Functional Privacy-preserving Outsourcing Scheme with Computation Verifiability in Fog Computing

Wenyi Tang[1], Bo Qin[1*], Yanan Li[2] and Qianhong Wu[2,3]
[1] School of Information, Renmin University of China
Beijing, 100872, China
[2] School of Cyber Science and Technology, Beihang University
Beijing, 100191, China
[3] Science and Technology on Information Assurance Laboratory,
Beijing, 100000, China
[e-mail: Bo.qin@ruc.edu.cn]
*Corresponding author: Bo Qin

*Received November 25, 2018; revised May 14, 2019; accepted July 8, 2019;
published January 31, 2020*

Abstract

Fog computing has become a popular concept in the application of internet of things (IoT). With the superiority in better service providing, the edge cloud has become an attractive solution to IoT networks. The data outsourcing scheme of IoT devices demands privacy protection as well as computation verification since the lightweight devices not only outsource their data but also their computation. Existing solutions mainly deal with the operations over encrypted data, but cannot support the computation verification in the same time. In this paper, we propose a data outsourcing scheme based on an encrypted database system with linear computation as well as efficient query ability, and enhance the interlayer program in the original system with homomorphic message authenticators so that the system could perform computational verifying. The tools we use to construct our scheme have been proven secure and valid. With our scheme, the system could check if the cloud provides the correct service as the system asks. The experiment also shows that our scheme could be as effective as the original version, and the extra load in time is neglectable.

Keywords: Data outsourcing; Computation verifiability; IoT Security; Fog Computing

This paper is supported by the National Key R&D Program of China through project 2017YFB1400700, the Natural Science Foundation of China through projects 61772538, 91646203 and 61672083, by the foundation of Science and Technology on Information Assurance Laboratory through project 61421120305162112006, by the National Cryptography Development Fund through project MMJJ20170106.

http://doi.org/10.3837/tiis.2020.01.016 **ISSN : 1976-7277**

1. Introduction

With the development of cloud computing, the technology of Internet of Things has made great progress. A large amount of portable, mobile and lightweight IoT devices have entered into people's daily lives, and such explosive growth also brings great challenges to the traditional cloud computing model. A major issue is that the cloud service is such a highly centralized service which will face heavy load of mass data under the limited network bandwidth. As a result, the service provided by the cloud might be both imbalance and unstable depending on the density and distance of IoT devices.

Fog computing was first proposed and officially defined by Bonomi from Cisco [1]. It emphasized the concept of edge-network and edge-computing. In fog computing, there is an extra layer between the centralized cloud and personal IoT devices. Such a layer consists of many cloud services so called edge-cloud services. The edge-cloud could be a personal computer, local cloud provider and some geographically closer cloud provider. Compared to the traditional cloud computing mode, the fog computing invokes the distributed edge-cloud nodes as a fog layer to address the need of high traffic load and latency-sensitive applications in network. Since the distance between IoT devices and edge-cloud is shortened and the traffic load between cloud and IoT devices is spread out, the service IoT devices receive could be much more stable.

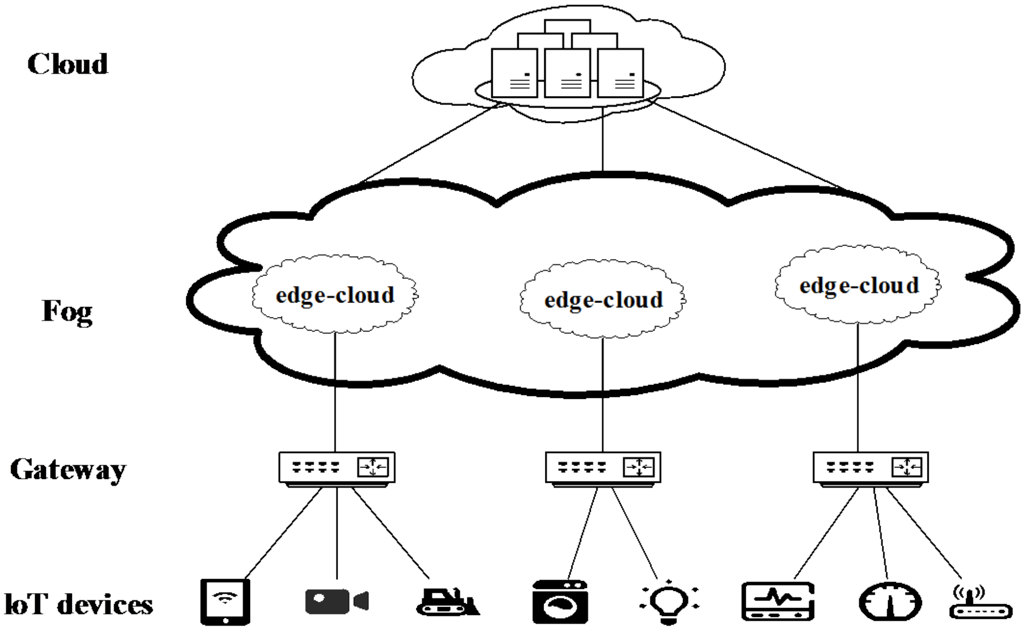


Fig. 1. Overview of fog computing

Fig. 1 shows an overview of fog computing. The IoT devices work in the bottom of network and the information streams of devices with close geographical position will be gathered and dealt by a universal gateway. This is a general scheme used in some scenarios like home, companies or office buildings. These information streams will be transmitted to the edge-cloud as described before, which is much closer in distance and more effective in timeliness than transmitting to the tradition cloud.

To make the IoT devices with low power consumption and long working hours, they are designed to be dramatically lightweight, which means such devices tend to outsource their data to the edge-cloud service provider (eCSP) rather than store locally under the fog computing architecture. The lightweight IoT devices are widely used in different fields like smart home, e-health and so on. Such devices are so close to our daily lives that they collect data about every action we take for analysis and providing better service. Researches have been done to show that the data collected by the IoT devices are quite relevant to user's privacy [2], [3], therefore the data outsource scheme of such devices brings huge threat. In the edge-cloud under the concept of fog computing, the data-outsource scheme confronts roughly the same threats as in the traditional cloud computing scheme, such as dishonest eCSP, curious eCSP and attackers obtaining the whole database. There are also some new demands should be concerned in the edge-cloud like the high timeliness requirements, temporary storage, and accurate computing.

With the popularization of Database as a Service (DBaaS) [4] and the development of cryptology, researchers have done many studies on the outsourcing encrypted database which support efficient operation over ciphertext data. Such schemes can be used to address the privacy problems in edge-cloud under fog computing. But, to ensure the service provided by edge-cloud is solid (e.g. some functions are computed correctly), verifiability is also necessary.

