,|_{j \in [1,\beta]}.$$

After encryption, each data contributor o_i computes the signature σ_i on the ciphertext vector $\vec{D}_i$ using her secret key:

$$\sigma_i = SK_i^1 \cdot {SK_i^2}^{h(D_i)}, \quad (3)$$

where "$\cdot$" denotes the group operation in $\mathbb{G}_1$, $h(\cdot)$ is a one-way hash function such as SHA-1 [7], and D_i is derived by concatenating all the elements of $\vec{D}_i$ together.

Eventually, the data contributor o_i submits her tuple $\langle \textit{PID}_i, \vec{D}_i, \sigma_i \rangle$ to the service provider. Once receiving the tuple, the service provider is required to post the pseudo identity $\textit{PID}_i$ on the certificated bulletin board for non-repudiation.

Phase IV: Data Processing and Verifications

Before introducing this phase, we assume that the service provider receives a bundle of data tuples from n distinct data contributors, denoted as $\{\langle \textit{PID}_i, \vec{D}_i, \sigma_i \rangle | i \in [1, n]\}$.

► **First-layer Batch Verification**

To verify data authentication and data integrity, the service provider needs to check whether

$$\begin{aligned} & \hat{e}\left(\prod_{i=1}^{n} \sigma_i, g_2\right) \\ = \quad & \hat{e}\left(\prod_{i=1}^{n} \textit{PID}_i^1, P_1\right) \hat{e}\left(\prod_{i=1}^{n} H(\textit{PID}_i^2)^{h(D_i)}, P_2\right). \end{aligned} \tag{4}$$

Proof of correctness can be derived by the bilinear property of admissible pairing. Besides, compared with individual verification, this batch verification scheme can dramatically reduce verification latency, especially when verifying a large number of signatures. Since 3 pairing operations in Equation (4) dominate the overall computation cost, the batch verification time is almost a constant if the time overhead of n MapToPoint hashings and n exponentiations is small enough to be emitted.

► **Data Processing and Signatures Aggregation**

To facilitate generating a precise and customized friending strategy, the data consumer also needs to provide her encrypted profile vector $\vec{D}_0$ and a threshold δ, where

$$\vec{D}_0 = \left(E({v_j}^2), E(v_j)^{-2} = E(-2v_j)\right)|_{j \in [1,\beta]}. \tag{5}$$

Now, the service provider can directly do matching on the encrypted profiles. To obliviously calculate the similarity difference, the service provider first preprocesses $\vec{D}_i$ and $\vec{D}_0$ by adding $E(1)$ to the first and the last places of two vectors, respectively, and gets new vectors $\vec{C}_i = \left(C_{ij}^1, C_{ij}^2, C_{ij}^3\right)|_{j \in [1,\beta]}$ and $\vec{C}_0 = \left(C_{0j}^1, C_{0j}^2, C_{0j}^3\right)|_{j \in [1,\beta]}$, where

$$\left(C_{ij}^1, C_{ij}^2, C_{ij}^3\right) = \left(E(1), E(u_{ij}), E({u_{ij}}^2)\right), \tag{6}$$

$$\left(C_{0j}^1, C_{0j}^2, C_{0j}^3\right) = \left(E({v_j}^2), E(-2v_j), E(1)\right). \tag{7}$$

After preprocessing, the service provider can compute the "dot product" of Equation (6) and Equation (7), by first applying homomorphic multiplication $\otimes$ and then homomorphic addition $\oplus$, and obtains R_{ij}, where

$$\begin{aligned} R_{ij} &= C_{ij}^1 \otimes C_{0j}^1 \oplus C_{ij}^2 \otimes C_{0j}^2 \oplus C_{ij}^3 \otimes C_{0j}^3 \\ &= E\left((u_{ij} - v_j)^2\right). \end{aligned} \tag{8}$$

Next, the service provider applies $\oplus$ to R_{ij} with $\forall j \in [1, \beta]$, and gets $R_i = E(\sum_{j=1}^{\beta} (u_{ij} - v_j)^2) = E(f(\vec{U}_i, \vec{V})^2)$.

At this point, the service provider sends R_i to the registration center for decryption. We note that for each data contributor, the registration center just needs to do one decryption, *i.e.*, she can only perform n decryptions in total. She cannot do more decryptions than required, since the service provider may still obtain a correct and complete matching strategy by revealing the profiles of all the data contributors and the data consumer. However, this case requires at least $(n+1)\beta$ decryptions. To speed up BGN decryption in outcome verification, the registration center should retain the decrypted similarity differences in storage for a preset validity period.

Upon getting $f(\vec{U}_i, \vec{V})^2$, the service provider can compare it with δ^2, and thus determines whether the data contributor o_i matches the data consumer. We assume that m data contributors are matched, and the subscripts of their pseudo identities are denoted as $\{c_1, c_2, \cdots, c_m\}$.

After data processing, to further reduce communication overhead, the service provider aggregates the signatures of m matched data contributors into one signature. In our scheme, the aggregate signature $\sigma = \prod_{i=1}^{m} \sigma_{c_i}$.

