IJNS
International Journal of Network Security

ISSN 1816-353X (Print)
ISSN 1816-3548 (Online)

Vol. 22, No. 1 (Jan. 2020)

INTERNATIONAL JOURNAL OF NETWORK SECURITY

PUBLISHING OFFICE

Min-Shiang Hwang
Department of Computer Science & Information Engineering, Asia University, Taichung 41354, Taiwan, R.O.C.
Email: mshwang@asia.edu.tw

International Journal of Network Security is published both in traditional paper form (ISSN 1816-353X) and in Internet (ISSN 1816-3548) at http://ijns.jalaxy.com.tw

PUBLISHER: Candy C. H. Lin

Volume: 22, No: 1 (January 1, 2020)

International Journal of Network Security

Packet Watermarking With ECG Biological Feature

Kuo-Kun Tseng[1], Xialong He[1], Xiaoxiao An[1], Chin-Chen Chang[2], Chao Wang[1], Xiangmin Guo[3]
(Corresponding author: Chin-Chen Chang)

School of Computer Science and Technology, Harbin Institute of Technology (Shenzhen), China[1]
Department of Information Engineering and Computer Science, Feng Chia University, Taiwan[2]
School of Architecture, Harbin Institute of Technology (Shenzhen), China[3]
(Email: alan3c@gmail.com)
(Received Feb. 7, 2018; Revised and Accepted Oct. 18, 2018; First Online Sept. 21, 2019)

Abstract

In this paper we proposed a novel security authentication mechanism, which is a secure authentication mechanism for network transmission with an ECG biological feature. The theory of this authentication mechanism can be applied within various network identification and authentication systems. In addition, the algorithm needs to strengthen the safety and performance for watermarking. In addition, experimentation on the packet watermarking of ECG is conducted. The criteria has two parameters, one is the SNR (signal-to-noise ratio) and the other is BER, to evaluate the overall performance. Not only that, we have also considered noise attack. According to our obtained results, our algorithm has proven to be robust, and it thus worth considering in the application.

Keywords: Authentication; Biological Feature; ECG Signal; Packet Watermarking; Secure Transmission

1 Introduction

The emergence of communication networks has made enormous changes to people's lives. The network has enabled the communication between people to become more concise, has greatly improved the quality of human life, and sped up the process of social development. But it also carries a variety of risk [14]. Nowadays, all over the world, all aspects of the military, economy, society, culture and so on, are increasingly dependent on computer networks. The dependence on computer networks of human society has achieved an unprecedented record [4]. However, at the same time, the scale of attacks and threats to computer networks are also unprecedented. Many kinds of attack continually emerge, which can sometimes quickly cause large-scale network debacles. For today's society, the impact is definitely not less than a natural disaster, and may even be worse than. Therefore the protection of network security is imperative. The threats to networks may be from computer virus, worm and hacker attacks. These security problems are commonly introduced due to negligence and careless management and cause a large adverse impact. For example, hosting on an exposed environment, supervision of staff not being strict and so on, also gives attackers attacking space.

Therefore a more powerful mechanism to identify authorized data or users is essential. In this paper a new kind of the network authentication technology with transmission watermarking is proposed. We hope that the use of the personal ECG characteristic to assist with network transmission, and then providing a feature sequence as a watermark will protect the transmission process. The watermark operation is on the transmitting end, and a de-watermark operation is at the receiving end. Through the negotiation of the two parties we can confirm the security of the network transmission. Currently, we propose a security authentication mechanism based on the packet size with the ECG feature to achieve a transmission security authentication mechanism. In this research we also examine the feasibility and security of our proposed approach.

The structure of the paper is as follows: In the second section the applications of this mechanism in real life are firstly proposed. Of course this is only a hypothesis; and the application of secure transmission is very broad and not just limited to this hypothesis.

In the third section, we introduce relevant studies, especially highlighting research on watermarks. The proposed algorithm belongs to a new type of secure transmission.

In the fourth part we focus on the relevant content, as well as the algorithm, including the network transmission principle, ECG data acquisition methods and sources, and the principle of digital watermarking. Of course, the most important part is the changes that apply to digital watermarking in this architecture, and the algorithm used.

The fifth part undertakes the key evaluation of the watermark. It not only implements our methods, but also those of related research according to our ideas. As can

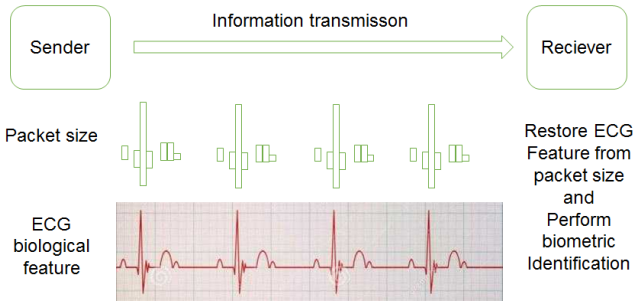


Figure 1: Application concept

be seen by comparing the data, the method has a certain advantage. In this method, the two assessment parameters of SNR and BER are mainly used as the evaluation criteria.

The final section of this paper provides a summary of the entire experiment, and comments on the literature review.

2 Application

As shown in Figure 1, we present a scheme of the proposed mechanism. It can work together with traditional biometric identification systems, such as fingerprint recognition or face recognition, distance certification, which can be applied to this authentication mechanism to enhance the safety performance of the authentication systems. Firstly, at the sender, is the extraction of the ECG, fingerprints or facial image as the biological features. Before transmission, the packet sizes of the network transmission are formed by biological features, such as the ECG waveform. In this, the series of packet size values has been formed as a sequence of waveforms, and then the data transmitted to the receiving end. After which the recipients receive these data, and decode the data packets. The correct sequence is obtained and the value of packet size extracted, and then compared with the ECG data to ensure that the transmission process is safe. Sequentially the extracted fingerprint or face images can also be converted to the related feature for identification and authentication.

Of course, this is just one kind of application of our proposed technology. The authentication mechanism can also be applied to many other applications, and requires further discussion in other research.

3 Background and Realted Work

3.1 Related Research

Articles on digital watermarking have been published continuously since 1994. Over time, the number of articles has presented a rapid increase, and several highly influential international conferences (such as IEEE ICIP, IEEE ICASSP, ACM Multimedia, *etc.*) as well as some international authoritative journals (such as the Proceedings of IEEE, Signal Processing, IEEE Journal of Selected Areas on Communication, Communications of ACM). "A digital watermark" [25] published by Van Schyndel for the ICIP'94 conference is the first article on digital watermarking published at major conferences [19]. May 30–June 6, 1996 saw the assembly of the first international symposium on information hiding (IHW) [22]. SPIE and IEEE International Conferences also featured related topics.

In the United States, a number of research institutions and enterprises represented by the MIT Media Lab have already applied for patents on digital watermarking. Digital watermarking has been supported or research conducted by government departments, universities and well-known enterprises, including the United States Finance Committee, United States Copyright Working Group, United States Air Force Institute, United States Army Research Laboratory, German National Information Technology Research Center, Japan NTT Information and Communications System Research Center, MIT, Illinois University, Minnesota University, and Cambridge University, Switzerland Lausanne Federal Institute of Technology, Vigo University, IBM Thomas J. Watson Research Center, Microsoft Research Cambridge, Lucent Technologies, Bay Networks, CA, Sony, NEC Research Institute and the Philips [25].

Chinese academic research on digital watermarking technology is not lagging behind compared to other countries. A number of famous scientific research institutions are already devoted to research in this area. In order to promote the research and application of digital watermark technology and other information hiding, by the end of 1999 several experts in the field of information security and related applied research units jointly held the first symposium on information hiding [22]. In early 2000, the national "863" intelligent machine expert group and the CAS Institute of Automation Pattern Recognition State Key Laboratory organised a symposium on digital watermarking, and reported the results of their own research.

Compared with the level of the rest of the world, the relevant Chinese academic field is not far off, and contains unique research ideas. So far, from the view of research, digital watermarking mainly relates to image, video, audio, text, and 3D grid data watermarking and so on, among which most of research and papers on watermarking focus on images. The reason for this is that the image is the most basic of multimedia data, and the development of the Internet provides large direct applications for image watermarking [13].

In addition, video watermarking has also attracted some researchers. Because video can be seen as a continuous image sequence, in a sense this is very similar to the principle of image watermarking, and many image watermarking research results can be directly applied to video watermarking. But there is an important difference between the two with respect to the magnitude of signal processing. In particular, the problem of real-time is considered in the study of video watermarking.

The research in this paper is similar to audio water-

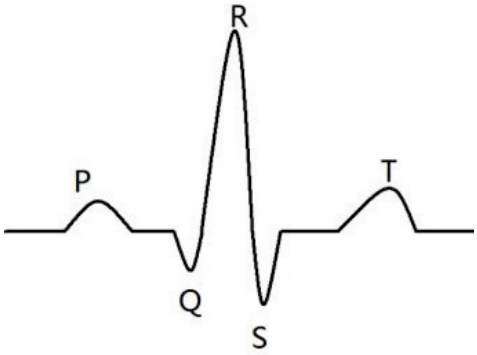


Figure 2: ECG schematic diagram

marking, because there are many similarities between ECG and audio. In this respect, the number of related studies at home and abroad is not extensive, but is in a gradually improving state. Simple watermarking technology has some shortcomings. Such as the watermark cannot be applied to images or signals with high accuracy, because watermarking causes some damage to the receptor and cannot fully recover during the restoration process.

The current watermarking applications have certain limitations, mostly used for copyright protection [7], adding fingerprinting [2], anti-tampering [9] and other aspects. The method proposed in this paper can guarantee transmission security with the adoption of watermark technology. Not only using ECG which is a kind of biological feature, but also establishing connections between the ECG feature and packet size, forming a relatively novel authentication mechanism. Whilst not exactly the same issue, the functionality of the authentication mechanism has yet to be further explored.

At present, it is undeniable that research on ECG information protection is still in its infancy, which is lack of related studies. The existing research may consist of the following categories:

1) Digital Watermarking Technology for Medical Imaging Area [5, 20];

2) ECG Monitoring Systems based on Sensor Network;

3) ECG Digital Watermarking Technology by Wavelet Transform;

4) ECG Transmission in Wireless Networks. Wavelet transform based digital watermarking encryption technology is mainly use for watermarking of the ECG signal. Thereby, our research provides great potential for researchers.

3.2 ECG Data Preparation

As shown in Figure 2, an electrocardiogram refers to the pacemaker, atrium and ventricle in each cardiac cycle, which continuously stimulates the electrocardiogram and the bioelectrical changes in the electrocardiogram. ECG traces are graphs of various forms of potential changes from the surface (referred to as ECG). An electrocardiogram is an objective indicator of the process of cardiac activation, transmission and recovery. ECG is an electrical activity in which the heart excites, and it has significant reference value for the basic functions of heart and pathology research. ECG can be used to analyze, even to identify a series of arrhythmias; it can also reflect the extent and development of myocardial injury and atrial and ventricular function as well as structural conditions. It provides the reference value for guiding cardiac surgery and advising on the necessary drug treatment.

The standard ECG lead to electrocardiogram waves, named by the Dutch physiologist William Einthoven (the inventor of the ECG). He divided one cardiac cycle into P, Q, R, S, T-waves.

P wave: P waves are generated by atrial depolarization, which is the excitement of the heart originates from the sinus node and then reaches the atria. This is the first wave of each wave group. In left and right atrium, p wave reflects the depolarization process. The front half of the P wave represents the right atrium, and the back half of the P wave reflects the left atrium.

QRS complex: Usually, A QRS complex consists of three closely connected waves. The first downward wave which is the Q-wave, along with a high-tip-Q-wave vertical wave called the R-wave. The downward wave followed by the R wave is called an S-wave. Science they are closely connected, and they reflect the excitement of the ventricular electrical process, it is collectively referred to as the QRS complex. This wave group reflects the left and right ventricular depolarization process.

T-wave: The T-wave is located in followed ST segment. It is relatively low and occupies much longer wave, which is produced by ventricular repolarization.

According to the above description, the ECG diagnosis mainly depends on the PQRST wave. Therefore, when we add a watermark, it is very indispensable to maintain the shape of these waveforms.

3.3 Principles of Network Transmission and Control

In this paper, we propose an architecture to ensure transmission security based on the control packet size. This will involve changes to the underlying network transmission. We need a control packet size for each transmission in a TCP/IP network protocol. Network transmission is a

communication process using a series of lines (such as optical fibres and twisted pairs) through the circuit changes and in accordance with the network transmission protocol [16]. Network transmission requires a medium, which is the network's physical path between the sender and the recipient, and has a certain influence on the data communication of the network. Common transmission media are optical fibers, twisted pair, coaxial cable and coaxial cable wireless transmission media. The network protocol is a specification for communicating and managing information in a network, including the Internet. Since the interaction between people needs to follow certain rules, the mutual communication between computers needs to comply with certain rules, which are called network protocols. Network protocols are usually divided into several levels, and communicating parties can only be connected to each other at in common level.

Common protocols are: TCP/IP, IPX/SPX, NetBEUI, *etc.* IPX/SPX is very usual in LAN. If the user wants to access the Internet, the TCP/IP protocol must be added to the network protocol. In this task, we mainly use the TCP/IP protocol [24]. TCP/IP is an abbreviation of "Transmission Control Protocol/Internet Protocol". TCP/IP is a network communication protocol that standardizes all communication devices on the network, especially the data format and transmission mode between the two parties.

TCP/IP is the basic protocol of the INTERNET, which is the standard method of data packing and addressing. In the process of data transmission, it could be represented as two envelopes with TCP and IP being like the envelope, the message is spliced into a number of segments, and each segment is delivered into a TCP envelope, and the envelope records the segment number information, then the TCP envelope is delivered in the IP envelope, and finally sent over the Internet.