In this paper, we propose an encrypted data outsourcing scheme to meet the needs and challenges in the edge-cloud in Fog Computing. Our scheme not only supports the data manipulation over encrypted data, but also supports computation verification of linear functions. The principle of our scheme is to encrypt data before uploading them to the edge-cloud. While the devices require such data (or results of some functions), the plaintext requests are firstly sent to a trusted transfer proxy, where an interlayer program is running on. The proxy will submit them to cloud after transfer the requests into the corresponding ciphertext requests. Then the proxy will first verify the results (if needed), then decrypt them, and transmit plain-result to the devices after getting responds from the cloud. Specifically, we use an encryption with properties of order-preserving and additive homomorphism to perform an efficient query and simple linear arithmetic operation over encrypted data. Furthermore, an authentication tag attribution is attached to the data so that the result of computation could be efficiently verified. Experiments show that the extra load of data storage is acceptable and the transfer layer works correctly and efficiently.

Our work is inspired by the progress of outsourcing database and verifiable computation, especially the work of [5] and [6], which implement practical applications in both Excel (formula-based data management) and database (SQL-based data management). Both applications are capable of describing and executing linear functions. We studied the features and demands of edge-cloud in fog computing and realized that the solution of contradiction between privacy protection and service providing is that eCSP has to perform efficient and accurate data managements over encrypted data. Therefor we choose a practical encrypted database scheme and an efficient verifiable computing scheme to construct our system.

2. Related Work

Secure outsourcing has been a hotspot in researches for a long time and progress has been made in many ways, such as access control, operational encryption and verifiability.

Access control policies provide prior approach to ensure authorized visit of legal user to the resource. Traditional role-based access control (RBAC) and attribute-based access control (ABAC) cannot meet the demand of cloud computing, because the cloud environment demand dynamic, scalable access control policies. Kuhn et. al. [7] creatively combined the RBAC and ABAC to perform efficient and dynamic access control. [8] and related work can evaluate the risk of an access and provide dynamic access control according to the evaluation. Other work like [9]–[11] are also being widely used to provide dynamic fine-grained access control.

Access control could be useless if the illegal access comes from CSP itself, in other word, access control is powerless in front of a curious CSP, which is highly possible in practice. The best way to address that is encrypting the data before outsource them. Furthermore, to make the CSP provide service perfectly, the encryption should be operational too. Fully homomorphic encryption proposed by Gentry et. al. [12] is the ideal scheme in data outsourcing, but, because of the enormous expenditure of both time and space, it still has long way to go in practicability. Order-preserving encryption (OPE) provide efficient range query over ciphertext, e.g. [5], [13]–[15], such scheme holds great practical value in the present. Wang et. al. presents an order-preserving encryption with additivity, which takes advantages of OPE and addition homomorphism, making the operations over ciphertext much more practical. [6] and [5] both are applications built on OPEA, which achieve functional as well as efficiency. Not only operations over numeric data, but operations over text data also attract many attentions. Boneh et. al. [16] proposed the first searchable encryption over single key word, then several schemes [17], [18] which support multiple keyword search were proposed. After that the schemes [19], [20] focused on the fuzzy query over encrypted text data. In 2011, MIT proposed the famous CryptDB [21] that supporting almost all kinds of data manipulation. CryptDB uses Onion encryption scheme to encrypt the data with multiple encryption algorithms, which does not comply to DBaaS framework. And it also has problem of efficiency in practical.

Many researches have been done to construct secure data outsourcing and verification. Deswarte et. al. [22] proposed an algorithm using RSA encryption based on hash functions to perform data verification on cloud server. Krohn and Freedman [23] constructed a scheme to verify the integrity of data in cloud with homomorphic hash function, which reduces the computational effort greatly. At present, the basic technology of data integrity verification can be divided into two categories: Provable Data Possession (PDP) and Proofs of Retrievability (POR). Ateniese [24] firstly putted out the definition of PDP and introduced a PDP model based on homomorphic signature. Swminathana et. al. [25] proposed a S-PDR built on homomorphic hash function. In the same year, Juels [26] gave the formal definition of POR for the first time and proposed a POR based on "Sentinels". After that, there are some data integrity verifying scheme using homomorphic message authenticator. Shacham [27] constructed a homomorphic message authenticator scheme based on symmetric cryptography system, which could be applied to POR, such scheme is also be called as POR-PRF (POR based on Pseudo-Random Function). Gennaro et. al. officially propose the formalize definition of fully message homomorphic authenticator in [28] as well as a homomorphic authenticator scheme based on fully homomorphic encryption, which has huge problem in efficiency. Catalano et. al. build a scheme in [29] only based on a pseudo-random function, using a polynomial to authenticate the messages and achieving computation verification over an arithmetic circuit. [30] uses homomorphic signatures to run efficient verification for polynomial functions. [29], [31], [32] focused on the computation verification over encrypted data.

3. Tools

In this section, we will introduce the principle of 2 existing cryptographic tools used in our scheme, order-preserving encryption with additivity and homomorphic messages authenticator.

3.1 Order-preserving Encryption with Additivity

In our scheme, we use the order-preserving encryption proposed in [5], [6]. An OPEA scheme $E : X \to Y$ is an encryption with following two properties：

1. Order-preservation: : $\forall\, a, b \in X$, if $a < b$, then $E(a) < E(b)$;
2. Additive order-preservation: $\forall\, a, b, c \in X$, if $a + b < c$, then $E(a) + E(b) < E(c)$.

where X, Y represent the plaintext domain and ciphertext domain respectively. [5] also proposed an OPEA extend version and a noise-increasing method such that the summation of ciphertext could be decrypted correctly into the summation of corresponding plaintext. In other words, OPEA (along with OPEA extend version) holds the additive homomorphism. OPEA scheme can be formally defined as a symmetrical encryption algorithm with 3 tuples of sub-algorithms working in order:

BoundryGen(): use a security parameter to generate a set of boundaries (equivalent to secret keys).

Encrypt(m): take a plaintext m as input, using the generated boundaries to output a ciphertext m′.

Decrypt(m′): take a ciphertext m′ as input, using the generated boundaries to output a plaintext m.

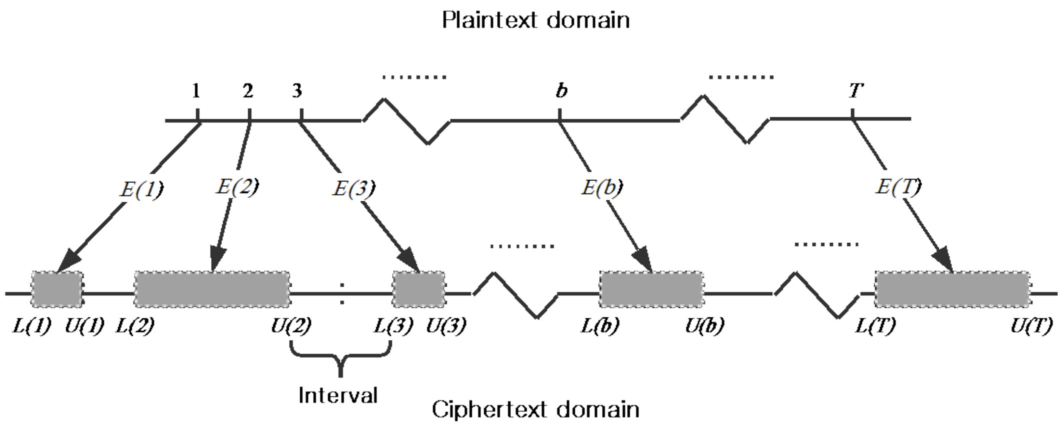


Fig. 2. The one-to-many mapping of OPEA

Apparently, as showed in **Fig. 2**, the OPEA algorithm is an one-to-many mapping, which makes OPEA a probabilistic encryption algorithm. As the theory in [15], it is necessary for an order-preserving encryption to achieve ideal security level.

