Toward Transparent and Accountable Privacy-Preserving Data Classification

Yanqi Zhao, Yong Yu, Ruonan Chen, Yannan Li, and Aikui Tian

Abstract

Machine learning provides an effective approach to execute big data analysis. As a branch of machine learning, classification has been widely adopted in data processing. However, the sensitivity of data raises the concern of data privacy. How to balance data utility and data privacy is a challenging issue. Privacy-preserving data classification, which supports flexible and privacy-friendly access to datasets and data classification, enables users' data to be collected in an authenticated manner. However, the privacy-preserving data classification approach has a limitation in that the correctness of data classification cannot be guaranteed. As a consequence, it is possible for a malicious classifier to manipulate the classification result. To solve these problems, in this article, we propose a transparent and accountable privacy-preserving data classification framework, which involves a tracer to assert the behavior of the classifier and maintains the utility and privacy of data. Specifically, we take advantage of cryptography techniques to balance data privacy and data utility, and use blockchain to achieve transparency and accountability for the behavior of the classifier. To illustrate the practicability of this framework, we implement concrete cryptographic algorithms and develop a prototype system to evaluate and test its performance.

Introduction

With the development of information technology, people produce a huge amount of data, such as browsing history and social data on the Internet and users' behavior data collected by smart home applications. According to IDC (https://www.seagate.com/files/www-content/our-story/trends/files/dataage-idc-report-final.pdf) forecasts, the global datasphere will grow from 45 ZB in 2019 to 175 ZB by 2025. Massive data could create enormous value. For example, Alibaba (https://damo.alibaba.com/labs/ai?lang=en) and IBM (https://www.ibm.com/sg-en/products/category/technology/cognitive-computing-and-AI) are actively collecting data from their user platform and taking advantage of artificial intelligence (AI) and machine learning to conduct big data analysis, make market decisions, and improve their services.

The employment of AI and machine learning technology brings convenience to people's lives, while data security incidents happen from time to time. In 2019, the U.S. AMCA (https://www.zdnet.com/article/amca-data-breach-has-now-gone-over-the-20-million-mark/) revealed about 20 million U.S. citizens' health data, including user name, social security number, and date of birth. In 2018, Facebook (https://en.wikipedia.org/wiki/Facebook-Cambridge_Analytica_data_scandal) revealed the personal information of more than 50 million users to a third-party company for data analysis and obtained huge profits. These events make us pay more and more attention to data security, especially the privacy of sensitive data.

In practice, when data are collected from users, protecting the privacy of these data is of crucial importance. In an intelligent transportation system, the navigation system provides users with navigation services by collecting their location information. If such information is leaked, malicious attackers will know a vehicle's driving routes and destinations, and even predict a user's behavior habits, which poses a threat to the user's security and privacy. Wearable activity trackers can measure and collect the blood pressure and heart rate information of a user for health condition analysis. If the health data leaks, it seriously threatens user privacy. Therefore, many countries and organizations enact relevant laws and regulations to protect the privacy of users, for example, the General Data Protection Regulation (GDPR) (https://gdpr-info.eu/) of Europe. Classification technology, a branch of machine learning, has been widely used in big data processing. By using algorithms such as decision tree or support vector machine, we can implement data classification and figure out the correlation among different data. However, these classification techniques only work for plaintext data, not encrypted data. How to protect data privacy while ensuring data utility is a tricky issue.

Privacy-Preserving Data Classification

Privacy-preserving data classification provides a solution for data utility and data privacy. As shown in Fig. 1, a privacy-preserving data classification framework enables user data to be collected in an authenticated and unlinkable manner, which guarantees the availability of data and provides flexible and privacy-friendly access to datasets. In the framework, there are four entities, namely, an issuer, a user, a verifier, and a classifier. The issuer issues a credential to users and authorizes them to collect data. The user collects data and generates a signature on the data. Then the user sends authenticated data to a verifier. Being a data collector or a data center, the verifier encrypts

Digital Object Identifier:
10.1109/MNET.011.2000662

Yanqi Zhao and Yong Yu (corresponding author) are with Xi'an University of Posts and Telecommunications; Ruonan Chen is with Beihang University; Yannan Li is with the University of Wollongong; Aikui Tian is with Shandong University of Technology.