At last, the service provider sends the aggregate signature, the indexes of matched data contributors, and their encrypted profile vectors to the data consumer. To prevent the service provider from changing/revaluating the similarity differences of unmatched data contributors in the completeness verification later, their one-way hashes should also be forwarded.

► **Second-layer Batch Verification**

Similar to the first-layer batch verification, the data consumer can verify the legitimacy of m matched data sources by checking whether

$$\begin{aligned} & \hat{e}\left(\sigma, g_2\right) \\ = \quad & \hat{e}\left(\prod_{i=1}^{m} \textit{PID}_{c_i}^1, P_1\right) \hat{e}\left(\prod_{i=1}^{m} H(\textit{PID}_{c_i}^2)^{h(D_{c_i})}, P_2\right). \end{aligned} \tag{9}$$

Proof of correctness is similar to that of the first-layer batch verification, where we can just replace σ with $\prod_{i=1}^{m} \sigma_{c_i}$. Besides, the pseudo identities in the equation can be fetched from the certificated bulletin board according to their indexes.

Given a matched data contributor o_{c_i}'s pseudo identity $\textit{PID}_{c_i}$, the registration center can use her master key s_1 to perform revealing by computing

$$\textit{PID}_{c_i}^2 \ \odot \ {\textit{PID}_{c_i}^1}^{s_1} = \textit{RID}_{c_i} \odot {P_0}^r \odot {g_1}^{s_1 \cdot r} = \textit{RID}_{c_i}. \tag{10}$$

The above equation indicates that the pseudo identities of m matched data contributors can be viewed as the friending strategy, since the data consumer can resort to the registration center, as a relay, for handshaking with those matched ones.

► **Outcome Verification**

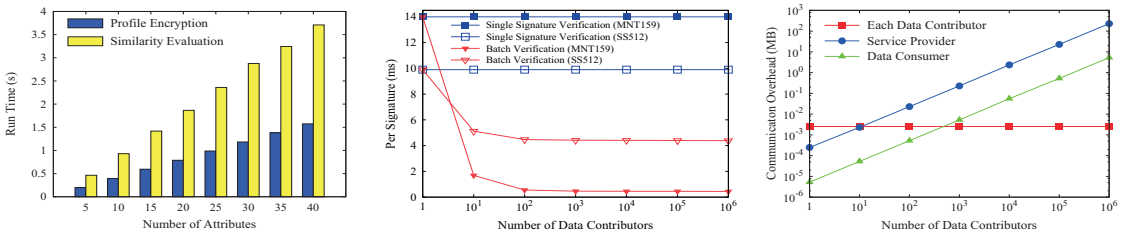
During the validity period preset by the registration center, the data consumer can verify the truthfulness of data processing via homomorphic properties. For correctness, the data consumer just needs to evaluate on the m matched profiles. Of course, for completeness, the data consumer reserves the right to do verification on the other $(n-m)$ unmatched ones. Note that the most time-consuming homomorphic multiplications are no longer needed in outcome verification, since Equation (8) can be computed by $E({u_{ij}}^2) \oplus E(u_{ij})^{-2v_j} \oplus E({v_j}^2)$. To further reduce verification cost, the data consumer can take the stratified sampling strategy in practice. We assume that the greedy service provider cheats by not evaluating each data contributor in the original data processing with a probability p. Then, the probability of successfully detecting an attempt for returning an incorrect/incomplete result, ϵ, increases exponentially with the number of checks t, *i.e.*, $\epsilon = 1-(1-p)^t$. When $p = 20\%$ and $t = 26$, the success rate ϵ is already 99.70%.

III. Evaluation Result

Table I. Computation overhead of identity-based signature scheme per data contributor.

	Preparation		Operation
Setting	Pseudo Identity Generation	Secret Key Generation	Signing
SS512	4.698ms (39.40%)	6.023ms (50.53%)	1.201ms (10.07%)
MNT159	1.958ms (57.33%)	1.028ms (30.10%)	0.429ms (12.57%)

In this section, we show the evaluation results of TPDM in terms of computation overhead and communication overhead.



(a) Computation Overhead of Profile Matching. (b) Batch Verification Efficiency. (c) Communication Overhead.

Figure 1. Performance of TPDM on Yahoo! Music Ratings Dataset.

Dataset: We use a real-world dataset, called R1-Yahoo! Music User Ratings of Musical Artists Version 1.0 [17]. The dataset contains 11,557,943 ratings of 98,211 artists given by 1,948,882 anonymous users. In this evaluation, we choose β common artists as the evaluating attributes, append each user's corresponding ratings ranging from 0 to 10, and thus form her fine-grained profile.

Evaluation Settings: We implemented TPDM using the latest Pairing-Based Cryptography (PBC) library [15]. The elliptic curves utilized in our identity-based signature scheme include a supersingular curve with a base field size of 512 bits and an embedding degree of 2 (SS512), and a MNT curve with a base field size of 159 bits and an embedding degree of 6 (MNT159). In addition, the group order q is 160-bit long, and all hashings are implemented in SHA1, considering its digest size closely matches the order of $\mathbb{G}_1$. The BGN cryptosystem is realized using Type A1 pairing, in which the group order is a product of two 512-bit primes. The running environment is a standard 64-bit Ubuntu 14.04 Linux operation system on a desktop with Intel(R) Core(TM) $i5$ $3.10GHz$.