3.2 Homomorphic Message Authenticator

We use the Homomorphic Message Authenticator scheme (HMAC) that Catalano and Fiore proposed in [29] to perform an efficient verifiable linear combination arithmetic operation over encrypted data. In our scheme, we only consider about the verifiability over linear functions (so called degree-1 functions in [29]). The general procedure of a HMAC scheme consists of 4-tuple of algorithms working as follows:

$(ek, sk) \leftarrow \mathrm{KeyGen}(1^{\lambda})$: Let p be a random prime of λ bits, choose a K as a seed of pseudorandom function $\mathrm{F_K}: \{0,1\}^{*} \rightarrow Z_p$ and a random value $x \in Z_p$. Output secret key $\mathrm{sk} = (K, x)$, and evaluation key $\mathrm{ek} = p$, and let the message space M be Z_p.

$t \leftarrow \mathrm{Auth}(sk, \tau, m)$: To authenticate a message m with label τ, compute $r_{\tau} = F_K(\tau)$, then, set the authentication tag $t = (r_{\tau} - m)/x \bmod p$. Output t.

$(m', t') \leftarrow \mathrm{Eval}(f, M, T)$: The evaluation algorithm homomorphiclly evaluate the linear function f. M and T represent the message set used in the computation of f and the corresponding tag set respectively. When computing the linear computation on message set M, the same computation should be played over the corresponding tag set T too. For example, when computing $m' = m_1 + m_2, t' = t_1 + t_2$ should be played too. Then, out put $m' = f(M)$ and $t' = f(T')$.

$True/False \leftarrow \mathrm{Ver}(sk, ek, m', t', f, L)$: L is a set of labels corresponding to messages that participate in the computation and the evaluate function f. For each τ_{i} in L, compute $r_{\tau_{\mathrm{i}}} = F_K(\tau_{\mathrm{i}})$. Then compute $\rho = f\left(r_{\tau_1}, r_{\tau_2}, \ldots r_{\tau_{\mathrm{n}}}\right)$. Check if $\rho = (m' + t' \cdot x) \bmod p$. If the equation holds, output $True$, otherwise, output $False$.

Note that the original scheme of [29] requires an authenticated arithmetic circuit info (be also known as labeled program)[33] before verification. The arithmetic circuit info contains a set of labels and function descriptions. To be specific, the labels specify the input data that participate in the computation, and the function descriptions show how to calculate the input data. In our scheme, the function description is obviously obtainable because we only consider the linear functions, which means as the inputs of the function are determined, the function could be computed with all inputs added. The scalar multiplication could be converted and performed as addition, in which the priority problem of operators could be ignored. The real problem is the acquisition of input labels, which, in our scheme, is not available until a query request is responded. The inputs of the arithmetic circuit are decided by the SQL statement, but the verification could be invalid if both the labeled program and the computation result are provided by the CSP. We propose a label-searching scheme, which will be presented in the rest of the paper, to address this practical limitation.

4. System Description

In this section, we will firstly give a brief introduction of our outsourcing scheme with computation verifiability, including the basic system model and data flow. Then, we present some details of system implemention, including the transfer layer and the building of the arithmetic circuit info.

4.1 System Model

The main design objective of data-outsourcing system in edge-cloud devices is to minimize the storage and computation cost of IoT devices as well as keeping data privacy. The T-DB system proposed by Wang et. al. [5] is capable of doing basic data management over encrypted data. Hence our outsourcing database with computation verifiability scheme is built upon the T-DB architecture.

Fig. 3 shows the architecture and the data flow of the proposed model. The model contains 3 roles: the IoT devices, the gateway running a interlayer program and the edge-cloud service provider. The IoT devices represents the data users which outsource their data sets to maintain a minimized local storage and computation load for a long duration. With the help of edge-cloud service, one can require the data manage service only when needed. The gateway represents the roles with superior manage authority, namely, the data owners. Specifically, the gateway manager is an interlayer program with higher privilege. It is responsible for secret key management, requests/responds translation/transmission, and computation verifying. The cloud service provider represents a third-party proxy who provides data storage and computation service to the IoT devices. Many of such edge-cloud service providers constitute a Fog Computing environment.

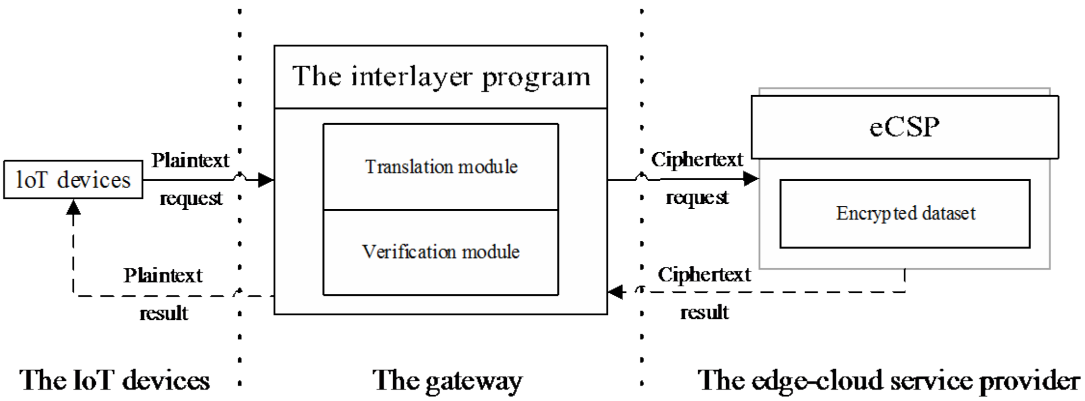


Fig. 3. The data flow of the proposed scheme

To initialize the system, The data owner first secretly generates and keeps the secret keys of OPEA, the secret key $sk = (K, x)$ and evaluate key p of HMAC. The raw data is encrypted into OPEA ciphertext, and for those data that need to be computed, choose a unique label τ_i to authenticate each OPEA ciphertext, and then generate authentication tags t_i, adding them into authentication fields in ciphertext records. After outsourcing the data, neither the plaintext nor the ciphertext data copy is preserved by the data owner, but only the secret key of encryption, labels used in authentication, and some structure information are stored in the interlayer program. Then, the eCSP receives and stores the encrypted data and some configuration information (such as some query-related UDFs [5]). Once the system is initiated, the data manipulation on encrypted data works as follows:

1. The data user first passes the access control policy to get access into the system, then submits a plaintext query request to the interlayer program.
2. The data owner then checks whether the request is legal (e.g. the request comes from a legal user). If it is, the interlayer program will translate the request into a corresponding ciphertext form and generates an arithmetic circuit info (if necessary), while the request will be rejected and dropped if it's an illegal one.
3. The data owner then submits the ciphertext request to cloud service provider. After the request is executed and responded, the interlayer program will first use the labels and arithmetic circuit info to verify the correctness of the execution results (if necessary). If the verification is failed, it means either the request was not being executed correctly or there was something wrong during the transmission. The result will be dropped before being decrypted. If the verification succeeds, the result will be decrypted and transmitted to the data user.