0890-8044/21/$25.00 © 2021 IEEE

the authenticated data and requests the classifier (data processor) to execute data classification on encrypted data. Then the classifier executes the data classification on the ciphertext without learning the plaintext by using a classification key. For instance, a healthcare provider would provide discounts for low-income people. while, the information of users' salaries is stored by the tax authority. Therefore, in order to check whether a user is eligible for a discount, the classifier (data processor) needs to verify the user's income.

Linkable group signature [1] provides an approach to balance data utility and data privacy. The classifier can use a linking key to execute classification operations on users' data. Furthermore, group signature with user-controlled linkable [2, 3] could also realize the data classification by linking the same pseudonyms. However, this approach cannot classify different types of pseudonyms. Recently, a selective linkable group signature [4] was proposed, which enables user data to be collected in an authenticated manner and the privacy-preserving data classification to be executed effectively. However, these privacy-preserving data classification approaches have a common limitation. That is, they cannot check the correctness of data classification, which could enable a malicious classifier to manipulate the classification results. In order to prevent malicious behavior of a classifier, we need to provide transparency and accountability for privacy-preserving data classification. Transparency is a legal requirement, which has been recognized as one of key principles in the European Data Protection Directive. Transparency guarantees that data subjects have the right to be informed when their data are being processed. Accountability means that it is able to detect and check the malicious behavior of a classifier effectively. However, the known privacy-preserving data classification frameworks do not provide the features of transparency and accountability.

Our Contributions

To guarantee the correctness of privacy-preserving data classification while ensuring data privacy, we propose a transparent and accountable privacy-preserving data classification framework. In the framework, there is a classifier to execute privacy-preserving data classification on ciphertexts. A verifier (data collector) can sample the result of data classification, and then a tracer asserts the correctness of privacy-preserving data classification. The tracer will generate an accountability proof for the result of data classification. Then anyone can verify the accountability proof to check the correctness of privacy-preserving data classification.

We take advantage of cryptography techniques to balance data privacy and data utility, and use blockchain technology (http://bitcoinbook.cs.princeton.edu/) to achieve transparency and accountability for the behavior of the classifier. We propose a generic construction for transparent and accountable privacy-preserving data classification, which is based on blockchain, Merkle hash tree [5], group signatures [6] with pseudonym [7], and re-randomizable encryption. Finally, we implement the concrete cryptographic algorithms and develop a prototype system to test the practicability of transparent and accountable privacy-preserving data classification.

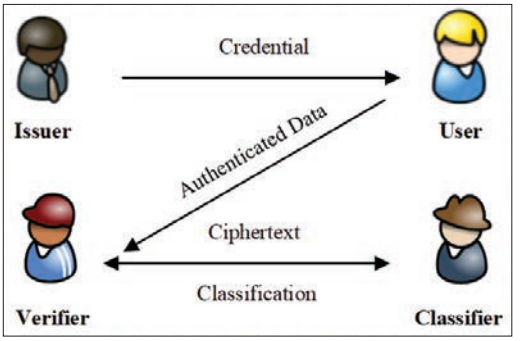

FIGURE 1. Privacy-preserving data classification framework.

Organization

We review the building blocks in the following section, and then the framework of transparent and accountable privacy-preserving data classification is given. Following that, we present a generic construction of transparent and accountable privacy-preserving data classification. With the concrete schemes, we implement performance evaluation. Finally, we conclude this article.

Building Blocks

In this section, we review the building blocks used in the proposed framework, including blockchain, smart contracts, Merkle hash tree, group signatures, and re-randomizable encryption.

Blockchain and Smart Contracts

Blockchain, a distributed ledger that combines a peer-to-peer network, cryptographic primitives, and consensus, is the backbone technology of the first cryptocurrency system, Bitcoin [8]. Blockchain has the features of decentralization, transparency, public auditing, and anti-tampering. In a blockchain network, each node runs consensus algorithms, such as proof of work (PoW) in Bitcoin, to generate a new block. Each block records the transactions, which include the addresses of the payer and receiver, and the transaction amount. When a transaction is published on blockchain, one can validate the transaction. In the blockchain, we can create more expressive applications by deploying the smart contract [9], which is an automatic execution program. We can use the scripting languages to deploy simple smart contracts in the Bitcoin system. Moreover, we can deploy more complex smart contracts in Ethereum [10], which is a public blockchain platform for the next-generation distributed applications. We can execute smart contracts using the Solidity language (https://solidity.readthedocs.io/en/v0.5.7/). The blockchain can execute the contract functions and record the transaction information. The operations of smart contracts can be publicly verified thanks to the features of open access of blockchain.