At the receiving end, the TCP package collects envelopes, extracts data and restores the order. If an error is found, TCP will issue a repeat request. Therefore, TCP/IP on the Internet provides almost error-free data transmission. Ordinary users only need to know the IP address format, regardless of the entire structure of the network protocol, and then they can communicate with the rest of the world.

3.4 Principles of ECG Digital Watermarking

A common understanding of digital watermarking technology is that some identification information is directly embedded in a digital carrier (including multimedia, software, documents) or indirectly (modification of a specific area of the structure), which does not affect the use value of the original carrier, and is not easy to detect and modify, but can be identified by the producer. The hidden information in the operator can calculate the content creator and the purchaser, and can send out the secret information to find out whether the operator has been tampered with. As an effective means of copyright protection, digital watermarking is an important branch and research direction of information hiding technology research. Digital watermarking systems must have certain conditions to become a trusted application system for digital product copyright protection and integrity identification. A safe and reliable watermarking system should generally meet the following requirements:

- Concealment: Also known as imperceptibility. For an invisible watermark system, the watermarking embedding algorithm should not produce appreciable data modifications, namely the watermark in the normal viewing conditions should not be visible, and watermarking should not affect the visual effects of works.

- Robustness:

  Watermarking must be difficult to get rid of (hopefully impossible to remove). Of course, in theory any watermark can be removed as long as sufficient understanding of the process of watermark embedding is held. But if only a partial understanding of the watermark embedding process is held, any attempts to destroy or eliminate the watermark should lead to carriers of severe degradation, resulting not available.

- Tamper resistance:

  Uulike robustness, tamper resistance means that once the watermark is embedded in the carrier it is difficult for attackers to change or forge. Applications demanding high robustness usually require a strong tamper resistance. It is more difficult to achieve a good tamper resistance in copyright protection.

- Watermark capacity:

  Embedded watermark information must be sufficient to represent the multimedia content creator or owner of the flag, or the serial number of the purchaser. So when copyright disputes occur, the information of the creator or copyright owner is available, and the sequence number is used to indicate the users who have breached the agreement and provide multimedia data for piracy [15].

- Safety:

  The embedded information should ensure confidentiality and a low false detection rate. Watermarks can be any form of data, such as numeric, text, images, and so on. All watermarking embedded systems contain a watermark and watermark recovery system.

- Low error rate:

  Even in the case of attack or signal distortion, the probability of not detecting a watermark (undetected, false-negative) and detecting a watermark (false detection, false-positive) where the is none should be very small.

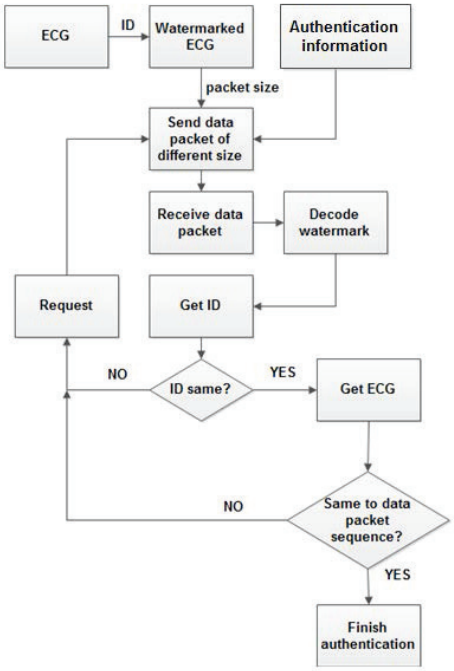


Figure 3: Overall flow of proposed algorithm

4 Proposed Algorithm

Our idea is like that at the transmitting end of the network we control the size of the transmission packet according to the ECG feature of the sender. Then we select a value calculated by the coefficient as the interval, to meet the range of packet sizes. Subsequently, senders send the designed packet, and at the same time the ECG sequence is transmitted as a key.

As shown in Figure 3, in the process of transmission the ECG sequence is watermarked, and the watermark is an ID number. The receiver receives the data, orders in accordance with the sequence, obtains the size of the packets, and puts them into a sequence. At the receiving end, the received ECG is de-watermarked to get the ID number and ECG sequence. According to the comparison of the ID number, the first step in the confirmation is obtained. If the ID number changes, or is beyond a certain range, it can detect that the information has be tampered with or suffered other attacks. And then the ECG signal, if the changes in the ECG signal do not exceed the threshold value, the network transmission can be identified as secure. Regardless of the kind of change to the network packet, its packet size will be changed, resulting in the export sequence being different to the original sequence. So we have to set the threshold standard. This is the preliminary idea of the authentication mechanism. This mechanism can be applied in various types of network identity authentication system, and can also be applied in a variety of network security transmissions.

The steps of the scheme are listed below:

1) Adding a watermark to the ECG;
2) Extracting number ID as the packet size;
3) Sending data as different sized packets;
4) Receiver open the packet;
5) Packet sizes are selected;
6) Decoding of watermark to get ID;
7) Checking of the size sequence (ID).

4.1 ECG Data Acquisition

In this research we use the MIT-BIH ECG data [12]. An ECG recording is composed of three parts:

1) Head files [.hea], which store the ASCII code character.
2) Data files [.dat], binary storage, each three bytes store two numbers, a number is 12 bit.
3) Notes file [.art], binary storage, format definition is more complex.

In the early stage of research, the methods of reading and the application of ECG data should be learnt and mastered [6].

4.2 ECG Watermarking Algorithm

We proposed a self-synchronous ECG digital watermarking encryption technology to achieve the protection of the transmission data. In addition, a series of ECG data is embedded in synchronous code, so that this data is self-synchronous. Further, some hidden data is embedded into DWT low frequency coefficients. Finally, the SNR (signal-to-noise ratio) and BER (bit error rate) are used for the analysis and evaluation of the overall effects. SNR measures the transparent of embedded data. The BER test after the addition of Gaussian noise, assesses the performance of the design algorithm.

In this process we use the mechanism of synchronous code. Next, we introduce the principle. It can be used to locate hidden information in order to prevent unpredictable attacks [26]. Supposing $\{a_i\}$ is a set of source-synchronous code, and $\{b_i\}$ is the location code which has the same length as A. If the difference between $\{a_i\}$ and $\{B_i\}$ is less than the set threshold, then $\{b_i\}$ will be recognised as a synchronisation code. Also there will be a fault-tolerance rate. Our formula is shown below. Assume that P_1 is a positive error rate, P_2 is a negative error rate, l is the length of the synchronisation code, and e is the threshold value we have set.

$$P_1 = \frac{1}{2^l} \cdot \sum_{k=l-e}^{l} C_l^k \tag{1}$$

$$P_2 = \sum_{k=e+1}^{l} C_l^k \cdot (BER)^k \cdot (l - BER)^{l-k} \tag{2}$$

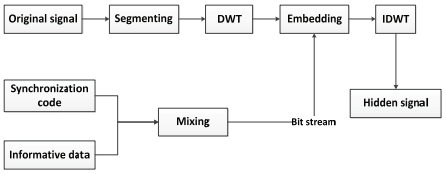


Figure 4: Embedding model

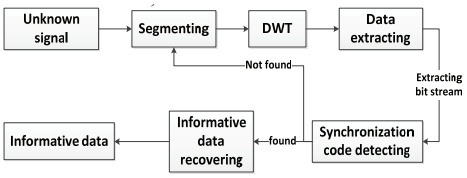


Figure 5: Extracting model

Taking into account the low-pass filtered improved robustness, the synchronisation code and the watermark is embedded in the low frequency sub-band coefficients of the seventh stage, which are the lowest frequency wavelet coefficients in our system. Then the synchronisation code and embedded information is mixed, when the watermarking ECG data is embedded in the low-frequency sub-band of the DWT factor, and then after the inverse transform, we obtained the watermarked ECG data. The detailed process model of embedment is shown in Figure 4.

In the data extraction section, data is firstly processed by the Wavelet transform which is the opposite to the embedding. It extracts binary data from the low frequency sub-band of wavelet coefficients. From these data, we can restore the synchronous code. In the signal analysis process, the selected action is repeated until the synchronous code is detected. The details of the process description are shown in Figure 5. After determining the location of the synchronisation code, we can extract the hidden information.

In this step, the ECG signal is segmented, and each segment executes the DWT transform. Then the sequence {MI} is embedded in each frequency band. The length of the signal segment depends on the size of the wavelet decomposition level. Of course, this length should be at least capable of accommodating a synchronous code and a number of data information. The embedded rules are as follows:

$$X_k^{'} = \begin{cases} \lfloor X_k/\alpha \rfloor \cdot \alpha + 3\alpha/4 & W_k = 1 \\ \lfloor X_k/\alpha \rfloor \cdot \alpha + \alpha/4 & W_k = 0 \end{cases} \quad (3)$$

In this equation, X_k is the original DWT transform coefficient, $X_k^{'}$is the watermarking DWT transform coefficient, and α is the embedding strength.

In addition to the synchronous code, when embedding the watermark in the ECG, we use the concept of payload that the number of bits, to measure the rate of each unit (bit per second). And as B are used in the following formula. Supposing the sampling rate is R(Hz), the wavelet decomposition level is K. And the formula is as follows:

$$B = R/2^K bps \quad (4)$$

During data extraction, the signal of the ECG embedded watermark is similarly segmented. These fragments include at least one synchronous code segment. Then each segment executes wavelet transformation. Assuming $X_k^{'}$ is the coefficient of the low-frequency sub-band, the sequence of W_k^* is extracted from $X_k^{'}$ using the rules shown below.

$$W_k^* = \begin{cases} 1, & if\ X_k^{'} - \left\lfloor X_k^{'}/\alpha \right\rfloor \cdot \alpha \geq \alpha/2 \\ 0, & if\ X_k^{'} - \left\lfloor X_k^{'}/\alpha \right\rfloor \cdot \alpha < \alpha/2 \end{cases} \quad (5)$$

5 Evaluation

In this section, our data are selected from the MIT-BIH ECG database. Under such conditions, the embedded data transmission rate of the synchronisation code is evaluated, and the payload adopting this method tested. The SNR and BER are mainly used in the assessment. SNR and BER have been introduced in the previous section, and their formulas are given as Equations (6) and (7). Of course, the characteristics of the watermark are transparent to the user, and the presence of the watermark does not affect the user. Therefore we should as far as possible ensure that the watermark signal is consistent with the original signal [1].

First of all, define SNR and BER.

$$SNR = -10\log_{10}\left[\left(\sum\nolimits_i \left(f_i^{'} - f_i\right)^2 \Big/ \left(\sum\nolimits_1 f_i^2\right)\right)\right] \quad (6)$$

$$BER = \frac{Number\ of\ error\ bits}{Number\ of\ total\ bits} \times 100 \quad (7)$$

In the above formula, f_i and $f_i^{'}$ represent the original signal and the modified signal.

Firstly we selected four sets of data from the MIT-BIH database. Then we tested the running of the algorithms using these four groups of data. The first step is to contrast an improved method with the original method which was applied to audio. In the last part of the experiment, the white noise attack multiplication factor was set to control the different degrees of attack. A total of four values were obtained: 1,50,500,1000. The table 1 is the contrast effect for the SNR value before and after modification.

From the above table it can be seen that after our modification the value of SNR has been improved to some extent. This shows that after our modifications this method is more suitable for the application of the ECG data encryption, and by using this method we can get a better result.

In Figure 6 the blue curve represents the original signal, and the green curve shows the watermarked signal.

From Figure 7 we can see that the difference between the original signal and watermarked signal is very small,

Table 1: Comparison of SNR before and after modification

Data	Before modification SNR	After modification SNR
Class 1	27.2382	30.4968
Class 2	25.6382	30.6291
Class 3	29.7743	31.7761
Class 4	28.3629	32.0753

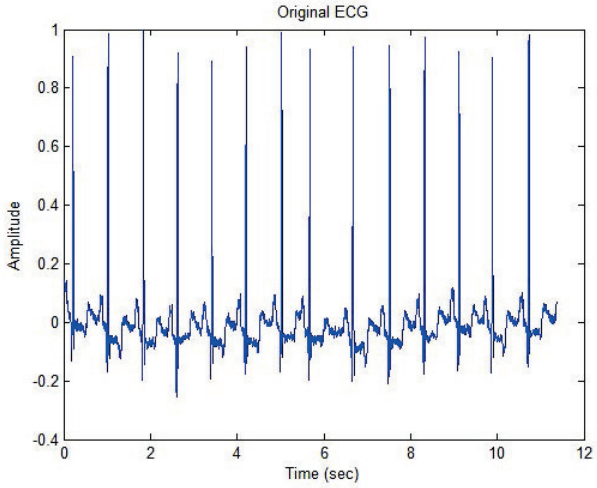


Figure 6: Original ECG signal

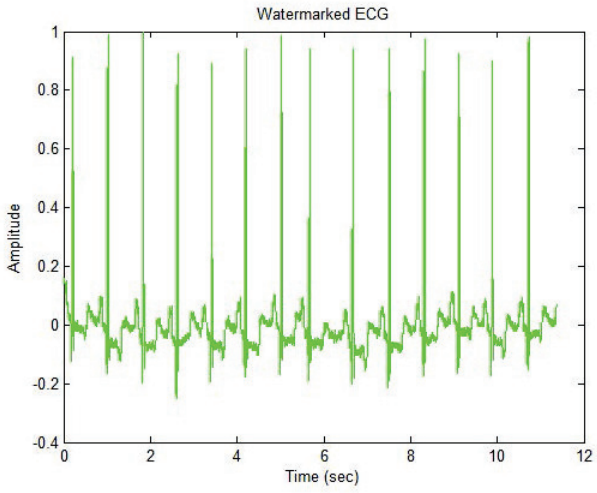


Figure 7: Merger of watermarked and original ECG signal

Table 2: Watermark basic parameters

Data	Fs	Time	SNR	level
Class 1	360	10	30.4968	7
Class 2	360	10	30.6291	7
Class 3	360	10	31.7761	7
Class 4	360	10	32.0753	7

Table 3: Watermark robustness test

Data	WhiteNoise	BER	Error	SNR_noise
Class 1	1	0	0	72.6337
Class 1	50	0	0	38.6543
Class 1	500	0	0	18.6543
Class 1	1000	21.8750	7	12.6337
Class 2	1	0	0	71.8875
Class 2	50	0	0	37.9081
Class 2	500	0	0	17.9081
Class 2	1000	21.8750	7	11.8875
Class 3	1	0	0	73.9786
Class 3	50	0	0	39.9992
Class 3	500	0	0	19.9992
Class 3	1000	21.8750	7	13.9786
Class 4	1	0	0	73.7127
Class 4	50	0	0	31.3995
Class 4	500	0	0	19.7333
Class 4	1000	21.8750	7	13.7127

and with the naked there is almost no difference. In the above test, the ECG sampling rate is 360. In each of the data signals, the fragments whose length is 4096 were chosen to test. Using a synchronous code whose length is 63, and a 256 bit watermark sequence. In the third chapter, the threshold is defined as 21. The Haar wavelet transform has eight layers of decomposition level, which is also mentioned in the previous section.