4.2 System Details

The interlayer program used in our scheme has the following functions:

1. Secrete key management of OPEA and HMAC;
2. Encoding and decoding for requests from IoT devices and responds from eCSP;

3. Verification for linear computation results.

For a typical database table, the OPEA needs to be applied in each field separately, which means the keys of OPEA are generated independently. It is also necessary for each field to generate its own secret key and evaluation key used in the HMAC, so that the authenticators could remain valid. In a word, each field in a database (namely, data from a column) uses its own OPEA secret key and HMAC key pairs to encrypt/authenticate the data, without sharing with other field.

The OPEA encryption-based shcemes [3,13] have implemented the basic functions of data manipulation in both SQL-based database and Fomula-based excel scenarios. To modify such schemes into a computation-verifiable version, just duplicate the linear function related operations on the encrypted data fields on the corresponding HMAC tag fields. For example, if a query asks for the result of “SELECT SUM(Att) FROM table”, the verifiable version of such query should be “SELECTSUM(Att), SUM(Att_t)FROM table”, in which the Att represents for a certain field and Att_t represents for the corresponding tag field.

The label is used to uniquely specify the input data of a function f, which means it is not re-usable over different records (rows), otherwise the eCSP may forge the authenticate tags. Furthermore, the authenticate tags generated by a label leaks no information a the label itself, because such tag is generated by a pseudo-random function, which is theoretically irreversible. So a natural thought is making the labels carry some information of the plaintext and be cached in the interlayer so that when a query submitted, the interlayer program can build the arithmetic circuit info locally. As a result, the interlayer program need to cache the labels, and the description of the arithmetic circuits should not be beyond the labels’ containment. There are two exaples on how to choose and use the labels in next subsection.

The verification process needs both the computation results from the eCSP and the arithmetic circuit information to proceed. So the interlayer program should be capable of building those information on its own, as a result, it needs to cache the certain labels during the uploading procedure.

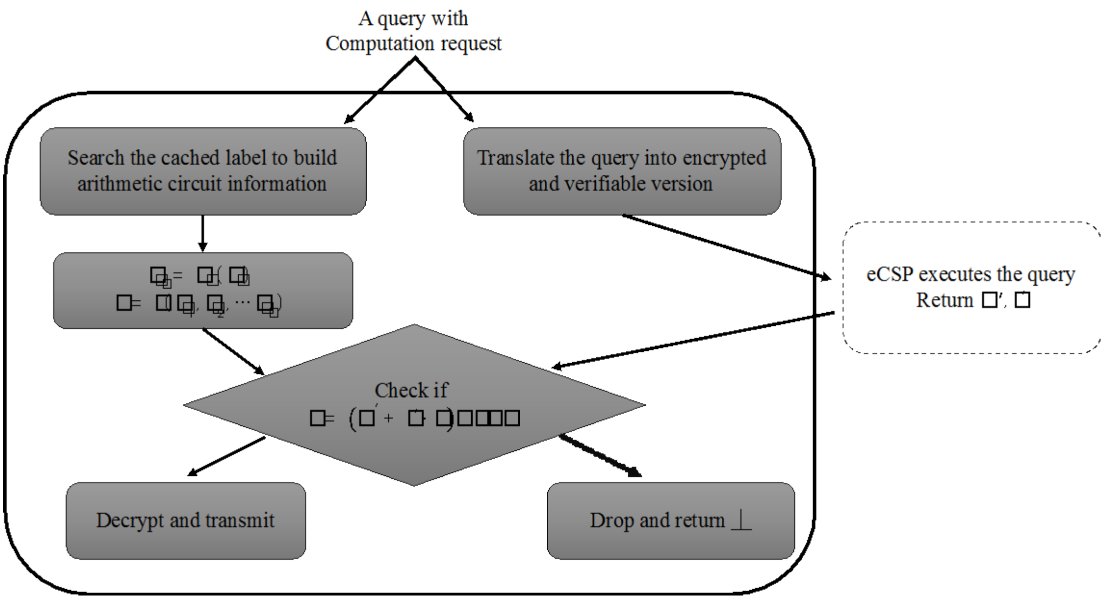

A query with
Computation request
Search the cached label to build arithmetic circuit information
Translate the query into encrypted and verifiable version
eCSP executes the query
Return □', □
Check if
Decrypt and transmit
Drop and return ⊥

Fig. 4. The detailed procedure of the interlayer program

Fig. 4 shows how a query with computation is dealt by the interlayer program. A query would be tranlated into encrypted and verifiable version by the translation module, then be posted to the eCSP for excution. In the meantime, the verification module would search its local cache to build arithmetic circuit information, in other word, find the labels (τ_i) corresponding to the participants of the computation, then caculate ρ for preparation of verification. After received the response from eCSP, the interlayer program would check if the result is valid. Then decide whether to decrypt the results and transmit them or drop them. It is also shown in Fig. 4 that the building procedure of arithmetic circuit info and the query translation as well as the query excution are parallelizable.

Note that in a certain field, the label of each record should not be reused, because the reuse of the same label may lead to the leakage of the secrete key in HMAC [29]. A brief proof is shown below:

Theorem 1*:* Reusing a label under the same secret key leads to leakage of secret key.

Proof: Let τ be a reused label over two different message m_1 and m_2 under the same secret key $sk = (K, x)$ and evaluation key $ek = p$, so the authenticators of the given messages are:

$$t_1 = \frac{r_1 - m_1}{x} \bmod p = \frac{F_K(\tau) - m_1}{x} \bmod p$$

$$t_2 = \frac{r_2 - m_2}{x} \bmod p = \frac{F_K(\tau) - m_2}{x} \bmod p$$

Since the t_1, t_2 and m_1, m_2 are all public (or are exposed to the eCSP), aparently there is

$$t_1 - t_2 = \frac{m_2 - m_1}{x} \bmod p$$

So that the secret key x is leaked. End of the proof.

Even though it is dangerous to reuse labels in one field, however, reusing a label of a certain record in different fields is permitted because the secret keys of HMAC were generated separately, and the PRF outputs were different from each other under different seed K, even for the same label. Theoretically it is impossible to obtain the inputs from the outputs of a PRF. So each one of the record (one row) could be authenticated by the same label, with different secret keys.