A. Computation Overhead

We show the computation overheads of three important components in TPDM, including profile matching, identity-based signature, and batch verification.

Profile Matching: In Figure 1(a), we plot the computation overheads of profile encryption and similarity evaluation per data contributor, when the number of attributes β increases from 5 to 40 with a step of 5. From Figure 1(a), we can see that the computation overheads of these two phases increase linearly with β. This is because the profile encryption requires 2β BGN encryptions, and the similarity evaluation mainly consists of β secure "dot" products, which are both proportional to β. Additionally, when $\beta = 10$, one decryption overhead at the registration center is 1.648ms in data processing, while in outcome verification, it is in tens of microseconds.

Identity-Based Signature: We now investigate the computation overhead of the identity-based signature scheme. In this set of simulations, we set the number of data contributors to be 10000. Table I lists the average time overhead per data contributor. From Table I, we can see that the time cost of the preparation phase dominates the total overhead in both SS512 and MNT159. This outcome stems from that the pseudo identity generation employs ElGamal encryption, and the secret key generation is composed of one MapToPoint hash operation and two exponentiations. In contrast, the operation phase mainly consists of one exponentiation.

Batch Verification: To examine the efficiency of batch verification, we vary the number of data contributors from 1 to 1 million by exponential growth. The performance of the corresponding single signature verification is provided as a baseline. Figure 1(b) depicts the evaluation results using SS512 and MNT159, where verification time per signature is computed by dividing total batch verification time by the number of data contributors. From Figure 1(b), we can see that when the scale of data acquisition or data trading is small, *e.g.*, when the number of data contributors is 10, TPDM saves 48.22% and 87.94% of verification time per signature in SS512 and MNT159, respectively. When the scale becomes larger, TPDM's advantage over the baseline is more remarkable.

B. Communication Overhead

Figure 1(c) plots the communication overheads of each data contributor, the service provider, and the data consumer in MNT159, where the number of attributes β is fixed at 10 and the threshold δ takes 12. Here, the communication overheads merely count in the amount of sending content. Besides, we only consider the correctness verification. In fact, when the number of data contributors is 10^4, if we randomly check 26 unmatched ones for completeness, it incurs additional communication overheads of 252.62KB at the service provider, and 3.35KB at the data consumer.

IV. Conclusion

In this paper, we have proposed the first secure mechanism TPDM for personal data markets, achieving both truthfulness and privacy preservation. We have instantiated TPDM with the profile-matching service, and extensively evaluated its performance on a real-world dataset. Evaluation results have demonstrated the scalability of TPDM in the context of large user base from computation and communication overheads.

References

[1] M. Balazinska, B. Howe, and D. Suciu, "Data markets in the cloud: An opportunity for the database community," in *VLDB*, 2011.

[2] M. Barbaro, T. Zeller, and S. Hansell, "A face is exposed for AOL searcher no. 4417749," *New York Times*, Aug. 2006.

[3] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing," in *CRYPTO*, 2001.

[4] D. Boneh, E. Goh, and K. Nissim, "Evaluating 2-dnf formulas on ciphertexts," in *TCC*, 2005.

[5] T. W. Chim, S. Yiu, L. C. K. Hui, and V. O. K. Li, "SPECS: secure and privacy enhancing communications schemes for VANETs," *Ad Hoc Networks*, vol. 9, no. 2, pp. 189 – 203, 2011.

[6] "DataSift," http://datasift.com/.

[7] D. Eastlake and P. Jones, "US Secure Hash Algorithm 1 (SHA1)," *IETF RFC 3174*, 2001.

[8] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Transactions on Information Theory*, vol. 31, no. 4, pp. 469–472, 1985.

[9] B. C. M. Fung, K. Wang, R. Chen, and P. S. Yu, "Privacy-preserving data publishing: A survey of recent developments," *ACM Computing Surveys*, vol. 42, no. 4, pp. 1–53, Jun. 2010.

[10] "Gallup Poll," http://www.gallup.com/.

[11] "Gnip," https://gnip.com/.

[12] R. Ikeda, A. D. Sarma, and J. Widom, "Logical provenance in data-oriented workflows?" in *ICDE*, 2013.

[13] "Infochimps," http://www.infochimps.com/.

[14] "Microsoft Azure Marketplace," https://datamarket.azure.com/home/.

[15] "PBC Library," https://crypto.stanford.edu/pbc/.

[16] M. Raya and J. Hubaux, "Securing vehicular ad hoc networks," *Journal of Computer Security*, vol. 15, no. 1, pp. 39–68, 2007.

[17] "Yahoo! Webscope datasets," http://webscope.sandbox.yahoo.com/.

[18] R. Zhang, Y. Zhang, J. Sun, and G. Yan, "Fine-grained private matching for proximity-based mobile social networking," in *INFOCOM*, 2012.