Table 2 is a test of four groups of data, for which different SNR values are obtained. In this table we can see that the SNR value is over 30, which is able to meet the accuracy demand. Under such conditions, it does not affect the diagnosis of the doctor for the ECG image.

Subsequently, white noise attack is tested on the watermarked signal. Thus we can better understand the robustness of watermark [21]. After the signal is watermarked, the white noise is added. They can be used to simulate transmission channel noise and possible attack. The specific data are shown in Table 3.

From Table 3 it can be seen that the ordinary size of noise cannot affect the watermark. From the data, when the noise factor is 1, 50 and 500, the values of BER are all 0. Even under the influence, the error rate is 0. When the white noise coefficients were 1 and 50, the SNR with noise is greater than 30. Today's popular ADSL broadband network basically does not produce much noise [3]. According to industrial regulations, if the network noise reaches 30–50 dB then the network is largely as unusable. When the noise size is about 5 dB, the network has begun

to enter an unstable state. In this state many failures and errors can occur in the network. Only when the noise is sufficiently large, such as a coefficient of more than 500 times, will some deviations appear. However, such a large noise is very rare in the real environment. Therefore the watermark robustness of this method is good.

In this section we conduct an in-depth study of some of the papers on ECG watermarking, and select a number of representative methods for comparison. Then, according to our understanding these methods are implemented and compared. According to the various parameters listed in the literature, an attempt is made to make the test data consistent with the original experiment. So the results and the original intention of the author are not too different. After testing it can be seen that the difference between the obtained data and the original data is not too large, and is basically the same. Although there is a little deviation, overall it is still in an acceptable range. The experimental results listed below have basically achieved the desired results [8].

One of these studies is "Wavelet Transformation Based Watermarking Technique for Human Electrocardiogram (ECG)", whose authors are Mehmet Engin, Oğuz Çıdam and Erkan Zeki Engin. The main idea of the article bears some similarities with our approach. Therefore there is value in contrasting it. Firstly, the ECG signal is decomposed into 8 sub-bands through discrete wavelet transform. Then they calculate the average power of each sub-band. Finally, the random sequence is inserted into the watermark signal and the selected low frequency subband. In the process they use the Daubechies Wavelet function (DB2) and bi-orthogonal functions (bior5.5) [23].

In the process of watermark embedding they use a random Gaussian noise and Theravada as the coefficients for generating a random sequence. However the embedding process is only a simple process of summation. In the watermark extraction process, they need to use the original ECG to calculate the watermarked sequence. In such a case, the significance of network security will be lost. And, in this way we cannot guarantee the security of the transmission. If the original ECG signal and watermarked ECG signal is modified at the same time, they may not be detected. In terms of data, they selected four typical data sets, including a normal ECG and diseased ECG. Their data are also from the MIT-BIH database [18]. After this method was studied we implemented it. In the original article they have compared the two methods of DB2 and bior5.5 [18], and proven that the effect of bior5.5 is worse than DB2, so we only tried the DB2 method.

Compared with their method, the most obvious difference with ours is the ability of self-synchronisation. Furthermore, in the process of watermark extraction, we do not require the use of the source ECG. The following table shows the implementation of the two methods, and their testing using the same data. It compares the SNR values of the two methods with watermark.

As can be seen from the table above, the SNR value using our methods are significantly higher. This shows

Table 4: SNR comparison

Data	Our SNR	Their SNR
Class 1	30.4968	20.9100
Class 2	30.6291	20.3561
Class 3	31.7761	21.2650
Class 4	32.0753	24.9800

Table 5: The robustness of their method

Data	WhiteNoise	BER	Error	SNR_noise
Class 1	1	28.1250	9	72.6197
Class 1	50	31.2500	10	38.6403
Class 1	500	37.5000	12	18.6403
Class 1	1000	50.0000	16	12.6197
Class 2	1	56.2500	18	71.9227
Class 2	50	53.1250	17	37.9433
Class 2	500	53.1250	17	17.9433
Class 2	1000	71.8750	23	11.9227
Class 3	1	62.5000	20	73.9056
Class 3	50	62.5000	20	39.9262
Class 3	500	56.2500	18	19.9262
Class 3	1000	53.1250	17	13.9056
Class 4	1	56.2500	18	73.6919
Class 4	50	56.2500	18	39.7125
Class 4	500	65.6250	21	19.7125
Class 4	1000	65.6250	21	13.6919

that after the watermark embedding process the use of our method produces less noise.

On the table 5, we can know the robustness with adding white noise when embed watermark [10]. The SNR values are listed in the table. After finishing the extraction of the watermark, BER was used to assess the robustness of this approach [11]. From the comparison of the data obtained with the data from Table 5, it can be seen that the error rate of extracting the watermark sequence is relatively low when using our method, and the noise coefficient increases.

As shown in Figure 8 this is a discount figure. It visually shows the effect of the BER comparison of the two methods. The error rate of the sequence watermark obtained using our method is significantly lower

In the end we tested 48 sets of data, are all from the

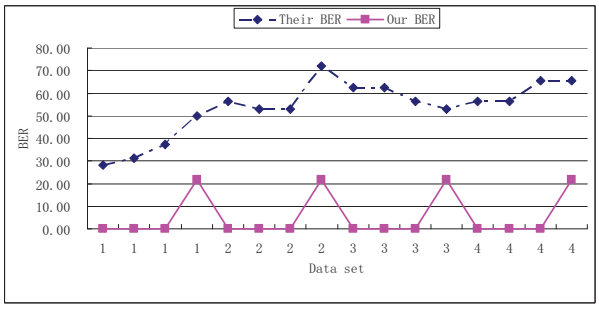


Figure 8: BER comparison of the two methods

MIT-BIH database. Using our method the data obtained is presented in Table 6. SNR represent the signal-to-noise ratio without noise [17]. SNR_noise represents the signal-to-noise ratio with noise attacks. The noise factor is 500.

Table 6: 48 datasets

	SNR	SNR_noise		SNR	SNR_noise
100	30.4968	18.6543	201	33.2101	20.1639
101	29.0157	17.9081	202	32.2859	19.8937
102	29.7	19.9992	203	34.9532	23.9485
103	31.3995	19.7333	205	29.9598	19.3858
104	30.9927	20.0396	207	36.2762	25.0338
105	32.6936	21.8737	208	36.3481	24.5461
106	33.5246	21.0478	209	31.0114	20.4217
107	37.5646	26.2966	210	34.9704	22.3635
108	32.534	22.831	212	33.2044	22.7276
109	34.7681	24.9335	213	33.9076	23.0179
111	33.3272	21.1172	214	34.4189	23.0512
112	33.4805	22.0718	215	34.3245	23.216
113	31.5873	21.0054	217	38.1804	25.8837
114	31.9882	21.6256	219	33.7499	21.7806
115	30.249	18.4942	220	30.6738	19.4284
116	33.6845	21.1607	221	32.2778	21.4587
117	34.5648	23.0738	222	30.3614	19.8714
118	36.1235	24.1986	223	35.0769	21.4536
119	31.5782	20.8796	228	33.4002	23.0628
121	34.9675	21.7999	230	31.2626	20.7424
122	33.9329	22.6893	231	31.627	19.9292
123	28.8724	18.2877	232	32.2554	20.3259
124	30.5468	19.9755	233	34.9552	24.7316
200	32.9446	22.4092	234	29.8672	20.7317

The overall trend of the data in Table 6 is shown in Figure 9.

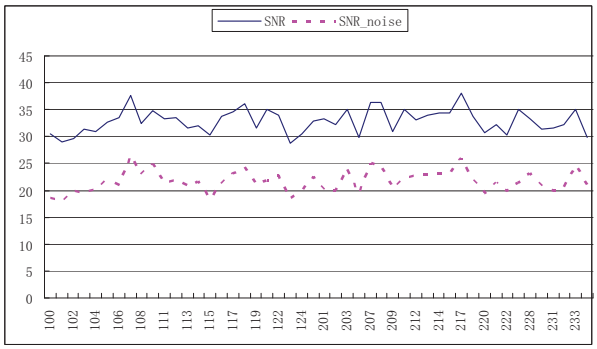


Figure 9: Overall trend

Now, in order to verify our search result we developed a system. The algorithm of the watermarking and decoding are implemented by Matlab, and the sender and receiver, which can send and receive the package, are coded by C code. We have built the project to transmit a message by TCP/IP packet size based on the ECG biological feature for a security network.

The execution screen can be viewed in Figure 10 for the sender (left widow) and for the receiver (right widow).

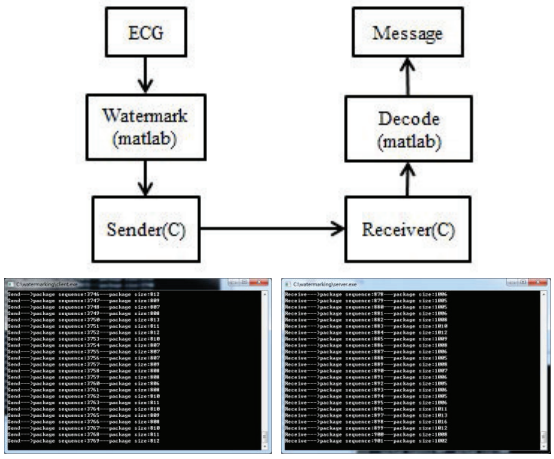


Figure 10: Execution screens for the implemention code

And we have shared the code on Google Code. If needed you can download the source code from website: `https://code.google.com/p/my-watermarking-ecg/`. If you want to run the system, you can refer to the process below. The system contains the programs: Watermarked.m, client.exe, server.exe and Decode_watermark.m. Firstly run Watermarked.m, with ecg_embed.dat, which contains the watermarked ECG. Next you can run client.ext and server.exe which will send and receive the package. The server will receive the message and place the message in ecg_embed_recv.dat and ecg_recv.data. ecg_embed_recv.data contains the package size message, and ecg_recv.data contains the receiving data. Finally, run Decode_watermark.m, which decodes the data of ecg_recv.data, and obtains the watermarked message.

6 Conclusion

In this paper we firstly discussed the security issues in network transmission, and presented a new secure transmission scenario, packet watermarking with ECG signal. Then we provided further discussion on the proposed algorithm, especially that on transmission watermarking research. This proposed an idea for secure watermarking transmission with an ECG feature to control the packet size. In order to guarantee the security of the watermarked transmission, we adopted the ECG feature and self-synchronisation technology, this focused on the idea of a self-synchronised watermark. The next detailed describe the specific steps about how the texts uses the self-synchronous wavelet watermark with ECG signal. In the end, we evaluated our system and compared it with some other methods.

In our proposed algorithm, the self-synchronous ECG digital watermark technology is used in security transmission to protect the key link. In order to enhance the robustness, the synchronisation code sequence and the watermark sequence are embedded into the lowest frequency wavelet coefficients. To improve efficiency, in the process of synchronisation code searching the time frequency of

the wavelet transform is used to to locate the code .

From the experimental results, we can see that the watermarked ECG signals have high SNR values; it can resist common attacks, and is robust to changes in the waveform. In the experiment, a number of representative ECG signals were selected to obtain a general conclusion. However, it is just a small part compared to the larger ECG database. At the same time, we think the experimental results can be improved, and better result obtained through different approaches. We also hope this technology can be applied to other fields, such as brain waves and hand signatures, which may need further study and research.

References

[1] D. Anand and U. C. Niranjan, "Watermarking medical images with patient information," in *Proceedings of the 20th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 703–706, Nov. 1998.

[2] N. Bousnina, S. Ghouzali, M. Lafkih, O. Nafea, M. Mikram, W. Abdul, and D. Aboutajdine, "Watermarking for protected fingerprint authentication," in *International Conference on Innovations in Information Technology*, pp. 1–5, 2017.

[3] S. T. Chen, H. N. Huang, J. S. Pan, K. K. Tseng, and S. Y. Tu, *Audio Watermarking Quantization Based on Minimum-length Scaling Scheme*, National Science Council, TAIWAN, under the NSC grant: NSC 98-2115-M-029-006-MY2, 2009.

[4] G. P. Feng, "computer network unsafe factor analysis and countermeasures," *Shanxi Coking Coal Science and Technology*, vol. 11, no. 11, pp. 30–31, 2005.

[5] J. J. Garciahernandez, W Gomezflores, and J Rubioloyola, "Analysis of the impact of digital watermarking on computer-aided diagnosis in medical imaging," *Computers in Biology & Medicine*, vol. 68, pp. 37–48, 2016.

[6] K. Ibrahim I. Ayman and S. Ron van, "A low complexity high capacity ecg signal watermark for wearable sensor-net health monitoring system," in *Computing in Cardiology*, pp. 393–396, Sep. 2011.

[7] A. Jadhav and M. Kolhekar, "Digital watermarking in video for copyright protection," in *in Proceedings of the International Conference on Electronic Systems, Signal Processing and Computing Technologies*, pp. 140–144, Dec. 2014.