4.3 Possible Optimization

The data privacy-preserving and computation verification bring extra storage load to both the eCSP and the interlayer program. To reduce the storage load in all ways, we also present some possible optimization methods.

Partial Authentication. It is clear that not all the fields need verification. In general, all the fields could be divided into three categories. *The non-sensitive field* contains anonymous information like the record *id* and *timestamps*. *The query-only field* contains the data that would only be used as query condition without any computation-based use. *The computation field* contains data that would apply to the fully function of OPEA encryption, which contains linear computation and other basic query functions. Obviously, the HMAC is capable of not only the computation verification, but also the single record verification (treated as a linear function with only one input). The HMAC could be used only in *the query-only field* and *the computation field*. As for *the non-sensitive field*, the HMAC field is not necessary.

Label Cache. As mentioned before, the label used to uniquely specify may contain some information for query use. One example is the real-time data. There are some IoT devices generates and uses real-time data, e.g. the heart rate monitor continously generates the heart rate data, then, at the end of the day, caculates the average heart rate of a patient in a day. In such situation, the labels of HMAC could be the timestamp of each heart rate record since the timestamp is unique. If the interlayer program stores every timestamp of the uploading record, the storage of label cache would grow linearly with the increasing the number of records, which provides arbitrarily verification ability on any time-based query.

Another example is the range-query based data, e.g. a hosipital need to constantly caculate some vital signs of patients from a certain age group. The query conditions are all based on age, so age could be a label to specify a patient record. But, in order to differ the same-aged patients, we also need to attach a serial number to the age to specify the record uniquely, like the form of “18-001”, means the number “001” 18-year-old patient. The storage is also linearly growth with the increasing of records amount, however, with some tricks like only store the age and count of patients of that age, the storage load could be reduced to constant level.

5. Evaluation

In this section, we comprehensively evaluate the scheme proposed in this paper. We firstly give security analysis to the scheme. Then we present both theoratical and emprical evaluation to the efficiency of the scheme in space and time consumption. We would also make some comparision between our scheme and some presented schemes.

5.1 Security Analysis

In this paper, our scheme achieves that the eCSP is completely kept away from the plaintext data. With the help of interlayer program, all requests from data user (e.g. some IoT devices) can work perfectly without changing the way it requests data. Here we suppose the data owner and the interlayer program remain safe. Since the eCSP stores both the encrypted data and authenticate tags which makes it a major threat of security breach, we firstly give the security description of the building blocks used in our scheme. Then we consider 3 kinds of exceptional situations that eCSP might be involved and analyze how the scheme works under such scenarios.

Security of OPEA. It has been proven in [5] that OPEA is secure under the IND-AOCPA game (IND-AOCPA is indistinguishability under an Additive Ordered Chosen-Plaintext Attack), which means OPEA is capable of resisting ciphertext-only attack, statistical attack and chosen-plaintext attack.

Security of HMAC. The security of a message authenticator scheme could be considered from 2 perspectives. On one hand, without reusing the authenticate label, each message is hidden by a random result generated by the PRF (as shown in the $Auth$ algorithm), which could be regarded as an one-way function, so it is theoratically impossible to extract information from the authentication tags. On the other hand, as long as the sk of HMAC remain confidential, it is also impossible to forge an authentication tag, even the message and label are given. This is due to the collision-resistant property of PRF ($F_K(\tau)$). In a word, HMAC could guarantee the integrity of both messages and homomorphic computations, as well as preserving confidentiality and collision-resistant. In our scheme, every field of one record share a same label τ, and each field uses its own $sk = (K, x)$, which means all values of PRF ($F_K(\tau)$) are without repeat. For formal proofs about HMAC please refer to [29].

Now we consider the three possible exceptional situations that eCSP might be.

Honest but curious. In such situation the CSP will honestly execute the requests that users submit, but it also tries to learn about some privacy information from the outsourcing data. This is also called semi-honest, which is the most common situation considered in data outsourcing. The OPEA algorithm used in our scheme has been proven secure under the IND-AOCPA game, which means the curious CSP cannot obtain any additional sensitive information except that inferred by the order and additivity of ciphertext. As for the HMAC part, the authentication tags are all generated from OPEA ciphertexts, and blinded by PRF, which means they contain no more information than ciphertexts (or to say basically no useful information).

Not honest. In such situation the CSP will not perform the correct requests submitted by the users. The CSP may not respond the users' requests to perform a denial of service, but this is not the main issue in practice since the user can soon notice the problem. A more worrisome issue is that the CSP continuing provides wrong responds that the users could barely notice. Considering there is no efficient method to address computation verifying and search verifying, we cannot detect all wrong responds in practice but the incorrect linear computation could be detected, which is a great help to catch an exception of CSP. As described before, every record is authenticated by a unique label. Forging the authentication tags is impossible since the security of HMAC can be reduced to a single pseudo-random function. As long as the sk remains secret, the tags leak no information about anything.

Compromised. In this situation, the CSP is compromised and the attacker obtains the entire database. Similar to the first situation, the attacker learns no more than the order and additivity of ciphertext. And since the authentication tags are generated based on the ciphertext, they leak no information about the plaintext neither.

Furthermore, the verification could be performed not only on the computation results, but also any other query based on the cached labels, which could help the interlayer program to check if the data responded by the eCSP were tampered during the transfer process.

5.2 Efficiency

The efficiency analysis will be presented in 2 aspects, the time and space efficiency. We use a wireless sensor dataset from the UCI Machine Learning Repository [34] which contains a HeartRate-monitor, collecting real-time heart rate data with 9Hz sampling frequency [35]. The heart rate data were encrypted and authenticated with the timestamp of the record. All of the experiments were performed on a Intel-Core i7, 8GB memory platform.

SPACE. Invoking encryption and verification brings extra storage load to eCSP. Specifically, for those fields needs addition computing, one plaintext field will be encrypted/authenticated into 4 fields (2 for OPEA and OPEA extend ciphertext [5,6], 2 for authenticators of OPEA and OPEA extend ciphertext separately), and for those do not need computing, one plaintext field will be encrypted into just 1 OPEA ciphertext field. For the privacy protection purpose, such price is worth to take and acceptable.

The encryption and verification also bring extra storage to the interlayer program. Specifically, the interlayer is responsible for key management of both OPEA algorithm and HMAC. Moreover, label cache also requires extra storage to preserve information about labels in the interlayer program. The key management storage is constant level while the unique label storage will grow linearly as the updated data increases. But, as we mentioned before, the edge-cloud in fog computing provides only temporary storage in most cases, the long-term and mass data storage would be the job of upper cloud. So the storage of label information is also capped. And with the optimization provided, the storage load of interlayer program could be limited in an acceptable level.