Merkle Hash Tree

The Merkle hash tree [5] is one of the authenticated data structures used to guarantee the integrity of transactions in the blockchain. A Merkle hash tree M is a labeled binary tree denoted as Y

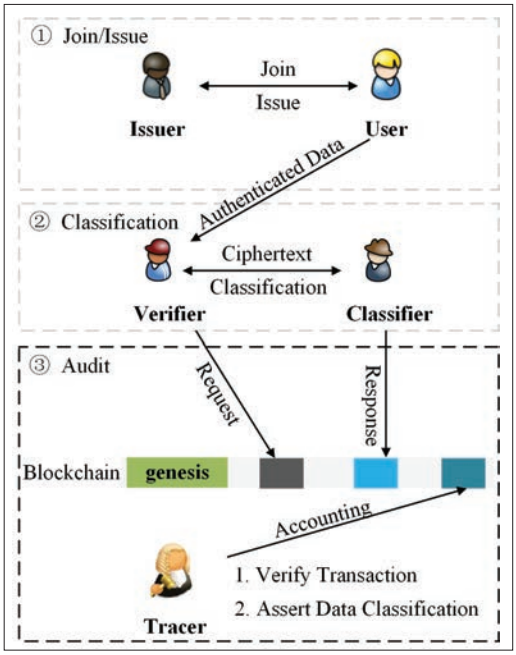

FIGURE 2. The framework of transparent and accountable privacy-preserving data classification.

$\leftarrow Mtree(x_1, \cdots, x_n)$, where x_i denotes the ith leaf node. When $n = 8$, we can compute the value of a chlid node $v_{12} = H(x_1, x_2)$ by leaf nodes x_1, x_2 in the Merkle hash tree. Then we compute the other value for non-leaf nodes $v_{1234} = H(v_{12}, v_{34})$ by non-leaf nodes v_{12}, v_{34}. Finally, we output a root value $Y = Mtree(x_1, \cdots, x_8)$ of the Merkle hash tree.

Using the Merkle hash tree, we can prove whether a leaf node x_i is in the tree or not. If we want to prove x_1 is in the tree, we can generate a proof *MProof* with $\pi = \{x_1, x_2, v_{34}, v_{5678}\}$. Then, using the *MVerify* algorithm, we can check whether leaf node x_1 is in the tree or not.

Group Signature

Group signature [6, 11] is a kind of privacy-preserving signature that allows a signer to sign on behalf of the group anonymously. Then a verifier could verify whether the signature was signed by a valid group member without knowing the identity. There are four entities in a group signature, namely, group manager, user, verifier, and opener.

Group Manager: The group manager issues group signing secret keys for users, and then the users can join the group.

User: Each user interacts with the group manager to join the group and obtains a group signing secret key. Then the user can sign on behalf of the group.

Verifier: A verifier can validate a group signature without knowing the identity of the signer.

Opener: An opener can trace and reveal the user's identity, which is used to prevent the behavior of abusing anonymity.

The linkable group signature [1] is a special kind of group signature, which includes an extra role of linker.

Linker: The linker can check if two group signatures are from the same user.

The group signatures and linkable group signatures have become important building blocks for certain privacy-preserving applications, such as anonymous reputation systems and privacy-preserving data classification.

Re-Randomizable Encryption

In order to protect the privacy of data, we usually use encryption techniques. An encryption scheme [12] includes three algorithms. The *Keygen* algorithm generates the public key and private key. The *Enc* algorithm encrypts a message m with public key and outputs a ciphertext c. The *Dec* algorithm inputs the ciphertext c and private key, and returns a plaintext m. Encryption is named homomorphic encryption if the operations on ciphertexts are equivalent to the corresponding operations on plaintexts, that is, $Enc(m_1 * m_2) = Enc(m_1) \circ Enc(m_2)$. A homomorphic encryption usually has the property of public re-randomizability [4]. That is, with the public key, one can generate a re-randomized ciphertext c', which is indistinguishable from the original ciphertext c.

Architectural Model

In this section, we describe the framework of transparent and accountable privacy-preserving data classification.

The Detail of the Framework

The framework of transparent and accountable privacy-preserving data classification is composed of five entities: issuer, users, verifier, classifier, and tracer, as shown in Fig. 2.