[8] Y. L. Kan, "Digital audio watermark technology based on fourier transform," *Journal of Beijing Broadcasting Institute*, vol. 01, 2005.

[9] Z. Ling, "Anti-tampering technology of digital watermarking in dynamic web page images," *Agro Food Industry Hi Tech*, vol. 28, no. 1, pp. 2195–2199, 2017.

[10] K. D. Manab P. Dipti and P. Smita, "Integration of FCM, PCA and neural networks for classification of ecg arrhythmias," *IAENG International Journal of Computer Science*, vol. 3, 2010.

[11] J. Mehrdad, E. Reza, S. Atena, F. Soheil, and Z. Shokoufeh, "Improving ecg classification accuracy using an ensemble of neural network modules," *Plos One*, vol. 6, no. 10, pp. e24386, 2011.

[12] M. S. Nambakhsh, A Ahmadian, M Ghavami, R. S. Dilmaghani, and S Karimi-Fard, "A novel blind watermarking of ecg signals on medical images using EZW algorithm," in *International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 3274–3277, 2006.

[13] J. Peng, "On the computer network security," *Sichuan University of Arts and Science Journal*, vol. 20, no. 9, pp. 73, 2010.

[14] F. Qin and C. L. Shi, "Authenticated encryption schemes: Current status and key issues," *Computer Knowledge and Technology*, vol. 6, no. 5, pp. 4118–4119, 2010.

[15] S. Riya, K. Suneet, F. Omar and S. A. Bhavneet, "Watermarking for protected fingerprint authentication," in *Proceedings of the 12th International Conference on Innovations in Information Technology (IIT'10)*, pp. 140–144, Mar. 2010.

[16] X. Shen, C. Lin, Y. Sun, J. Pan, P. Langendoerfer, and Z. Cao, "Wireless network security," *Wireless Communications & Mobile Computing*, vol. 6, no. 3, pp. 269–271, 2006.

[17] P. Singh and P. K. Mann, "Fast fourier transformation based audio watermarking using random sample," *ResearchGate*, 2011. (`https://www.researchgate.net/publication/268184106_Fast_Fourier_Transformation_Based_Audio_Watermarking_using_Random_Sample`)

[18] L. W. Tang, K. M. Zheng and X. Qian, "Watermarking technology for electrocardiogram signal certification," *Computer Engineering and Applications*, vol. 45, no. 20, pp. 231–233, 2009.

[19] A. Z. Tirkel R. G. van Schyndel and C. F. Osborne, "A digital watermark," in *Proceedings of 1st International Conference on Image Processing*, pp. 86–90, Nov. 1994.

[20] C. S. Tsai N. I. Wu, C. M. Wang and M. S. Hwang, "A certificate-based watermarking scheme for coloured images," *The Image Science Journal*, vol. 56, no. 6, pp. 326–332, 2008.

[21] D. H. S. Wu, J. Huang and Y. Q. Shi, "Efficiently self-synchronized audio watermarking for assured audio data transmission," *IEEE Transactions on Broadcasting*, vol. 51, no. 1, pp. 69–76, 2005.

[22] Y. Wu, "digital watermark technology research matlab simulation," *Technology Applications*, pp. 37, 2005.

[23] H. Y. Yang, X. Y. Wang, and H. Zhao, "Digital audio watermarking algorithm based on adaptive quantization," *Technical Acoustics*, vol. 02, 2004.

[24] F. Zhang, and J. F. Ma, "Authentication mechanisms, performance and security analysis," *Xi'an University of Electronic Technology*, vol. 4, 2004.

[25] J. M. Zhang, *Digital Watermarking Technology and Computer Application Technology*, Shandong University of Science and Technology, Master's Degree Thesis, 2005.

[26] K. M. Zheng, and Q. Xu, "Reversible data hiding for electrocardiogram signal based on wavelet transforms," in *In Proceedings of the International Conference on Computational Intelligence and Security*, pp. 295–299, Dec. 2008.

Biography

Kuo-Kun Tseng, is an associate Professor and Shenzhen Peacock B-level talent, born in 1974, received his doctoral degree in computer information and engineering from National Chiao Tung University of Taiwan in 2006. Since 2004 he has many years of research and development experience, long engaged in biometric systems and algorithms research. The current research results, published more than 70 articles, of which about 30 is a high impact factor of the SCI or the famous ACM / IEEE series of journals.

Xialong He, was a graduate student at Harbin University of Technology (Shenzhen), and now is a senior software engineer. His expertise is watermarking and biometric processing.

Xiaoxiao An was a graduate student at Harbin University of Technology (Shenzhen), and now is a senior software engineer in Alibaba Corporation. His expertise is machine learning and biometric processing.

Chin-Chen Chang received his Ph.D in computer engineering in 1982 from the National Chiao Tung University, Taiwan. He was the head of, and a professor in, the Institute of Computer Science and Information Engineering at the National Chung Cheng University, Chiayi, Taiwan. From August 1992 to July 1995, he was the dean of the College of Engineering at the same university. From August 1995 to October 1997, he was the provost at the National Chung Cheng University. From September 1996 to October 1997, Dr. Chang was the Acting President at the National Chung Cheng University. From July 1998 to June 2000, he was the director of Advisory Office of the Ministry of Education of the R.O.C. Since February 2005, he has been a Chair Professor of Feng Chia University. He is currently a Fellow of IEEE and a Fellow of IEE, UK. He also published several hundred papers in Information Sciences. In addition, he has served as a consultant to several research institutes and government departments. His current research interests include database design, computer cryptography, image compression and data structures.

Chao Wang is a master student at Harbin University of Technology (Shenzhen), his interested in the deep learning algorithm and bio-signal processing.

Xiangmin Guo is the deputy dean of Shenzhen Branch of Urban Planning & Design Institute of Harbin Institute of Technology (HIT), and associate professor in Harbin Institute of Technology, Shenzhen Graduate School.

Provable Secure for the Ultra-lightweight RFID Tag Ownership Transfer Protocol in the Context of IoT Commerce

Jia-Qi Wang[1], Yun-Feng Zhang[2], and Dao-Wei Liu[3]
(Corresponding author: Jia-qi Wang)
Information Engineering Institute, Zhujiang College of South China Agricultural University[1]
Guangzhou 510900, China
Continuing Education College, Guangzhou City Construction College[2]
Department of Computer Science and Engineering, Guangzhou College of Technology and Business[3]
(Email: wjqldw@126.com)
(Received July 22, 2018; Revised and Accepted Dec. 28, 2018; First Online June 22, 2019)

Abstract

Under the business environment, the ownership of an (Radio Frequency Identification, RFID) RFID tag embedded item often shifts, and the ownership of the corresponding item must also be transferred, so the privacy of the original owner and the new owner needs to be protected during the transfer of ownership. In order to protect the privacy of tag's ownership during the transfer process, an RFID tag's ownership transfer protocol base on bitwise operation (*PSU-TOTP*) is proposed. The proposed protocol uses bitwise cross-synthesis and cross-connect operations to encrypt the transmitted information and reduce the amount of computation at the tag. The flag *FLAG* is introduced to record the ownership of the current owner. The abstract description of the security model and protocol is given, and the proposed protocol is comprehensively analyzed to meet the corresponding security requirements under the security model. Security analysis shows that the proposed protocol meets the security requirements for the tag ownership transfer. The formalization of GNY logic proves the correctness of the proposed protocol. Performance analysis shows that the proposed protocol can effectively reduce the computational load on the tag side and achieve the goal of reducing the tag's cost. *PSU-TOTP* is suitable for low-cost RFID systems.

Keywords: Cro-Link; Cro-Syn; Index Terms-IoT Business; Ownership Transfer; RFID; Ultra-Lightweight

1 Introduction

RFID is a kind of technology that automatically recognizes and obtains data. By embedding an RFID tag into a specific target, such as embedding an RFID tag in a bus card, the reader can recognize the specific target and read the data without directly contact [15]. The RFID tag has been widely used in manufacturing, transportation, wholesale and retail, and other fields because of its low cost, wide range of read and write, easy to carry, long service life and data encryption [25].

In practical applications, its owner will change frequently during the lifecycle of an RFID tag [12]. For example, an embedded RFID tag product, before it is not shipped, its ownership should be attributed to the manufacturer. When the product is sold by the manufacturer to the wholesaler, the ownership of the product is attributed to the wholesaler at this time. After the wholesaler resells the product to the retailer, the ownership of the product is owned by the retailer [28].

In the transfer process, the ownership of the RFID tag belongs to various owners, so we must protect the privacy of the corresponding owners [24].For example, after the manufacturer wholesales the product to the wholesaler, it must ensure that the manufacturer does not have permission to read the private information stored in the tag and that the wholesaler does not have access to the private information stored by the manufacturer [4]. In the process of the RFID tag ownership transfer, more and more scholars pay more attention to the security of the tag's private information, and also propose many ownership transfer protocols. However, there are more or less certain security flaws or large computations in these protocols [3].In order to solve the above problems, a provably secure and ultra-lightweight protocol for the RFID tag ownership transfer (*PSU-TOTP*) is proposed. The protocol uses bitwise operations to encrypt information so it can achieve the ultra-lightweight level. At the same time, the use of bitwise operation can effectively reduce the computational load of the tag. The introduction of flag *FLAG* can identify the current owner of ownership according to its value. Security and performance analysis

shows that the proposed protocol can meet the security requirements of ownership transfer and reach the goal of reducing the cost of the tag. *PSU-TOTP* can be appropriately used in existing RFID systems.

The first section of this article is the introduction, which tells that the ownership of RFID tag embedded items often changes during its life cycle. To protect the privacy of the tag, the ownership transfer protocol is proposed, which leads to the focus of this paper. The second section introduces some of the classic RFID tag ownership transfer protocol, and points out some of the shortcomings and deficiencies. The third section introduces the mathematical knowledge and symbol meaning used in the design process of *PSU-TOTP*. The fourth section establishes a security model of *PSU-TOTP* for RFID system. The fifth section gives an abstract description of the ownership transfer protocol for the applicable security model. The sixth section systematically describes the design steps of *PSU-TOTP*. The seventh section analyzes the security requirements that *PSU-TOTP* satisfies the transfer of ownership under the security model. The eighth section uses GNY formal logic to rigorously prove *PSU-TOTP*. The ninth section analyzes the performance of *PSU-TOTP* in detail from the aspects of the tag computation and storage space. The tenth section summarizes the whole paper, and gives the next research direction.

2 Related Works

Molnar *et al.* first proposed the concept of ownership transfer of RFID tags in 2005, and gave a protocol about the transfer of ownership of RFID tags in Reference [13], but it required both the original owner and the new owner to believe the trusted center, which made the protocol limited.

An ownership transfer protocol based on the hash function mechanism is proposed in Reference [14], but the analysis finds that the protocol can‘t resist denial of service attacks.

In Reference [6], a new protocol is proposed. Since the RFID tag returns a hash value of Kp and a random number each time, but the random number is generated by the tag itself, the attacker can reuse the return value, and thus can impersonate the tag, so the protocol can‘t resist impersonate attacks.

In Reference [10], a simple and efficient ownership transfer scheme is proposed, which is based on the existing problems in Reference [14] and can effectively solve the denial of service attacks existing in the original protocol.

The security and privacy protection requirements of the RFID tag ownership transfer protocols are given in Reference [16] and three sub-protocols are also given. However, the analysis shows that they can‘t resist the desynchronization attacks.

Later, Song himself proposed an improved solution to the existing deficiencies in Reference [16] and Reference [17], but the improved scheme still does not resist the desynchronization attacks and can‘t meet the security requirements of backward privacy protection.

Based on SQUASH, Reference [11] gives a scheme of ownership transfer. According to the analysis, the new owner can obtain the public and private keys shared by the original owner and the tag, so that the new owner can access the private information stored by the original owner. Therefore, the protocol does not meet the security requirements of forward privacy protection. An attacker could obtain the information and block the new owner from communicating with the tag. Through the remessage, the shared private key between the tag and the new owner may be out of synchronization, so the protocol can‘t resist the replay attacks and desynchronization attacks.

In Reference [5], a solution of ownership transfer is proposed and a security model is given. However, the analysis shows that the security model has some limitations. This assumption makes the solution unable to provide effective privacy protection in practical application.

In Reference [1], a provable secure RFID tag ownership transfer protocol is proposed. It is found that the transfer of some random number in the ownership transfer protocol is caused by transferring the plain text, which allows the attacker to obtain the random number through wiretapping and then to forcibly crack some of the tag's private information by brute-force means, so the protocol can‘t resist brute-force attack.

In Reference [27], an ownership transfer protocol based on the quadratic residue theorem is proposed. It is found that the protocol does not implement bidirectional authentication between the original owner and the tag, so the protocol can‘t ensure that the transferred tag is the target tag. Therefore, it can‘t resist impersonation attacks.

3 Related Knowledge Introduction

1) Bitwise cross-synthesis operation.

In this paper, to facilitate the use of the symbolic description, we use the symbol *CroSyn(X, Y)* to represent the cross-synthesis operator. The cross-synthesis operation *CroSyn(X, Y)* is defined as follows: Let X, Y, Z be three binary numbers all of which are even l bits, $X = x_1x_2\cdots x_L$, $Y = y_1y_2\cdots y_L$, $Z = z_1z_2\cdots z_L$, where $X \in \{0,1\}^l$, $Y \in \{0,1\}^l$, $Z \in \{0,1\}^l$. We obtain the i-th bit in the binary number X, and simultaneously obtain the $(i+1)$-th bit in the binary number Y. We perform different operations according to the Hamming weight for obtaining two bits, then place them in order and finally synthesize a new binary number Z. If Han-ming weight is odd, bitwise XOR operation is performed; otherwise, bitwise AND operation is performed [9, 19].