TIME. In the time evaluation, we used SUM() function in SQL as a typical utility of linear computation to evaluate the effciency of our scheme. In other words, we played SUM() queries over different time window (refering to different amount of computation participants) and evaluate the effect of verifiability. So we run the following SQL statements respectively:

Query 1. SELECT SUM(Heart_rate) FROM Encrypted_data WHERE Time_stamp BETWEEN lower AND upper;

Query 2. SELECT SUM(Heart_rate), SUM(Heart_rate_tag) FROM Authenticated_data WHERE Time_stamp BETWEEN lower AND upper;

Query 3. SELECT Time_stamp FROM Labels WHERE Time_stamp BETWEEN lower AND upper.

Note that we do not consider the network communication latency in the experiment.

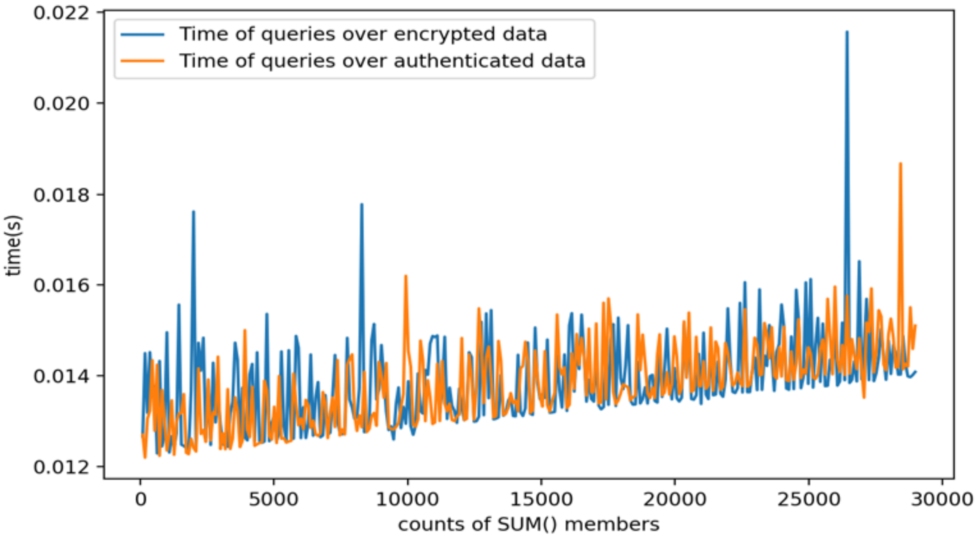


Fig. 5. Time spent on querying results

Fig. 5 shows the effieiency of computing the sum of heart rates data (Query 1) and computing the sum of both heart rates data and authenticated tags (Query 2) over certain amount of records. Due to the inner optimization of database system (mysql 8.0 here), the SUM() computation over an extra attribute did not increase the time cost significantly, meaning that the process of the eCSP would not bring much more extra time load compared with the system in [5].

The building procedure of arithmetic circuit info includes searching the labels of target inputs to compute the pseudo-random function and calculating the labeled program with the pseudo-random values. To build the arithmetic circuit info, the interlayer program will firstly find the labels specifying the input data and compute every PRF value respectively, then substitute the PRF values into the function f and calculate. With the cached labels, these work could be done locally. Apparently, the time complexity of such procedure is $O(|f|)$, where $|f|$ denotes the size of input dataset of function f. As **Fig. 6** showed, the trend of time spent on querying cached labels (Query 3) is little steeper than that on authenticated data (Query 2) because it returns more records. It is also shown that, with the increasing amount of returned labels, the time spent on the preparing of verification (computing PRF values) grows linearly, which is mainly cost by the computation of PRF. **Table 1** demonstrates the detailed time cost in the verification procedure under different parameter settings, which also shows that the computation of PRF values is the major cost. To reduce such load, we may pre-compute the PRF values of the cached labels.

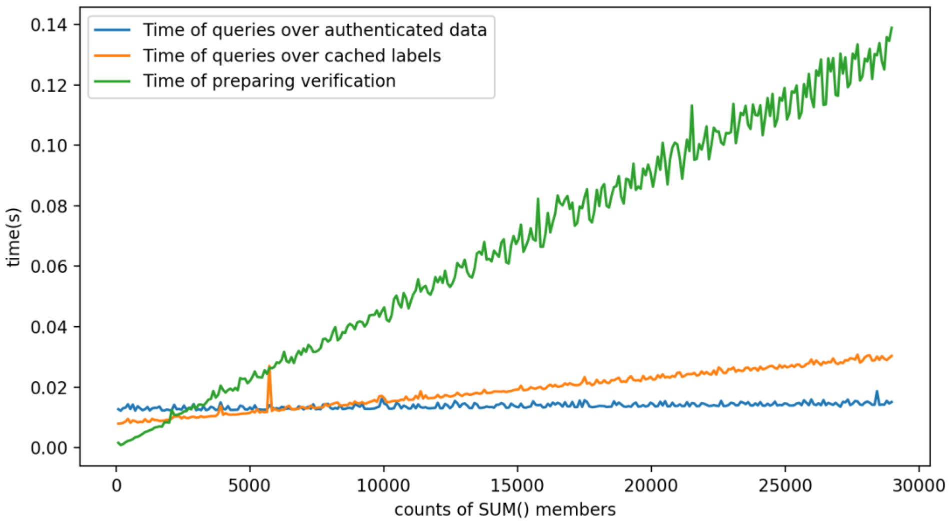


Fig. 6. Time spent on evaluation

Table 1. Time efficiency of the proposed scheme

	Time(s)					
HMAC parameter λ	80-bits		128-bits		160-bits	
Number of inputs	100	10000	100	10000	100	10000
PRF computing	0.00110	0.112	0.00112	0.173	0.00122	0.173
Computing ρ over PRF values	0.000002	0.000187	0.000003	0.000120	0.000007	0.000145

As for the verification procedure, since the eCSP will finally respond a computation result m' and a tag t' to the interlayer program, the only thing verification module has to do is just check if $\rho = (m' + x * t') \bmod p$, which is a constant time. And the experiments also show that the verification can be finished within 0.000001s. Experiments also show that invoking the authentication tags in the computing of linear functions will bring under 10% extra time load than the original scheme. Moreover, the time cost of the building procedure of arithmetic circuit info is far less than the time cost of executing query and network latency. So it is fairly to say that the verification time cost is the only extra cost in our scheme, which is within 0.000001s.

5.3 Comparison

We compare our scheme with 2 previously presented data outsourcing schemes. One is CryptDB [21], a well-known encrypted database system, the other is VDB (Verifiable Database) presented in [36], which is built from vector commitment, a widely-used cryptographic primitive to construct VDB. The basic properties comparison among these schemes is shown in **Table 2**.