The framework consists of three phases:

- Join/Issue: A user interacts with the issuer to obtain a credential. Then the user generates the authenticated data.
- Classification: The verifier interacts with the classifier to execute privacy-preserving data classification.
- Audit: A tracer verifies the transactions submitted by the verifier and the classifier. Then the tracer outputs an accountability proof to assert the correctness of the data classification result.

Blockchain, as a decentralized data storage ledger, records all transactions that have been previously submitted. In the blockchain network, every participant can access the data on blockchain and post transactions on blockchain. Here, we assume the ledger has a genesis block that includes the security parameters. The operation of five entities are as follows:

- *Issuer:* The issuer initializes the system and authorizes a user to collect data by issuing a credential. Then the issuer puts the public parameters on blockchain.
- *User:* Each user interacts with the issuer to get a credential and signs on the data. The user sends the signature with authenticated data to the verifier.
- *Verifier:* The verifier, as a data collector, checks the validity of authenticated data. Then the verifier interacts with the classifier and sends the ciphertext data to the classifier to execute privacy-preserving data classification. The verifier will sample the result of data classification and request a tracer to assert the correctness of privacy-preserving data classification.

- *Classifier:* The classifier, as a data processor, interacts with a verifier and holds a classification key, which can be used to execute privacy-preserving data classification and respond to the sample request.
- *Tracer:* The tracer interacts with the blockchain and verifies the validity of transactions from the verifier and classifier. It asserts the correctness of data classification and puts the result on the blockchain.

Design Goals

The transparent and accountable privacy-preserving data classification has the following design goals:

Anonymity: The signature generated by the user does not reveal relevant information about the user's identity.

Classify Blindness: A classifier cannot learn about the plaintext data and classified pseudonyms it computes.

Non-Frameability: The authenticated data are available from honest users. An adversary should be unable to impersonate an honest user.

Accountability: A tracer can always execute the tracing and auditing operations on the classified data and the data to be classified.

A Generic Construction

In this section, we propose a generic construction for transparent and accountable privacy-preserving data classification, which is based on the group signatures with pseudonym, Merkle hash tree, and re-randomizable encryption. In the blockchain network, we assume each participant has an account address and can generate the transactions and call smart contracts. Smart contracts record the user's request or assert some claims about the user operations, which could guarantee the transparency of privacy-preserving data classification. In terms of accountability, smart contracts provide efficient operations to verify the correctness of privacy-preserving data classification. Then anyone can check its correctness due to the blockchain setting.

The process of transparent and accountable privacy-preserving data classification is illustrated in Fig. 3.

Join/Issue: The issuer sets up the system. Spe cifically, it generates the public parameters and generates a *PPTx* transaction. Then it publishes the public parameters and *PPTx* on blockchain. The issuer interacts with a user to execute the authentication operation and issues a credential to the user. The user collects data and generates a group signature on data with the credential, then sends the authenticated data with signature to the corresponding verifier.

Classification: The verifier checks the validity of the signature and then generates a re-randomized ciphertext c' by the blind operation of re-randomizable encryption. The verifier interacts with the classifier to conduct the privacy-preserving data classification. It sends a batch of ciphertexts $c_1, \cdots, c_k$ (k is the number of ciphertexts) to the classifier. Then the verifier computes a root value of ciphertexts $c_1, \cdots, c_k$ by using the Merkle hash tree *Mtree* algorithm. Next, the verifier computes a root value for the authenticated data by the *Mtree* algorithm. Then the verifier generates a *BlindTx* transaction and publishes the root value to the blockchain. Upon receiving ciphertexts $c_1, \cdots, c_k$, the classifier executes classification operation. It runs the equivalence test algorithm with the classification key to execute equality tests on the ciphertexts and outputs the classification result *Cla*. The classifier computes a root value of the classification result *Cla* by using *Mtree* and generates a *ClaTx* transaction. Then it publishes the root value to the blockchain.