Cross-synthesis operation in the tag is implemented in the form of pointers, making it more efficient than the direct use of logic gates. Two pointers are introduced, one for P_X and the other for P_Y; where pointer P_X points to binary number X and pointer P_Y points to binary number Y.When the pointer P_X traverses from the first bit of the binary number X, the pointer P_Y starts traversing from the second bits of the binary number Y at the same time. Based on the traversal, we can get the value of the two bits, and judge their value of Hamming weight (if it's an odd value, perform XOR operation; if it's an even value, perform AND operation), and store the operation result in turn. Finally, calculate *CroSyn(X, Y)* to get a new binary number Z. For example, if $l = 8$, *X=11011001, Y=01100101*, then *CroSyn(X,Y)=11101101*. The specific process may refer to Figure 1.

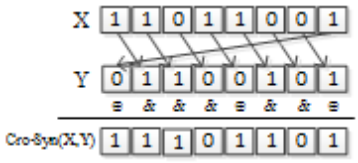


Figure 1: Flow chart of bitwise cross-synthesis operation

2) Bitwise cross-link operation.

In this paper, to facilitate the use of the symbolic description, we use the symbol *CroLink(X, Y)* to represent the cross-link operator. The cross-link operation *CroLink(X, Y)* is defined as follows: Let X, Y, Z be three binary numbers all of which are even l bits, $X = x_1x_2\cdots x_L$, $Y = y_1y_2\cdots y_L$, $Z = z_1z_2\cdots z_L$, where $X \in \{0,1\}^l$, $Y \in \{0,1\}^l$, $Z \in \{0,1\}^l$. We obtain the i-th bit and the $(i+1)$-th bit in the binary number X, and simultaneously obtain the i-th bit and the $(i+1)$-th bit in the binary number Y. According to the obtained four bits' Hamming weight we perform different operations, and then place them from low to high position to get a new left half of the binary Z. Similarly, place them from high to low position to get a new right half of the binary Z. Finally, the left and right half can be linked to get the binary number Z with the length of even l bits. If the Hamming weight is an odd value, perform AND operation; if it's an even value, perform XOR operation [26].

Cross-link operation in the tag is implemented as described below. Two pointers are introduced, one for P_1 and the other for P_2; where pointer P_1 points to the beginning of binary number X and pointer P_2 points to the beginning of binary number Y. When the pointer P_1 traverses from the beginning of the binary number X, the pointer P_2 starts traversing from the beginning of the binary number Y at the same time. We obtain the i-th bit and the $(i+1)$-th bit in the binary number X, and simultaneously obtain the i-th bit and the $(i+1)$-th bit in the binary number Y. Then judge the Hamming weight of the four bits. If it's an odd value, perform AND operation; if it's an even value, perform XOR operation. The calculated value is placed on the left half and the right half of the binary number, respectively, of which the left half is placed from low to high position and the right half is placed from high position to low position. Finally, link the left half and the right half and we can obtain the binary number Z with the length of even l bits.

Cross-link operation only needs traversal, bitwise OR operation, bitwise AND operation and the final link operation, which reduces the amount of system computing and storage, and achieves the ultra-lightweight level. For example, if $l = 8$, $X = \{11011001\}$, $Y = \{01100101\}$, then CroLink(X,Y)= $\{01111110\}$. The specific process may refer to Figure 2.

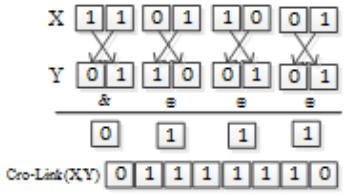


Figure 2: Flow chart of bitwise cross-link operation

4 Security Model

4.1 Communication Model

The existing RFID system generally consists of three parts: The tag T, the reader R and the database DB. The computing power of the tag T is limited, and its storage space is small, but the database DB has a strong data processing capabilities (such as data calculation, data query). The reader R is located between the tag T and the database DB. The reader R completes the communication process by forwarding the information of the tag T to the background database DB or forwarding the information of the database DB to the tag T [20–23]. In the current research, the RFID system is generally suitable for the following assumptions: The communication link between the tag T and the reader R is not secure, and the communication link between the reader R and the database DB is secure. The general communication process of RFID system is shown in Figure 3.

The tag ownership refers to the ability to identify the tag and be able to control all the information associated with the tag. The tag ownership transfer refers that the original owner no longer have the ownership of the tag, and the new owner has the control of the tag. Because the communication link between the reader and the database is safe and reliable, we can the both as a whole. In this paper, there are mainly three entities involved in the final

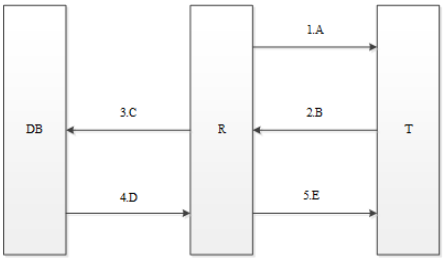


Figure 3: Flow chart of the communication process of RFID system

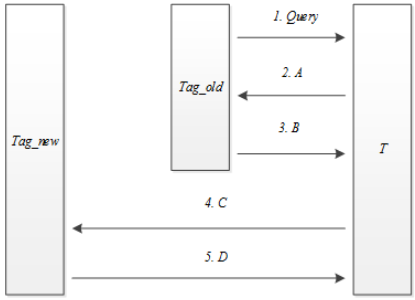


Figure 4: The flow chart of the abstract description of PSU-TOTP protocol

protocol communication by simplifying the research object in the process: The tag's original owner Tag_{old}, the tag's new owner Tag_{new} and the tag Tag.

4.2 Attack Model

In the RFID tag ownership transfer protocol, the method that an attacker A may use is mainly based on channel. For the channel-based attacks, we assume that the attacker A has full control over the communication channel between the original owner Tag_{old} and the tag Tag, and has complete control over the communication channel between the new tag's owner Tag_{new} and the tag Tag. The connotation of control here is that attacker A can arbitrarily read, tamper, delete, replay any message in the channel, and at the same time, he can initiate any conversation with any participant at any time [8]. Channel-based attacks mainly include replay attacks, man-in-the-middle attacks, privacy attacks, desynchronization attacks, tracking attacks and impersonation attacks.

4.3 Security Requirements

A secure and reliable RFID tag ownership transfer protocol needs to meet the following security requirements [18].

1) Backward privacy protection: After the ownership transfer completes, the tag's original owner Tag_{old} can no longer recognize the tag Tag, and can't access the session information between the Tag. Tag and the tag's new owner Tag_{new}.

2) Forward privacy protection: After the ownership transfer completes, the tag's new owner Tag_{new} can't access the session information between the tag Tag and the tag's original owner Tag_{old}.

3) Mutual authentication: During the transfer process, the ownership transfer can be performed only after the mutual authentications are completed between the tag Tag and the tag's original owner Tag_{old}, and between the Tag Tag and the tag's new owner Tag_{new}.

4) Anti-asynchronous attack: The attacker interrupts the ownership transfer protocol by any attack mode, so that the information between any two no longer maintains the synchronization. The protocol should be able to guarantee the success of the tagged Tag authentication again and realize the resynchronization of the information.

5) Anti-replay attack: An attacker deliberately replays a certain type of intercepted message in an attempt to analyze the private information stored in the tag Tag. The protocol should ensure that the attacker replays the message without affecting the communication among the three entities, and the attacker can't crack any useful privacy information.

5 The Abstract Description of the Ownership Transfer Protocol

The ownership transfer protocol mainly solves the problem that what the ownership transfer protocol belongs to. In order to guarantee the security of the information stored in the tag, after the tag's owner is changed, the tag's ownership must be changed respectively.

In order to make the protocol not lose the generality, the abstract description of the protocol is given in this section firstly, and the implementation steps of the protocol will be given in the next section. The abstract description of *PSU-TOTP* protocol is shown in Figure 4.

The abstract description of the *PSU-TOTP* protocol is as follows.

Step 1: The tag owner Tag_{old} sends a $Query$ command to the tag Tag and initiates a transfer of ownership request.

Step 2: After the tag Tag receives the information, it first looks at the value of the flag $FLAG$, and the current $FLAG = 0$ indicates that the ownership belongs to the original tag owner Tag_{old} and can start the ownership transfer process. Then calculate the value of A, and then send the value of A to the tag owner Tag_{old}.

Step 3: After the original owner Tag_{old} of the tag receives the information, the authenticity of the tag

Tag is determined by verifying the authenticity of A. If true, the original tag owner Tag_{old} calculates the value of B and then sends the value of B to the tag Tag; otherwise, the protocol terminates.

Step 4: After the tag Tag receives the information, the authenticity of the original tag Tag_{old} is identified by verifying that B is true or false. If it is true, the tag Tag calculates the value of C and sends the value of C to the tag's new owner Tag_{new}; otherwise, the protocol terminates.

Step 5: After receiving the information, the tag's new owner Tag_{new} identifies the authenticity of the tag Tag by verifying the authenticity of C. If true, the tag's new owner Tag_{new} calculates the value of D and passes the value of D to the tag Tag; otherwise, the protocol terminates.

Step 6: After the tag Tag receives the information, the authenticity of the tag's new owner Tag_{new} is discriminated by verifying the authenticity of D. If it is true, the tag Tag sets the value of the flag $FLAG$ to 1, indicating that the ownership transfer is successful and the current ownership belongs to the tag new owner Tag_{new}; otherwise, the protocol terminates.

6 The Design of PSU-TOTP

The reader and the database communicate through a secure link, so this article will see the two as a whole, so there are three communication entities involved in the *PSU-TOTP*: The tag's original owner Tag_{old}, the tag's new owner Tag_{new}, the tag Tag.

6.1 The Symbol Description

The description of the communication entity symbols and operation symbols involved in the *PSU-TOTP* is shown in Table 1 below.

Table 1: Symbol description

Symbol	Description
Tag	tag
Tag_i	the i-th tag
Tag_{old}	the tag's original owner
Tag_{new}	the tag's new owner
ID_{t_i}	the i-th tag's identifier ID
$ID_{t_{i_L}}$	the left half of ID_{i_t}
$ID_{t_{i_R}}$	the right half of ID_{i_t}
$K_{i_{old}}$	the shared private key generated between Tag_{old} and Tag_i
$K_{i_{new}}$	the shared private key generated between Tag_{new} and Tag_i
R_{Tag}	the random number generated by the tag
$R_{Tag_{new}}$	the random number generated by Tag_{new}
$R_{Tag_{old}}$	the random number generated by Tag_{old}
$\oplus$	bitwise XOR operation
$\|\|$	bitwise concatenation operation
&	bitwise AND operation
$CroLink(X, Y)$	cross-link operation
$CroSyn(X, Y)$	cross-synthesis operation
$FLAG$	the flag of the tag's ownership

6.2 Initial Assumptions and Initialization Phase

In order to make the *PSU-TOTP* design not lose the generality, the *PSU-TOTP* design also makes the following assumptions:

1) The communication link between the tag Tag and the tag's new owner Tag_{new} is not secure;

2) The tag Tag and the communication link between the tag's original owner Tag_{old} is not secure;

3) The communication link between the tag's original owner Tag_{old} and the tag's new owner Tag_{new} is secure. The attacker can listen to the communication messages in 1) and 2), and the attacker can't listen to the communication messages in 3). At the same time, it is assumed that the information stored in the tag Tag, the tag's original owner Tag_{old}, and the tag's new owner Tag_{new} is safe and reliable, and the attacker cannot know it in advance.

Before the tag's ownership transfer starts, the tag's original owner Tag_{old} and tag's new owner Tag_{new} both store all the tags' identifiers, because they don't know which specific tag is to be transferred. After the protocol is initialized, the tag Tag stores the following four-tuple data structure:$(ID_{t_{i_L}}, ID_{t_{i_R}}, K_{i_{old}}, K_{i_{new}})$. The tag's new owner Tag_{new} stores the following three-tuple data structure: $(ID_{t_{i_L}}, ID_{t_{i_R}}, K_{i_{new}})$. The tag's original owner Tag_{old} stores the following three-tuple data structure: $(ID_{t_{i_L}}, ID_{t_{i_R}}, K_{i_{old}})$. The flag $FLAG$ has an initial value of 0. When $FLAG = 0$, the ownership of the tag currently belongs to the tag's original owner. When $FLAG = 1$, the ownership of the tag has been transferred and at this time, the ownership belongs to the tag.

6.3 The Protocol Description

The *PSU-TOTP* flow chart is shown in Figure 9. The following describes the specific meanings of the formulas of $M0$ to $M7$ in Figure 9 as shown in Table 2, and then a description of the specific steps of the *PSU-TOTP* is given in conjunction with Figure 9.

The *PSU-TOTP* flow chart is shown in Figure 5.

The detailed steps of the *PSU-TOTP* process are de-

Table 2: Formula Description

Symbol	Description
$M0$	$ID_{t_{i_R}} \oplus R_{Tag}$
$M1$	$CroLink(R_{Tag}, K_{i_{old}})$
$M2$	$R_{Tag_{old}}$
$M3$	$CroSyn(R_{Tag_{old}}, K_{i_{old}})$
$M4$	$R_{Tag} \oplus K_{i_{new}}$
$M5$	$CroLink(R_{Tag}, ID_{t_{i_R}})$
$M6$	$R_{Tag_{new}} \oplus ID_{t_{i_R}}$
$M7$	$CroSyn(R_{Tag_{new}}, K_{i_{new}})$

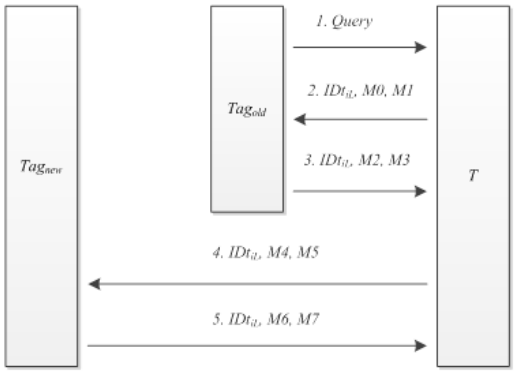


Figure 5: The flow chart of the ownership transfer

scribed below.