Table 2. Comparison with other schemes

	CryptDB[21]	VDB[36]	Our scheme
Database Type	Encrypted database	Common database*	Encrypted database
Functionality	Almost any	Any query*	Almost any
DBaaS Framework?	No	Yes	Yes
Security Basis	Onion Encryption	-	OPEA
Verification Basis	-	Vector Commitment and other premitives	HMAC
Public Verification	-	Yes	No
Verification Area	-	Data integrity	Data intergrity and computation
Homomorphic Verification	-	No	Linear computation

The CryptDB uses a so called “onion encrytion” to achieve multiple functions over the encrypted data. For each field (namely, a column) of a database, encryption schemes with different properties are applied on the data layer upon layer. When a specific property is needed to perform certain query, the data has to be firstly sent to the third proxy for decryping to certain layer, which violate the DBaaS framework, making it difficult to fitting in current applications and it also brings massive time overhead compared with our scheme as mentioned in [5]. Besides, the CryptDB does not support any kind of verification, the integrity of the database could not be protected.

Common VDB schemes like [36] was designed mainly to protect the integrity of database, the vector commitment could only be used to check if the data were tampered. The VDB does not protect the confidentiality of data, and the verification is also invalid when the data user requests for an aggregated result. Our scheme somehow make up some defects of VDB. On one hand the database itself is protected by a functional encryption which supports range query as well as linear computation. On the other the HMAC is capable of verifying the function result with almost no change of current system efficiently. Even the functionality and verification ability of our scheme is perfectly complete, it could be a great choice in the scenario given above.

6. Conclusion

In this paper, we proposed a computation verifiable data outsourcing scheme in the edge-cloud under fog computing. Our scheme is based on the proxy re-encryption framework, the dataset is outsourced in the form of ciphertext and all data requests need to be transmitted and translated by the interlayer program. The interlayer program is also able to verify the result of linear functions, so that the IoT devices will obtain the correct responds. Note that the HMAC scheme proposed in [29] has the ability to verify degree-n polynomial function, which means the verification can run over the homomorphic encryption. But it still has a long way to go in practicability. Our scheme focuses on the efficient manipulation over encrypted data under DBaaS framework. We test our scheme over proposed scheme [6] and [5], experiments show that the verification brings neglectable time cost and fits perfectly in the given fog computing scenario.

References

[1] F. Bonomi, R. A. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the internet of things," in *Proc. of the first edition of the MCC workshop on Mobile cloud computing, MCC@SIGCOMM 2012, Helsinki, Finland*, pp. 13–16, 2012. Article (CrossRef Link)

[2] D. Slamanig and C. Stingl, "Privacy Aspects of eHealth," in *Proc. of the The Third International Conference on Availability, Reliability and Security, ARES 2008, March 4-7, 2008, Technical University of Catalonia, Barcelona, Spain*, pp. 1226–1233, 2008. Article (CrossRef Link)

[3] K. Islam, W. Shen, and X. Wang, "Security and privacy considerations for Wireless Sensor Networks in smart home environments," in *Proc. of IEEE 16th International Conference on Computer Supported Cooperative Work in Design, CSCWD 2012, Wuhan, China*, pp. 626–633, 2012. Article (CrossRef Link)

[4] H. Hacigümüs, S. Mehrotra, and B. R. Iyer, "Providing Database as a Service," in *Proc. of the 18th International Conference on Data Engineering, San Jose, CA, USA*, pp. 29–38, 2002. Article (CrossRef Link)

[5] X. Wang, Q. Wu, and Y. Zhang, "T-DB: Toward Fully Functional Transparent Encrypted Databases in DBaaS Framework," *CoRR, vol. abs/1708.08191*, 2017. Article (CrossRef Link)

[6] Y.-N. Li, Q. Wu, W. Tang, B. Qin, Q. Wang, and M. Miao, "Outsourcing Encrypted Excel Files," in *Proc. of Information Security Practice and Experience - 13th International Conference, ISPEC 2017, Melbourne, VIC, Australia*, pp. 506–524, 2017. Article (CrossRef Link)

[7] D. R. Kuhn, E. J. Coyne, and T. R. Weil, "Adding Attributes to Role-Based Access Control," *IEEE Computer*, vol. 43, no. 6, pp. 79–81, 2010. Article (CrossRef Link)

[8] L. Teo, G.-J. Ahn, and Y. Zheng, "Dynamic and risk-aware network access management," in *Proc. of 8th ACM Symposium on Access Control Models and Technologies, SACMAT 2003, Villa Gallia, Como, Italy*, pp. 217–230, 2003. Article (CrossRef Link)

[9] J. Li, X. Chen, J. Li, C. Jia, J. Ma, and W. Lou, "Fine-Grained Access Control System Based on Outsourced Attribute-Based Encryption," in *Proc. of Computer Security - ESORICS 2013 - 18th European Symposium on Research in Computer Security, Egham, UK*, pp. 592–609, 2013. Article (CrossRef Link)

[10] J. K. Liu, M. H. Au, X. Huang, R. Lu, and J. Li, "Fine-Grained Two-Factor Access Control for Web-Based Cloud Computing Services," *IEEE Trans. Information Forensics and Security*, vol. 11, no. 3, pp. 484–497, 2016. Article (CrossRef Link)

[11] J. Li *et al.*, "Fine-Grained Data Access Control Systems with User Accountability in Cloud Computing," in *Proc. of Cloud Computing, Second International Conference, CloudCom 2010, Indianapolis, Indiana, USA*, pp. 89–96, 2010. Article (CrossRef Link)

[12] C. Gentry, "Fully homomorphic encryption using ideal lattices," in *Proc. of the 41st Annual ACM Symposium on Theory of Computing, STOC 2009, Bethesda, MD, USA*, pp. 169–178, 2009. Article (CrossRef Link)

[13] R. Agrawal, J. Kiernan, R. Srikant, and Y. Xu, "Order-Preserving Encryption for Numeric Data," in *Proc. of the ACM SIGMOD International Conference on Management of Data, Paris, France*, pp. 563–574, 2004. Article (CrossRef Link)

[14] A. Boldyreva, N. Chenette, Y. Lee, and A. O'Neill, "Order-Preserving Symmetric Encryption," in *Proc. of Advances in Cryptology - EUROCRYPT 2009, 28th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cologne, Germany*pp. 224–241, 2009. Article (CrossRef Link)

[15] R. A. Popa, F. H. Li, and N. Zeldovich, "An Ideal-Security Protocol for Order-Preserving Encoding," in *Proc. of 2013 IEEE Symposium on Security and Privacy, SP 2013, Berkeley, CA, USA*, pp. 463–477, 2013. Article (CrossRef Link)

[16] D. Boneh, G. D. Crescenzo, R. Ostrovsky, and G. Persiano, "Public Key Encryption with Keyword Search," in *Proc. of Advances in Cryptology - EUROCRYPT 2004, International Conference on the Theory and Applications of Cryptographic Techniques, Interlaken, Switzerland*, pp. 506–522, 2004. Article (CrossRef Link)

[17] N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, "Privacy-Preserving Multi-Keyword Ranked Search over Encrypted Cloud Data," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 1, pp. 222–233, 2014. Article (CrossRef Link)