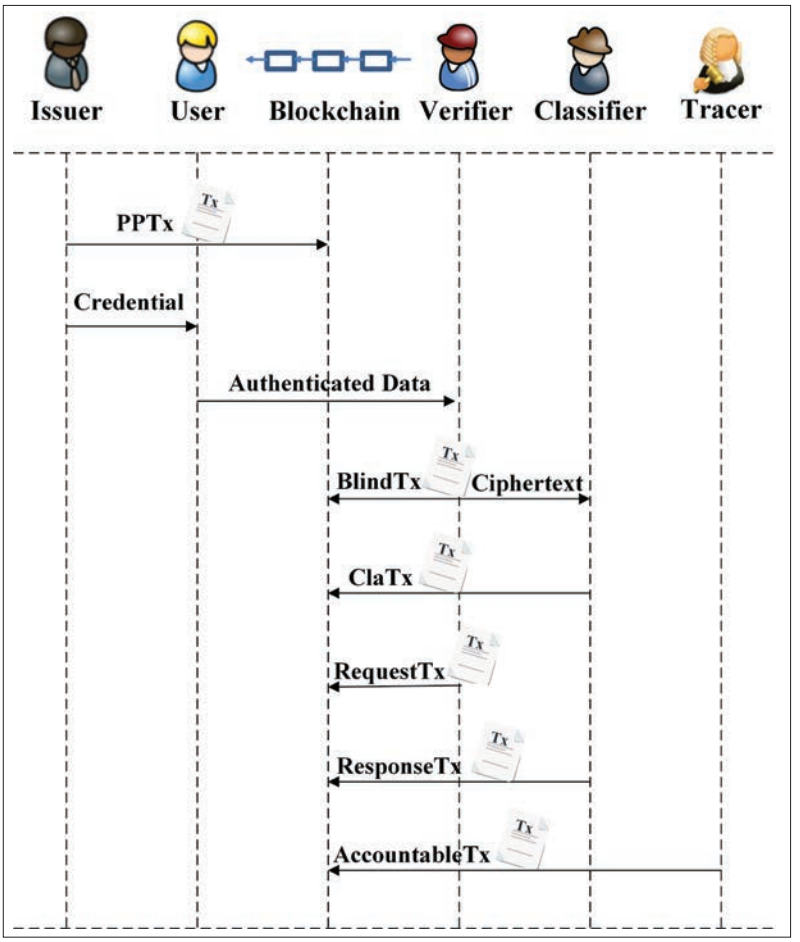

FIGURE 3. The process of transparent and accountable privacy-preserving data classification.

Audit: The verifier requests to assert the correctness of privacy-preserving data classification using the root value of the classification result. The verifier first samples the classified data by generating an index set, such as (2, 4, 8). It generates a request *RequestTx* transaction and publishes the index set with a request proof to the blockchain. The classifier retrieves the data in the index set and calls *MProof* to generate a proof for them. Then the classifier generates a *ResponseTx* transaction and publishes a response proof to the blockchain. Using the transactions, the tracer executes the accounting operation and verifies the proof of Merkle hash tree by running the *MVerify* algorithm. To assert the correctness of data classification, the tracer uses the tracing key to check the sample data. It generates an accountability proof for the sample data, which includes the pseudonym identify, the proof of possession of tracing key, and correctness proof of data classification. Then the tracer constructs an *AccountableTx* transaction and publishes the

accountability proof to the blockchain. Using the *AccountableTx*, a verifier can check the validity of assertion.

Performance Evaluation

In this section, we test the efficiency of the transparent and accountable privacy-preserving data classification with concrete cryptography schemes and evaluate the time overhead of the cryptographic algorithms. We deploy a prototype system on an Ethereum geth client to evaluate the functions of transparent and accountable privacy-preserving data classification and test its practicability.

Implement Analysis

We implement the cryptographic algorithms used in the framework of transparent and accountable privacy-preserving data classification. Our experiment platform is based on the intel® Core™ i5-4590S CPU 3.00 GHz, 6.00 GB RAM with the Ubuntu 16.04 LTS operation system (OS). We develop the codes of the cryptographic primitives by using python programming language with python version 3.6.5 (https://www.python.org/downloads/release/python-365/). For the pairing library, we use the BN curve for bilinear pairing (https://github.com/ethereum/py_pairing). We implement 1000 rounds for each algorithm to obtain an average running time.