Step 1: The tag's original owner Tag_{old} sends a *Query* command to the tag Tag and initiates a transfer of ownership request.

Step 2: After the tag Tag receives the ownership transfer request, the value of the flag $FLAG$ is checked, and the current $FLAG$=0 indicates that the ownership belongs to the tag original owner Tag_{old}, and the ownership transfer can be started. Then the tag Tag generates a random number $R_{Tag} \in \{0,1\}^l$, and then calculates the values of $M0$, $M1$, and finally sends $M0$, $M1$, and $ID_{t_{i_L}}$ to the tag's original owner Tag_{old}.

Step 3: After the tag's original owner Tag_{old} receives the message, it first looks for $ID_{t_{i_L}}$ in the database. If $ID_{t_{i_L}}$ exists, Step 4 is performed; otherwise, the tag is forged by the attacker and *PSU-TOTP* terminates immediately.

Step 4: The tag's original owner Tag_{old} finds the $ID_{t_{i_R}}$ corresponding to $ID_{t_{i_L}}$, calculates the value of $ID_{t_{i_R}} \oplus M0$, then calculates the value of $M1$', and finally compares whether the values of $M1$' and $M1$ are equal.

If they are equal, the tag's original owner Tag_{old} correctly verifies the tag Tag, and Step 5 is performed; otherwise, the tag is false and the *PSU-TOTP* terminates immediately.

Besides, $M1' = CroLink(ID_{t_{i_R}} \oplus M0, K_{i_{old}})$.

Step 5: The tag's original owner Tag_{old} generates a random number $R_{Tag} \in \{0,1\}^l$, then calculates the values of $M2$, $M3$, and finally sends the $M2$, $M3$, and $ID_{t_{i_L}}$ to the tag Tag.

Step 6: After the tag Tag receives the information, it first calculates the value of $ID_{t_{i_R}} \oplus M2$, then calculates the value of $M3$', and finally compares whether the values of $M3$' and $M3$ are equal.

If they are equal, the tag Tag verifies that the tag's original owner Tag_{old} passes and proceeds to Step 7; otherwise, it indicates that the tag's original owner Tag_{old} is false and the *PSU-TOTP* terminates immediately.

Besides,$M3' = CroSyn(ID_{t_{i_R}} \oplus M2, K_{i_{old}})$.

Step 7: The tag Tag calculates the values of $M4$ and $M5$. Finally, $M4, M5$, and $ID_{t_{i_L}}$ are sent to the tag's new owner Tag_{new}.

Step 8: After receiving the information, the tag's new owner Tag_{new} first looks for $ID_{t_{i_L}}$ in the database. If $ID_{t_{i_L}}$ exists, Step 9 is performed; otherwise, the tag is forged by the attacker and the *PSU-TOTP* terminates immediately.

Step 9: The tag's new owner Tag_{new} finds the $ID_{t_{i_R}}$ corresponding to $ID_{t_{i_L}}$, calculates the value of $K_{i_{new}} \oplus M4$, then calculates the value of $M5$', and finally compares whether the values of $M5$' and $M5$ are equal.

If they are equal, the tag's new owner Tag_{new} correctly verifies the tag Tag and Step 10 is performed; otherwise, the tag is false and the *PSU-TOTP* terminates immediately.

Besides, $M5' = CroLink(K_{i_{new}} \oplus M4, ID_{t_{i_R}})$.

Step 10: The tag's new owner Tag_{new} generates a random number $R_{Tag_{new}} \in \{0,1\}^l$, then calculates the values of $M6$ and $M7$, and finally sends the $M6, M7$, and $ID_{t_{i_L}}$ to the tag Tag.

Step 11: After the tag Tag receives the information, it first calculates the value of $ID_{t_{i_R}} \oplus M6$, then calculates the value of $M7$', and finally compares whether the values of $M7$' and $M7$ are equal.

If they are equal, the tag Tag correctly verifies the tag's new owner Tag_{new} and Step 12 is performed; otherwise, it indicates that the tag's new owner Tag_{new} is false and the *PSU-TOTP* terminates immediately.

Besides, $M7' = CroSyn(ID_{t_{i_R}} \oplus M6, K_{i_{new}})$.

Step 12: The tag Tag sets the value of the flag $FLAG$ to 1, indicating that the ownership transfer is completed. At this time, the ownership of the tag is attributed to the tag's new owner Tag_{new}.

7 Security Analysis

7.1 Replay Attack

After the attacker listens on a complete communication session, all the communication messages can be obtained. The attacker tries to obtain the private information of the tag by replaying the message, but the attacker cannot succeed. In the communication message, each message in $M0$ to $M7$ is transmitted after encryption, not in plain text; and random numbers are used in the message encryption process. Random numbers are different each time, and at the same time they have unpredictability. Therefore, the attacker cannot replay messages for any private information.

7.2 Backward Privacy Protection

The new protocol can protect the privacy of the tag's new owner. After the ownership transfer is completed, the value of flag $FLAG$ in the tag is 1, is: $FLAG$=1, indicating that the ownership of the current tag belongs to the tag's new owner. In the process of ownership transfer, the value of $FLAG$ cannot be changed arbitrarily. The value of $FLAG$ will only change after the strict authentication and the correctness is determined. When the value of $FLAG$ changes, it indicates that the ownership transfer is completed; When $FLAG$=1, the tag's original owner sends a message to the tag for information access. At this time, the tag will recognize that the original owner of the current tag does not have the ownership of the tag according to the value of $FLAG$. Therefore, the access request of the tag's original owner will be rejected. So *PSU-TOTP* has backward privacy protection.

7.3 Forward Privacy Protection

The new protocol can protect the privacy of the tag's original owner from infringement. Before the transfer of ownership is completed, the value of the $FLAG$ is always 0, and it cannot be physically destroyed. If the value of $FLAG$ is not 1, the tag's new owner does not possess the ownership of the tag. If the tag's new owner sends a message to the tag, the tag will give an access deny response according to the value of $FLAG$. The tag's new owner has no right to access the tag, so the tag's new owner cannot obtain the communication message between the tag's original owner and the tag. Therefore, *PSU-TOTP* has forward privacy protection.

7.4 Bidirectional Authentication

Bidirectional authentication in *PSU-TOTP* refers to mutual authentication between the tag's original owner and the tag. It determines that the tag is the target tag of the transfer, and determines that the tag's original owner actually has the ownership of the target tag. Bidirectional authentication also refers to mutual authentication between the tag's new owner and tag. It determines that the tag is the target tag of the transfer, and determines that the tag's new owner is indeed the owner of the upcoming tag ownership.

Mutual authentication between the tag and the tag's original owner. The tag's original owner in the *PSU-TOTP* will confirm the authenticity of the tag for the first time in Step 3. Even if the attacker obtains the $ID_{t_{i_L}}$ through the interception method, the attacker still cannot pass the subsequent authentication. In Step 4, the tag's original owner will perform a second authentication on the tag. Since the attacker cannot obtain the $ID_{t_{i_R}}$ and $K_{i_{old}}$, the attacker cannot calculate the correct $M0$ and $M1$. The calculation can identify the authenticity of the tag. The authenticity of the tag to the tag's original owner is accomplished through Step 6: The attacker does not obtain the random number $R_{Tag_{old}}$ through the previous steps, and the attacker does not know the values of $ID_{t_{i_R}}$ and $K_{i_{old}}$; even the attacker can obtain the value of $ID_{t_{i_L}}$. However, the communication message does not use $ID_{t_{i_L}}$ in the calculation process, but uses $ID_{t_{i_R}}$. $ID_{t_{i_R}}$ does not have any relationship with $ID_{t_{i_L}}$.Therefore, the attacker cannot calculate the correct $M2$ and $M3$, so the tag can implement the tag's original owner's certification.

The certification between the tag and the tag's new owner. The tag's new owner's certification is completed in the Step 9. The random number R_{Tag} generated by the tag can be obtained by calculating the $M4$, which is then substituted into the $M5$ for comparison, and the authenticity of the tag can be identified according to the comparison result. Because the attacker does not know the values of $ID_{t_{i_R}}$, $K_{i_{new}}$, and R_{Tag}, the attacker cannot calculate the correct $M4$ and $M5$, thereby making it impossible for the attacker to pass the authentication in Step 9. The verification of the tag's new owner and the tag is performed in the Step 11. After receiving the $M6$ and $M7$, the tag first calculates the random number $R_{Tag_{new}}$ generated by the tag's new owner according to the $M6$, and then substitutes it into the $M7$ for verification; The user does not have the $ID_{t_{i_R}}, K_{i_{new}}$, or $R_{Tag_{new}}$ values, so the correct $M6$ and $M7$ values cannot be calculated.

In summary, *PSU-TOTP* enables bidirectional authentication between communicating entities.

7.5 Asynchronous Attack

Asynchronous attack, also known as desynchronization attack, refers to the attacker adopting some measure to make the shared private key between communication parties no longer maintain consistency. The attacker breaks

the consistency of shared private keys shared by both parties by using the following methods:

The parameters used by the two parties sharing the secret value update process are different;

The shared key is updated by one of the communication parties and the other party is not updated. In *PSU-TOTP*, no shared private key update mechanism is used, which makes it impossible for an attacker to use the above-mentioned method to destroy the shared private key between the two communication parties; the communication message is encrypted and then transmitted, and the communication message is calculated. Random numbers are useful in this process, and the random numbers are different each time, which assures that it is safe and reliable even if the shared private key is not updated. Therefore, the *PSU-TOTP* can resist the asynchronous attack.

7.6 Impersonation Attack

During the communication process, the attacker may fake the information exchange between any of the communication entities and other communication entities. Therefore, the protocol must be able to resist impersonation attacks by any of the attackers.

The attacker counterfeit the tag to communicate. When an attacker disguised as a tag to communicate, but the attacker does not know the following information: $ID_{t_{i_R}}, K_{i_{new}}, K_{i_{old}}$, so that the attacker can't calculate any of the correct value of $M0$ to $M7$. Even if the attacker has previously acquired all the messages of the previous round of communication by listening, and then replays the messages, the attacker still cannot obtain any private information because the attacker replays the message and the tag's original owner or the tag's new owner. A new random number will be generated and the new M_i value will be calculated at the same time, making the attacker's authentication fail. In the same way, the attacker would fake the tag's original owner to communicate or the counterfeit tag's new owner would fail to communicate, making it impossible to obtain any private information. Therefore, *PSU-TOTP* can resist the impersonation attack.

7.7 Brute Force Attack

The protocol must be able to resist the deliberately mandatory attack of the attacker, that is, the attacker uses a computing-intensive computer and cannot crack any useful private information.

By listening to a complete communication process, the attacker can obtain the following messages: $ID_{t_{i_L}}, Query, M0, M1, M2, M3, M4, M5, M6 and M7$. The attacker wants to use some of the useful information from the above information in the intercepted message, but the attacker cannot succeed. Here, messages $M0$ and $M1$ are selected as examples for analysis. In the formula $M0 = ID_{t_{i_R}} \oplus R_{Tag}$, the attacker only knows $M0$, and the two quantities of $ID_{t_{i_R}}$ and R_{Tag} are not known by the attacker, so the attacker cannot enumerate useful messages; meanwhile, the attacker is in the exhaustive process. As long as any one of $ID_{t_{i_R}}$ and R_{Tag} has an error, it is impossible for an attacker to obtain valid private information. In the formula $M1 = CroLink(ID_{t_{i_R}} \oplus M0, K_{i_{old}})$, even if the attacker substitutes M0 from the interception, the attacker can't exhaust any useful private information. First, the attacker does not know $ID_{t_{i_R}}$, $K_{i_{old}}$; second, the attacker does not know the details of each bit encryption in the cross-link encryption method, making it impossible for an attacker to violently crack private information. In the same way, the attacker analyzes and cracks $M2$, $M3$, $M4$, $M5$, $M6$ and $M7$, and cannot obtain private information. Therefore, the *PSU-TOTP* can resist the brute force attack.

Table 3 is a comparison of the security between PSU-TOTP and other RFID tag ownership transfer protocols.

8 GNY Logic Formal Proof

That the security of a complete protocol can be analyzed in words is far from enough. It can also be proved by the rigorous mathematical formulas. Based on this thought, in 1989 *Burrows* et al proposed a BAN formal logic analysis method, which was regarded as a milestone in the analysis of security protocols [2]. BAN logic is only concerned with the part of the protocol that is directly related to the authentication logic, and the rest is not a concern. It uses the rigorous mathematical rules to formalize the analysis and proof of the certification of the protocol. It also derives the target authentication step from the initialized hypothesis step of the protocol.

Because BAN form logic analysis has certain limitations, Gongli, *etc.* in 1990 put forward the GNY formal logic analysis method [7]. GNY formal logic analysis method is an expansion for BAN formal logic analysis method. GNY formal logic analysis method is more comprehensive than BAN logic analysis method, mainly in expanding the type and scope of analyzing the protocol. In this paper, the formal analysis and proof of *PSU-TOTP* protocol are carried out by using GNY formal logic analysis method.

1) Formal description of the protocol.

   To make the *PSU-TOTP* protocol easy to describe in the GNY formal logic language, the following convention is used: Tag_{old} indicates the tag's original owner, Tag indicates the tag, and Tag_{new} indicates the tag's new owner. The *PSU-TOTP* protocol flow is as follows:

   $Msg1$: $Tag_{old} \rightarrow Tag$: $Query$; indicates Tag receives message $\{Query\}$.

   $Msg2$: $Tag \rightarrow Tag_{old}$: $ID_{t_{i_L}}, M0 = ID_{t_{i_R}} \oplus R_{Tag}$, $M1 = CroLink(RTag, K_{i_{old}})$; indicates Tag_{old}

Table 3: Security comparison of authentication protocols

Attack Type	Reference[9]	Reference[11]	Reference[12]	Reference[13]	Reference[15]	Reference[16]	This Paper
Replay Attack	√	√	√	×	√	√	√
Backward Privacy Protection	√	√	×	√	√	√	√
Forward Privacy Protection	√	√	√	×	√	√	√
Bidirectional Authentication	√	√	√	√	√	√	√
Asynchronous Attack	√	×	×	×	√	√	√
Impersonation Attack	×	√	√	√	√	×	√
Brute Force Attack	√	√	√	√	×	√	√

Note: × means not provided; √ means provided

receives messages $\{M0, M1, ID_{t_{i_L}}\}$.