[18] W. Sun *et al.*, "Privacy-preserving multi-keyword text search in the cloud supporting similarity-based ranking," in *Proc. of 8th ACM Symposium on Information, Computer and Communications Security, ASIA CCS '13, Hangzhou, China - May 08 - 10*, pp. 71–82, 2013. Article (CrossRef Link)

[19] J. Li, Q. Wang, C. Wang, N. Cao, K. Ren, and W. Lou, "Fuzzy Keyword Search over Encrypted Data in Cloud Computing," in *Proc. of INFOCOM 2010. 29th IEEE International Conference on Computer Communications, Joint Conference of the IEEE Computer and Communications Societies, San Diego, CA, USA*, pp. 441–445, 2010. Article (CrossRef Link)

[20] B. Wang, S. Yu, W. Lou, and Y. T. Hou, "Privacy-preserving multi-keyword fuzzy search over encrypted data in the cloud," in *Proc. of 2014 IEEE Conference on Computer Communications, INFOCOM 2014, Toronto, Canada*, pp. 2112–2120, 2014. Article (CrossRef Link)

[21] R. A. Popa, C. M. S. Redfield, N. Zeldovich, and H. Balakrishnan, "CryptDB: protecting confidentiality with encrypted query processing," in *Proc. of the 23rd ACM Symposium on Operating Systems Principles 2011, SOSP 2011, Cascais, Portugal*, pp. 85–100, 2011. Article (CrossRef Link)

[22] Y. Deswarte, J.-J. Quisquater, and A. Saïdane, "Remote Integrity Checking - How to Trust Files Stored on Untrusted Servers," in *Proc. of Integrity and Internal Control in Information Systems VI - IFIP TC11/WG11.5 Sixth Working Conference on Integrity and Internal Control in Information Systems (IICIS) 13-14 November 2003, Lausanne, Switzerland*, pp. 1–11, 2003. Article (CrossRef Link)

[23] M. N. Krohn, M. J. Freedman, and D. Mazières, "On-the-Fly Verification of Rateless Erasure Codes for Efficient Content Distribution," in *Proc. of 2004 IEEE Symposium on Security and Privacy (S&P 2004), 9-12 May 2004, Berkeley, CA, USA*, pp. 226–240, 2004. Article (CrossRef Link)

[24] G. Ateniese et al., "Provable data possession at untrusted stores," in *Proc. of the 2007 ACM Conference on Computer and Communications Security, CCS 2007, Alexandria, Virginia, USA*, pp. 598–609, 2007. Article (CrossRef Link)

[25] A. Swaminathan *et al.*, "Confidentiality-preserving rank-ordered search," in *Proc. of the 2007 ACM Workshop On Storage Security And Survivability, StorageSS 2007, Alexandria, VA, USA*, pp. 7–12, 2007. Article (CrossRef Link)

[26] A. Juels and B. S. K. Jr, "Pors: proofs of retrievability for large files," in *Proc. of the 2007 ACM Conference on Computer and Communications Security, CCS 2007, Alexandria, Virginia, USA*, pp. 584–597, 2007. Article (CrossRef Link)

[27] H. Shacham and B. Waters, "Compact Proofs of Retrievability," in *Proc. of Advances in Cryptology - ASIACRYPT 2008, 14th International Conference on the Theory and Application of Cryptology and Information Security, Melbourne, Australia*, pp. 90–107, 2008. Article (CrossRef Link)

[28] R. Gennaro and D. Wichs, "Fully Homomorphic Message Authenticators," in *Proc. of Advances in Cryptology - ASIACRYPT 2013 - 19th International Conference on the Theory and Application of Cryptology and Information Security, Bengaluru, India*, pp. 301–320, 2013. Article (CrossRef Link)

[29] D. Catalano and D. Fiore, "Practical Homomorphic MACs for Arithmetic Circuits," in *Proc. of Advances in Cryptology - EUROCRYPT 2013, 32nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Athens, Greece*, pp. 336–352, 2013. Article (CrossRef Link)

[30] D. Catalano, D. Fiore, and B. Warinschi, “Homomorphic Signatures with Efficient Verification for Polynomial Functions,” in *Proc. of Advances in Cryptology - CRYPTO 2014 - 34th Annual Cryptology Conference*, *Santa Barbara*, *CA*, *USA*, pp. 371–389, 2014. Article (CrossRef Link)

[31] M. Backes, D. Fiore, and R. M. Reischuk, “Verifiable delegation of computation on outsourced data,” in *Proc. of 2013 ACM SIGSAC Conference on Computer and Communications Security*, *CCS’13*, *Berlin*, *Germany*, pp. 863–874, 2013. Article (CrossRef Link)

[32] D. Fiore, R. Gennaro, and V. Pastro, “Efficiently Verifiable Computation on Encrypted Data,” in *Proc. of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, *Scottsdale*, *AZ*, *USA*, pp. 844–855, 2014. Article (CrossRef Link)

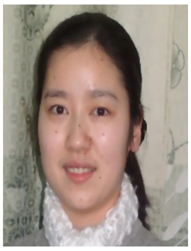
[33] A. Shpilka and A. Yehudayoff, “Arithmetic Circuits: A survey of recent results and open questions,” *Foundations and Trends in Theoretical Computer Science*, vol. 5, no. 3–4, pp. 207–388, 2010. Article (CrossRef Link)

[34] D. Dua and C. Graff, *UCI Machine Learning Repository*, University of California, Irvine, School of Information and Computer Sciences, 2017. Article (CrossRef Link)

[35] A. Reiss and D. Stricker, “Introducing a New Benchmarked Dataset for Activity Monitoring,” in *Proc. of 2012 16th International Symposium on Wearable Computers*, *Newcastle*, *United Kingdom*, pp. 108–109, 2012. Article (CrossRef Link)

[36] D. Catalano and D. Fiore, “Vector commitments and their applications,” in *Proc. of International Workshop on Public Key Cryptography*, pp. 55–72, 2013. Article (CrossRef Link)

Wenyi Tang received his B.S. degree in School of Information, Renmin University of China, where he is currentluy pursuing a master's degree of information security. His research interests include applied cryptography.



Bo Qin received her Ph.D. degree in Cryptography from Xidian University in 2008 in China. She is currently an associate professor in School of Information, Renmin University of China. Her research interests include applied cryptography, data security, and privacy protection. She has been a holder/co-holder of 15 China/Spain funded projects, and she has authored over 100 publications.

Yanan Li received her B.S. degree in Shandong University. After that she studied in Beihang University for her Master’s degree. She is currently pursuing the Ph.D degree in Georgia Institute of Technology, USA. Her research interests include applied cryptography and mobile security.

Qianhong Wu received his Ph.D. in Cryptography from Xidian University in 2004. He is currently a professor with Beihang University in China. His research interests include cryptography, information security and privacy, VANET security and cloud computing security. He has been a holder/co-holder of 22 China/Australia/Spain funded projects. He has authored 37 patents and over 130 publications. He has served as associate/guest editor in several international ISI journals and in the program committee of dozens of international conferences. He is a member of IACR, IEEE and ACM.