We execute three types of group signatures including CL type group signatures [13], PS type group signatures [14] and BBS type [15] group signatures [4] in the first phase *Join/Issue*. We evaluate the time cost of signature algorithm and verify algorithm for these group signatures. The time cost of three types of group signatures is shown in Fig.4a. Experiment results show that the BBS type signature [4] is the fastest one to generate a group signature, and its verification time is also short. The time cost of the CL type group signature [13] is the highest because multiple pairing operations need to be executed in the sign and verification algorithms. In addition, generating a PS type group signature costs 4.7183 s, and running the verify algorithm costs 14.0763 s. The time cost of PS type group signature [14] is between the CL type scheme and the BBS type scheme. Thus, it is optimal to use the BBS type scheme to construct a transparent and accountable privacy-preserving data classification scheme.

In the *Classification* phase, we evaluate the cryptographic algorithms including re-randomizable encryption and privacy-preserving data classification operations [12]. The time cost of encryption algorithm and classification algorithm is shown in Fig. 4b. Experiment results show the re-randomizable encryption costs about 90 s for 1600 data tuples, and the classification operation costs about 40 s. A classifier executes the data classification on the ciphertext data, and the average running time is 0.025 s on each data tuple.

Finally, we evaluate the cryptographic algorithms in the *Audit* phase. We evaluate the accounting function by testing the request operation of the verifier, the response operation of the classifier, and the accounting operation of the tracer. The time cost of these three operations is shown in Fig. 5. We construct a Merkle hash tree with eight leaf nodes. For a request operation, the verifier samples the data with $|Index|$=1/3/5/7. The classifier responds to the related request in $|Index|$. Experiment results show the time cost of request operation and response operation increases with the increase of the size of $|Index|$. The time cost of accounting operation is proportional to the amount of sample data $|Index|$.

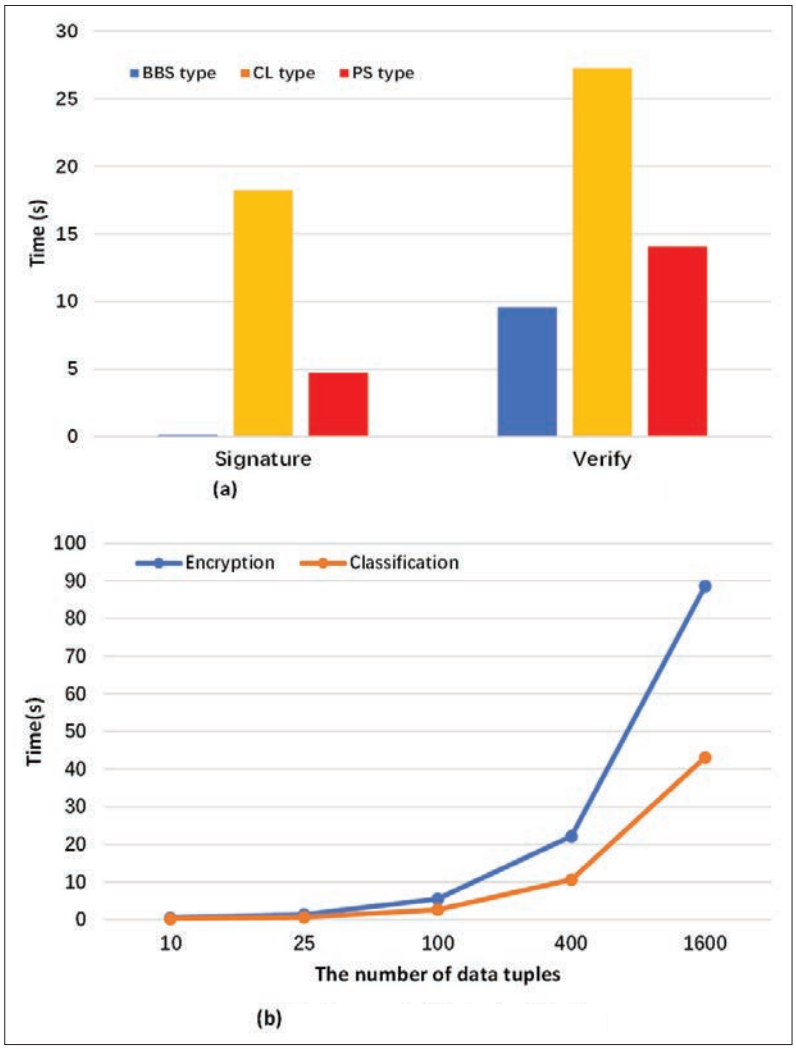

FIGURE 4. The implement of cryptographic algorithms in *Join/Issue* phase and *Classification* phase: a) the time cost of three type group signatures; b) the time cost of data classification.