$Msg3$: $Tag_{old} \rightarrow Tag$: $ID_{t_{i_L}}$, $M2 = R_{Tag_{old}} \oplus ID_{t_{i_R}}$, $M3 = CroSyn(R_{Tag_{old}}, K_{i_{old}})$; indicates Tag receives messages $\{M2, M3, ID_{t_{i_L}}\}$.

$Msg4$: $Tag \rightarrow Tag_{new}$: $ID_{t_{i_L}}$, $M4 = R_{Tag} \oplus K_{i_{new}}$, $M5 = CroLink(R_{Tag}, ID_{t_{i_R}})$; indicates Tag_{new} receives messages $\{M4, M5, ID_{t_{i_L}}\}$.

$Msg5$: $Tag_{new} \rightarrow Tag$: $ID_{t_{i_L}}$, $M6 = R_{Tag_{new}} \oplus ID_{t_{i_R}}$, $M7 = CroSyn(R_{Tag_{new}}, K_{i_{new}})$; indicates Tag receives messages $\{M6, M7, ID_{t_{i_L}}\}$.

The above protocol is specified in the GNY formal logic language and can be described as follows:

$Msg1$: $Tag < * \, Query$.

$Msg2$: $Tag_{old} < * \{ID_{t_{i_L}}, M0 = ID_{t_{i_R}} \oplus R_{Tag}, M1 = CroLink(RTag, K_{i_{old}})\}$;

$Msg3$: $Tag < * \{ID_{t_{i_L}}, M2 = R_{Tag_{old}} \oplus ID_{t_{i_R}}, M3 = CroSyn(R_{Tag_{old}}, K_{i_{old}})\}$;

$Msg4$: $Tag_{new} < *\{ID_{t_{i_L}}, M4 = R_{Tag} \oplus K_{i_{new}}, M5 = CroLink(R_{Tag}, ID_{t_{i_R}})\}$;

$Msg5$: $Tag < *\{ID_{t_{i_L}}, M6 = R_{Tag_{new}} \oplus ID_{t_{i_R}}, M7 = CroSyn(R_{Tag_{new}}, K_{i_{new}})\}$.

2) Protocol initialization supposition.

The *PSU-TOTP* protocol is assumed to be as follows: Tag_{old}, Tag_{new}, and Tag indicate the main entities, that is, Tag_{old} indicates the tag's original owner, Tag indicates the tag, and Tag_{new} indicates the tag's new owner.

$Sub1$: $Tag \ni (ID_{t_{i_L}}, ID_{t_{i_R}}, K_{i_{old}}, K_{i_{new}}, R_{Tag})$; indicates Tag has the shared private key $K_{i_{old}}$, $K_{i_{new}}$, and has self-identifiers $ID_{t_{i_L}}$, $ID_{t_{i_R}}$ and self-generated random number R_{Tag}.

$Sub2$: $Tag_{old} \ni (ID_{t_{i_L}}, ID_{t_{i_R}}, K_{i_{old}}, R_{Tag_{old}})$; indicates Tag_{old} has the shared private key $K_{i_{old}}$, and has Tag's identifiers $ID_{t_{i_L}}$, $ID_{t_{i_R}}$ and self-generated random number $R_{Tag_{old}}$.

$Sup3$: $Tag_{new} \ni (ID_{t_{i_L}}, ID_{t_{i_R}}, K_{i_{new}}, R_{Tag_{new}})$; indicates Tag_{new} has the shared private key $K_{i_{new}}$, and has Tag's identifiers $ID_{t_{i_L}}$, $ID_{t_{i_R}}$ and self-generated random number $R_{Tag_{new}}$.

$Sup4$: $Tag_{old}| \equiv \#(R_{Tag_{new}}, R_{Tag_{old}}, R_{Tag})$; indicates Tag_{old} believes the random numbers $R_{Tag_{new}}$, $R_{Tag_{old}}$, R_{Tag} are fresh.

$Sup5$: $Tag| \equiv \#(R_{Tag_{new}}, R_{Tag_{old}}, R_{Tag})$; indicates Tag believes the random numbers $R_{Tag_{new}}$, $R_{Tag_{old}}$,R_{Tag} are fresh.

$Sup6$: $Tag_{new}| \equiv \#(R_{Tag_{new}}, R_{Tag_{old}}, R_{Tag})$; indicates Tag_{new} believes the random numbers $R_{Tag_{new}}$,$R_{Tag_{old}}$, R_{Tag} are fresh.

$Sup7$: $Tag| \equiv Tag_{new} \overset{\{ID_{t_{i_R}}, ID_{t_{i_L}}, K_{i_{new}}\}}{\leftrightarrow} Tag$; indicates that Tag believes the information $ID_{t_{i_L}}$, $ID_{t_{i_R}}$, $K_{i_{new}}$ shared between the Tag and Tag_{new}.

$Sup8$: $Tag| \equiv Tag_{old} \overset{\{ID_{t_{i_R}}, ID_{t_{i_L}}, K_{i_{old}}\}}{\leftrightarrow} Tag$; indicates that Tag believes the information $ID_{t_{i_L}}$, $ID_{t_{i_R}}$, $K_{i_{old}}$ shared between the Tag and Tag_{old}.

$Sup9$: $Tag_{old}| \equiv Tag \overset{\{ID_{t_{i_R}}, ID_{t_{i_L}}, K_{i_{old}}\}}{\leftrightarrow} Tag_{old}$; indicates that Tag_{old} believes the information $ID_{t_{i_L}}$, $ID_{t_{i_R}}$, $K_{i_{old}}$ shared between the Tag_{old} and Tag.

$Sup10$: $Tag_{new}| \equiv Tag \overset{\{ID_{t_{i_R}}, ID_{t_{i_L}}, K_{i_{new}}\}}{\leftrightarrow} Tag_{new}$; indicates that Tag_{new} believes the information $ID_{t_{i_L}}$, $ID_{t_{i_R}}$, $K_{i_{new}}$ shared between the Tag_{new} and Tag.

3) Protocol proof target.

The *PSU-TOTP* protocol has four targets for proof, which are mainly the trust in the freshness of mutual information exchange with the tag and the tag's new owner, and with the tag and the tag's original owner. The proof formulas of the goal are as follows:

$Goal1$: $Tag_{old}| \equiv Tag| \sim \#\{M0 = ID_{t_{i_R}} \oplus R_{Tag}, M1 = CroLink(R_{Tag}, K_{i_{old}})\}$;

$Goal2$: $Tag| \equiv Tag_{old}| \sim \#\{M2 = R_{Tag_{old}} \oplus ID_{t_{i_R}}, M3 = CroSyn(R_{Tag_{old}}, K_{i_{old}})\}$;

$Goal3$: $Tag_{new}| \equiv Tag| \sim \#\{M4 = R_{Tag} \oplus K_{i_{new}}, M5 = CroLink(R_{Tag}, ID_{t_{i_R}})\}$;

$Goal4$:$Tag| \equiv Tag_{new}| \sim \#\{M6 = R_{Tag_{new}} \oplus ID_{t_{i_R}}, M7 = CroSyn(R_{Tag_{new}}, K_{i_{new}})\}$;

4) Protocol proof process.

The proof of the *PSU-TOTP* protocol is based on the initial supposition. The proof process follows the logical reasoning rules, being-told rules, freshness rules and possession rules in Reference [24]. The message interpretation rules follow the written form of the GNY logical reasoning rule in Reference [24], which are represented by T, P, F, I respectively.

Since the proof process of $Goal2$: $Tag| \equiv Tag_{old}| \sim \#\{M2 = R_{Tag_{old}} \oplus ID_{t_{i_R}}, M3 = CroSyn(R_{Tag_{old}}, K_{i_{old}})\}$; $Goal3$: $Tag_{new}| \equiv Tag| \sim \#\{M4 = R_{Tag} \oplus K_{i_{new}}, M5 = CroLink(R_{Tag}, ID_{t_{i_R}})\}$, $Goal4$: $Tag| \equiv Tag_{new}| \sim \#\{M6 = R_{Tag_{new}} \oplus ID_{t_{i_R}}, M7 = CroSyn(R_{Tag_{new}}, K_{i_{new}})\}$ is similar to the proof process of $Goal1$: $Tag_{old}| \equiv Tag| \sim \#\{M0 = ID_{t_{i_R}} \oplus R_{Tag}, M1 = CroLink(R_{Tag}, K_{i_{old}})\}$.Therefore, Goal1 is used as an example in this section. The proof process is described as follows.

$\because$ $RuleP1$: $\frac{P<X}{P \ni X}$ $and Msg2 : Tag_{old} < * \{ID_{t_{i_L}}, M0 = ID_{t_{i_R}} \oplus R_{Tag}, M1 = CroLink(R_{Tag}, K_{i_{old}})\}$;

$\therefore$ $Tag_{old} \ni \{M0 = ID_{t_{i_R}} \oplus R_{Tag}, M1 = CroLink(R_{Tag}, K_{i_{old}}\}$.

$\because$ $Rule1F1$: $\frac{P|\equiv(X)}{P|\equiv(x,y), P|\equiv \#F(X)}$ and $Sup5$: $Tag| \equiv \#(R_{Tag_{new}}, R_{Tag_{old}}, R_{Tag})$;

$\therefore$ $Tag_{old} = \#\{M0 = ID_{t_{i_R}} \oplus RTag, M1 = CroLink(R_{Tag}, K_{i_{old}})\}$.

$\because$ $RuleP2$: $\frac{P \ni X, P \ni Y}{P \ni (X,Y), P \ni F(X,Y)}$, $Sup1$: $Tag \ni (ID_{t_{i_L}}, ID_{t_{i_R}}, K_{i_{old}}, K_{i_{new}}, R_{Tag})$ and $Sup2$: $Tag_{old} \ni (ID_{t_{i_l}}, ID_{t_{i_R}}, K_{i_{old}}, R_{Tag_{old}})$;

$\therefore$ $Tag_{old} \ni \{M0 = ID_{t_{i_R}} \oplus R_{Tag}, M1 = CroLink(R_{Tag}, K_{i_{old}})\}$.

$\because$ $RuleF10$: $\frac{P|\equiv(X), P \ni X}{P|\equiv \#(H(X))}$ and the derived formula $Tag_{old} = \#\{M0 = ID_{t_{i_R}} \oplus R_{Tag}, M1 = CroLink(R_{Tag}, K_{i_{old}})\}$, $Tag_{old} \ni \{M0 = ID_{t_{i_R}} \oplus R_{Tag}, M1 = CroLink(R_{Tag}, K_{i_{old}})\}$;

$\therefore$ $Tag_{old}| \equiv \#\{M0 = ID_{t_{i_R}} \oplus R_{Tag}, M1 = CroLink(R_{Tag}, K_{i_{old}})\}$.

$\because$ Rule $I3$: $\frac{P<H(X,<S>)>, P \ni (X,S), P|\equiv P \leftrightarrow Q, P|\equiv \#(X,S)}{P|\equiv Q|\sim(X,S), P|\equiv Q \sim H(X,<S>)}$;

$\because$ $Sup8$: $Tag| \equiv Tag_{old} \overset{ID_{t_{i_R}}, ID_{t_{i_L}}, K_{i_{old}}}{\leftrightarrow} Tag$, $Sup9$: $Tag_{old}| \equiv Tag \overset{ID_{t_{i_R}}, ID_{t_{i_L}}, K_{i_{old}}}{\leftrightarrow} Tag_{old}$ and $Msg2$: $Tag_{old} < *\{ID_{t_{i_L}}, M0 = ID_{t_{i_R}} \oplus R_{Tag}, M1 = CroLink(R_{Tag}, K_{i_{old}})\}$;

$\therefore$ $Tag_{old}| \equiv Tag \sim \{M0 = ID_{t_{i_R}} \oplus R_{Tag}, M1 = CroLink(R_{Tag}, K_{i_{old}})\}$.

$\because$ The definition of freshness and the derived formula $Tag_{old} = \#\{M0 = ID_{t_{i_R}} \oplus R_{Tag}, M1 = CroLink(R_{Tag}, K_{i_{old}})\}, Tag_{old} \mid = Tag \sim \{M0 = ID_{t_{i_R}} \oplus R_{Tag}, M1 = CroLink(R_{Tag}, K_{i_{old}})\}$;

$\therefore$ $Goal1$: $Tag_{old}| \equiv Tag| \sim \#\{M0 = ID_{t_{i_R}} \oplus R_{Tag}, M1 = CroLink(R_{Tag}, K_{i_{old}})\}$ is proved.

9 Performance Analysis

In the process of the tag ownership transfer, there are three communication entities involved: The tag, the original owner of the tag, and the new owner of the tag. The original owner of the tag and the new owner of the tag all include the database. Therefore, the two parts of the communication entities have powerful query capabilities, data calculation capabilities, and storage space. The tag does not have the above capabilities, there-by, the performance analysis will focus on three aspects of the tag calculation, storage space, and session times. Table 4 shows the performance comparison between *PSU-TOTP* and other RFID tag ownership transfer protocols.

Table 4: Performance comparison of ownership transfer protocol

Ref.	Calculation	Storage	Session Times
Ref[9]	5P+H	$2l$	5
Ref[11]	13P+6H	$1l$	7
Ref[12]	2P+7H	$3l$	6
Ref[13]	6P+M	$3l$	5
Ref[15]	3P+H+M	$3l$	5
Ref[16]	3H+3M	$4l$	6
This Paper	4P+2N+2Q	$3l$	5

In Table 4, H represents a hash function operation. P represents a bitwise operation. M represents a modular square operation. Q represents a cross-synthesis operation. N represents a cross-link operation.