Prototype System

Here, we develop a prototype system for the transparent and accountable privacy-preserving data classification. We use Solidity language to develop the smart contract. Solidity is an object-oriented, high-level language that can implement the smart contract on an Ethereum geth client (https://geth.ethereum.org/). We deploy the smart contract and estimate the gas cost of calling the smart contracts to generate the related transactions. We implement the request operation in the Audit phase, which is based on a Merkle hash tree with n = 16 leaf nodes and the size of sample data set $|Index|$ = 5. The gas cost of transactions is provided in Table 1. The *BlindTx* and *ClaTx* transactions only embed the root value of the Merkle hash tree, so the implementation of *BlindTx* and *ClaTx* transactions costs low gas. The sample data and the corresponding ciphertext data are uploaded to blockchain, so the

Transaction	Gas units	Gas cost (ether)
PPTx	153356	0.00306712
BlindTx	62773	0.00125546
ClaTx	42204	0.00084408
RequestTx	4586737	0.09173474
ResponseTx	2322720	0.04645440
AccountableTx	382970	0.00765940

TABLE 1. Gas cost of transactions.

RequestTx and *ResponseTx* cost much gas. The *AccountableTx* transaction includes an accountability proof of the correctness of data classification, which will be uploaded to blockchain and cost 382,970 gas.

Conclusion

Privacy-preserving data classification provides the utility of data while protecting privacy of data. In this article, we propose a transparent and accountable privacy-preserving data classification framework, which could protect the privacy of data while ensuring the correctness of privacy-preserving data classification. We implement concrete cryptographic algorithms for performance evaluation and deploy smart contracts in the Ethereum geth client for testing the practicability of the proposed construction.

Acknowledgments

This work is supported by the National Natural Science Foundation of China under Grants 61872229 and U19B2021, the Key Research and Development Program of Shaanxi under Program 2020ZDLGY09-06, 2021ZDLGY06-04, and the Blockchain Core Technology Strategic Research Program of the Ministry of Education of China under Grant 2020KJ010301.

References

[1] J. Y. Hwang *et al.*, "Group Signatures with Controllable Linkability for Dynamic Membership," *Info. Sciences*, 2013, 222, pp. 761–78.
[2] J. Camenisch and A. Lysyanskaya, "An Efficient System for Nontransferable Anonymous Credentials with Optional Anonymity Revocation," *Proc. Int'l. Conf. Theory and Applications of Cryptographic Techniques 2001*, pp. 93–118.
[3] J. Camenisch and A. Lysyanskaya, "Signature Schemes and Anonymous Credentials from Bilinear Maps," *Proc. Annual Int'l. Cryptology Conf. 2004*, pp. 56–72.
[4] L. Garms and A. Lehmann, "Group Signatures with Selective Linkability," *Proc. Int'l. Conf. Practice and Theory of Public-Key Cryptography* 2019, pp. 190–220.
[5] S. Dziembowski, L. Eckey, and S. Faust, "FairSwap: How to Fairly Exchange Digital Goods," *Proc. ACM Conf. Computer and Commun. Security* 2018, pp. 967–84.
[6] D. Chaum and E. Van Heyst. "Group Signatures," *Proc. Int'l. Conf. Theory and Applications of Cryptographic Techniques 1991*, pp. 257–65.
[7] J. Camenisch and A. Lehmann, "(Un)linkable Pseudonyms for Governmental Databases," *Proc. ACM Conf. Computer and Commun. Security 2015,* pp 1467–79.
[8] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," 2008; http://pdos.csail.mit.edu/6.824/papers/bitcoin.pdf.
[9] N. Szabo, "Smart Contracts," 1994; http://www.fon.hum.uva.nl/rob/C ourses/InformationInSpeech/CDROM/Literature/LOTwinterschoo l2006/szabo.best.vwh.net/smart.contracts.html.
[10] G. Wood, "Ethereum: A Secure Decentralised Generalised Transaction Ledger," Ethereum Project Yellow Paper, 151, 1–32, 2014; http://www.cryptopapers.net/papers/ethereum-yellowpaper.pdf.

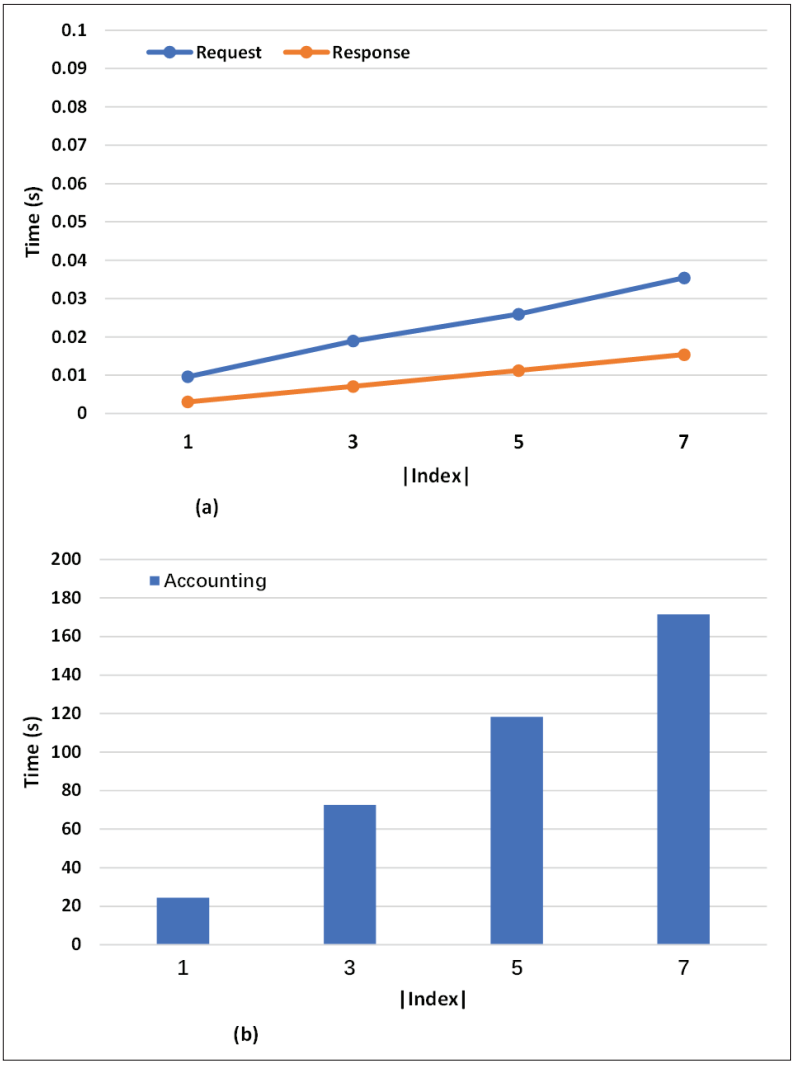

FIGURE 5. The implementation of cryptographic algorithms in the *Audit* phase: a) the time cost of request and response operations; b) the time cost of accounting operation.

[11] J. Camenisch and M. Stadler, "Efficient Group Signature Schemes for Large Groups," *Proc. Annual Int'l. Cryptology Conf. 1997*, pp. 410–24.
[12] T. ElGamal, "On Computing Logarithms Over Finite Fields," *Proc. Annual Int'l. Cryptology Conf. 1985*, LNCS, vol. 218, Aug. 1986, pp. 396–402
[13] P. Bichsel *et al.*, "Get Shorty via Group Signatures without Encryption," *Proc. Security and Cryptography for Networks 2010*, pp. 381–98.
[14] D. Pointcheval and O. Sanders, "Short Randomizable Signatures," *Cryptographers Track at the RSA Conf.*, 2016, pp. 111–26.
[15] D. Boneh, X. Boyen, and H. Shacham, "Short Group Signatures," *Proc. Annual Int'l. Cryptology Conf. 2004, LNCS*, vol. 3152. Springer, Aug. 2004, pp. 41–55.

Biographies

Yanqi Zhao is currently an associate professor at Xi'an University of Posts and Telecommunications, China. His research interests are cryptography, machine learning, and blockchain.

Yong Yu is currently a professor at Xi'an University of Posts and Telecommunications. His research interests are cryptography and cloud security.

Ruonan Chen is currently a Ph.D. candidate at Beihang University, China. Her research interests are machine learning and blockchain.

Yannan Li is currently a Ph.D. candidate at the University of Wollongong, Australia. Her research interests are blockchain and cloud security.

Aikui Tian is currently a professor at Shandong University of Technology, China. His research interest is information security.