1) The tag calculation. Compared to other references, the *PSU-TOTP* in this paper does not encrypt the information by using a large computational hash function or a modular squaring operation. Instead, it selects ultra-lightweight bitwise operation to encrypt its transmission information, which can greatly reduce the amount of computation on the tag. The computational complexity of the tag in this paper differs from other references by more than one order of magnitude, which can greatly reduce the computational cost of the tag. At the same time, the bitwise AND operation and the bitwise XOR operation are also used in the cross-synthesis operation and the cross-link operation, so that some circuits can be shared among the four operations. The cost of the tag will also be reduced.

2) The storage space on the tag. Set the ID_t, $K_{i_{new}}$ and $K_{i_{old}}$ with the length of l bits, so the storage space

on the tag only needs $3l$ bits. Compared with other references, $3l$ storage space has been improved. In the protocol of this paper, only one random number is generated at the tag. In other references, multiple random numbers are generated at the tag. Therefore, the overall cost of the storage space in this paper is not too large and it is acceptable.

3) Session times. Relative to references [16, 17, 27], the protocol in this paper reduces the times of session, which can reduce the cost of communication time of the entire protocol. Although the times of session in this paper are equivalent to the references [1, 6, 11], this protocol can make up for the security flaws existing in other protocols.

To sum up, the protocol in this paper can effectively reduce the tag calculation and it's much improved compared to other protocols. In terms of the storage space and the session times, the protocol in this paper is not improved much, but it can make up for the security flaws existing in other protocols, so this protocol still has some advantages, which is suitable for low-cost RFID systems.

10 Conclusions

In the life cycle of RFID tag, ownership often changes. In order to ensure the security of the privacy of the tag, an ultra-lightweight RFID tag ownership transfer protocol *PSU-TOTP* based on bitwise operation is proposed. Based on the analysis of the deficiencies in existing protocols, the paper proposes an improved protocol *PSU-TOTP*. It introduces cross-synthesis operation and cross-link operation to encrypt the transmission information so that the protocol can achieve ultra-lightweight levels. At the same time, the use of bitwise operations described above can effectively reduce the tag calculation and reduce the cost of tag. According to the different values of flag $FLAG$, the corresponding operation is performed to ensure the uniqueness and definiteness of the ownership. The security analysis shows that the $PSU-TOTP$ can meet the requirements for the ownership transfer. GNY formal logic proves the accuracy of $PSU-TOTP$. Comprehensive performance analysis shows the advantages of *PSU-TOTP* and achieve the goal of reducing tag calculation, so it is suitable for low-cost RFID systems. The next research directions of the paper are: To optimize the *PSU-TOTP* protocol in order to reasonably reduce the communication traffic; To implement the prototype of the *PSU-TOTP* RFID system and figure out the total number of required gate circuits and the time of a complete communication, so as to combine the theory with practice.

References

[1] Y. Bian-qing and L. Ji-qiang, "Provable secure ownership transship protocol for RFID tag," *Journal of Communications*, vol. 36, no. 8, pp. 83–90, 2015.

[2] M. Borrows, M. Abadi and R. M. Needham, "A logic of authentication," in *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, pp. 233–271, 1989.

[3] J. S. Chen, C. Y. Yang, and M. S. Hwang, "The capacity analysis in the secure cooperative communication system," *International Journal of Network Security*, vol. 19, no. 6, pp. 863–869, 2017.

[4] Y. C. Chen, W. L. Wang, M. S. Hwang, "RFID authentication protocol for anti-counterfeiting and privacy protection", in *The 9th International Conference on Advanced Communication Technology*, pp. 255–259, 2007.

[5] K. Elkhiyaout, E. O. Blass and R. Molva, "ROTIV: RFID ownership transfer with issuer verification[A]," in *LNCS7055: 7th International Workshop on RFID Security and Privacy*, pp. 163–182, 2012.

[6] S. Fouladgar and H. Afifi, "An efficient delegation and transfer of ownership protocol for RFID tags[C]," in *Proc of the 1st Int EURASIP Workshop on RFID Technology*, pp. 10–14, 2007.

[7] L. Gong, R. Needham, and R. Yanhalom, "Reasoning about belief in cryptographic protocols," *IEEE Computer Society Symposium in Security and Privacy*, pp. 234–248, 1990.

[8] M. S. Hwang, T. H. Sun, and C. C. Lee, "Achieving dynamic data guarantee and data confidentiality of public auditing in cloud storage service," *Journal of Circuits Systems and Computers*, vol. 26, no. 5, 2017.

[9] M. S. Hwang, W. Y. Chao, C. Y. Tsai, "An improved key management scheme for hierarchical access control," *International Journal of Network Security*, vol. 19, no. 4, pp. 639–643, 2017.

[10] P. Jappinen and H. Hamalainen, "Enhanced RFID security method with ownership transfer[C]," in *Proc of Int Conf on Computational Intelligence and Security*, pp. 382–385, 2008.

[11] Y. M. Jin, H. P. Sum and Z. Guan, "Ownership transfer protocol for RFID tag," *Journal of Computer Research and Development*, vol. 48, no. 8, pp. 1400-1405, 2011.

[12] D. W. Liu, J. Ling, X. Yang, "An improved RFID authentication protocol with backward privacy," *Computer Science*, vol. 43, no. 8, pp. 128–130, 2016.

[13] D. Molnar, A. Soppera and D. Wagner, "A scalable, delegatable pseudonym protocol enabling ownership transfer of rfid tags[A]," in *12th International Workshop on Selected Areas in Cryptography[C]*, pp. 276–290, 2006.

[14] K. Osaka, T. Takagi and K. Yanazaki, "An efficient and secure RFID security method with ownership transfer[A]," in *Proceedings of IEEE International Conference on Computational Intelligence and Security[C]*, pp. 1090–1095, 2006.

[15] H. T. Pan, C. S. Pan, S. C. Tsaur, and M. S. Hwang, "Cryptanalysis of efficient dynamic id based remote

user authentication scheme in multi-server environment using smart card," in *12th International Conference on Computational Intelligence and Security (CIS'16)*, pp. 590–593, Dec. 2017.

[16] B. Song, *RFID Tag Ownership Transfer [EB/OL]*, 2008. (`http://rfidsec2013.iaik.tugraz.at/RFIDSec08/Papers/Publication/15%20-%20Song%20-%20Ownership%20Transfer%20-%20Paper.pdf`)

[17] B. Song, and C. J. Mitchell, "Scalable RFID security protocols supporting tag ownership transfer," *Computer Communications*, vol. 34, no. 4, pp. 556–566, 2011.

[18] C. Y. Tsai, C. Y. Liu, S. C. Tsaur, and M. S. Hwang, "A publicly verifiable authenticated encryption scheme based on factoring and discrete logarithms," *International Journal of Network Security*, vol. 19, no. 3, pp. 443–448, 2017.

[19] Y. L. Wang, J. J. Shen, and M. S. Hwang, "An improved dual image-based reversible hiding technique using LSB matching," *International Journal of Network Security*, vol. 19, no. 5, pp. 858–862, 2017.

[20] C. H. Wei, M. S. Hwang, and A. Y. H. Chin, "A secure privacy and authentication protocol for passive RFID tags," *International Journal of Mobile Communications*, vol. 15, no. 3, pp. 266–277, 2017.

[21] C. H. Wei, M. S. Hwang, Augustin Y. H. Chin, "An improved authentication protocol for mobile agent device in RFID," *International Journal of Mobile Communications*, vol. 10, no. 5, pp. 508–520, 2012.

[22] C. H. Wei, M. S. Hwang, A. Y. H. Chin, "A mutual authentication protocol for RFID", *IEEE IT Professional*, vol. 13, no. 2, pp. 20–24, Mar. 2011.

[23] C. H. Wei, M. S. Hwang, Augustin Y. H. Chin, "An authentication protocol for low-cost RFID tags", *International Journal of Mobile Communications*, vol. 9, no. 2, pp. 208–223, 2011.

[24] N. I. Wu and M. S. Hwang, "Development of a data hiding scheme based on combination theory for lowering the visual noise in binary images," *Displays*, vol. 49, pp. 116–123, 2017.

[25] R. Xie, J. Ling, and D. W. Liu, "Wireless key generation algorithm for rfid system based on bit operation," *International Journal of Network Security*, vol. 20, no. 5, pp. 938–950, 2018.

[26] R. Xie, B. y. Jian, and D. w. Liu, "An improved ownership transfer for RFID protocol," *International Journal of Network Security*, vol. 20, no. 1, pp. 149–156, 2018.

[27] C. Xiuqing, C. Tianjie, and Z. Jingxuan, "Provable secure ownership transship protocol for RFID tag," *Journal of Electronics & Information Technology*, vol. 36, no. 8, pp. 83–90, 2015.

[28] C. Y. Yang, T. Y. Chung, M. S. Hwang, C. Y. Li, and J. F. J. Yao, "Learning performance evaluation in elearning with the web-based assessment," in *8th iCatse Conference on Information Science and Applications (ICISA'17), Lecture Notes in Electrical Engineering*, pp. 645–651, Mar. 2017.

Biography

Jia-qi Wang ,received her Master's degree in computer science from Sun Yat-sen University (China) in December 2012. She is a lecturer in Information Engineering Institute in Zhujiang College of South China Agricultural University, director of the Teaching and Research Office of electronic business and information management. Her current research interest fields include Electronic Business and computer applications.

Yun-feng Zhang, from 2009 to 2011 he was an expert in RFID tag in The Automation Society of Guangdong Province. Now he is a teacher in Continuing Education College of Guangzhou City Construction College, and his main research interest is information security.

Dao-wei Liu received a master's degree in School of Computers from Guangdong University of Technology (China) in June 2016. He is a teacher in department of computer science and engineering, Guangzhou College of Technology and Business. His current research interest fields include information security.

Anti-Leakage Client-Side Deduplication with Ownership Management in Fog Computing

Hua Ma, Guo-Hua Tian, and Lin-Chao Zhang
(Corresponding author: Guo-Hua Tian)
School of Mathematics and Statistics, Xidian University
Xi'an 710126, China
(Email: gh_tian0621@163.com)
(Received May 16, 2018; Revised and Accepted Oct. 18, 2018; First Online Sept. 21, 2019)

Abstract

In commercial fog computing, block-level client-side deduplication (BC-Dedu) can be used to save storage space and network bandwidth. However, the existing BC-Dedu schemes cannot support ownership management, which leads to the degradation of forward and backward secrecy of the outsourced data. Besides, BC-Dedu schemes are vulnerable to the side information leakage issue since the existence of data is revealed to the outside adversary. In this paper, we propose an anti-leakage BC-Dedu scheme that supports ownership management in fog computing. Specifically, we present a dual-level ownership list and key update mechanism to achieve ownership management in the proposed scheme. Besides, we construct a novel deduplication protocol to alleviate the side information leakage issue. Furthermore, a dynamic data storage strategy is proposed to reduce service costs and latency. Security and performance analyses demonstrate that the proposed scheme achieves the desired security requirements while saving resource efficiently.

Keywords: Deduplication; Dynamic Data Storage; Fog Computing; Ownership Management

1 Introduction

The ever-increasing volume of users in cloud computing results in an indisputable predicament that the computing power and storage capacity of the centralized cloud will be unable to provide satisfactory services for users timely in the near future [22]. To overcome this problem, fog computing is presented by Cisco in 2012 [2], which is a hierarchical service structure consists of the central cloud, fog devices, and end-users. The fog device connects to the central cloud and other fog devices through *inter-network*, and connects to end users through *intra-network* [16]. In this architecture, the central cloud offers a wide range of low-latency computing services by using fog devices adjacent to the users. Even so, the fog computing still faces the challenge of insufficient storage resources and network bandwidth caused by the growing volume of the outsourced data. Thus, researchers try to adopt cross-user deduplication technique in fog computing to save storage space and network bandwidth [10, 11], where each data is stored only once, and the subsequent versions are deleted. All of the legitimate users access the outsourced data through a link. Besides, the central cloud maintains the deduplicated data after fog devices implement deduplication over the outsourced data. This work model provides a rapid data deduplication while reducing the pressure on the central cloud.

In commercial fog computing, the service providers are most concerned about maximizing profits while ensuring system security. A feasible method is to reduce the service costs as much as possible. Considering space savings, the block-level deduplication performs more excellent in space savings than file-level one since the identical blocks of the different file can also be deduplicated in block-level deduplication. As regards bandwidth consumption, the client-side deduplication performs more excellent than the server-side one in bandwidth savings since only the unduplicated data is required to upload to the cloud server in client-side deduplication. The BC-Dedu is no doubt the best choice for service providers. However, there are some security issues remain to be solved [18].

Primarily, the data encryption key is rarely updated after its generation [15] and the data users may remove their data from fog storage to reduce service expenses. In this case, the more frequent the data ownership change is, the greater the impact on the key information disclosure. To alleviate this issue, the deduplication scheme should prevent revoked users from accessing the plaintext of outsourced data (forward secrecy). Likewise, the unauthorized user should be deterred to access the plaintext of outsourced data before she/he obtains the valid ownership (backward secrecy) [7]. Unfortunately, most of the existing block-level deduplication schemes [3, 23] do not consider the ownership management, and the existing ownership management techniques are only suitable for file-level deduplication. Therefore, it is a significant re-

search to realize the ownership management in block-level deduplication for better security and great space savings.

Besides, the client-side deduplication is vulnerable to the side information leakage issue, which means that the malicious adversary can learn the existence of outsourced data during the upload phase by analyzing the respond of the server, namely confirmation-of-file (CoF) attack [17]. This issue is an obstinate security flaw for file-level client-side deduplication, and most of the existing BC-Dedu schemes cannot alleviate this issue efficiently. Thus, the side information leakage issue is an opening security flaw that is worth exploring.

In this paper, we propose a BC-Dedu scheme over encrypted data in fog computing. Our main contributions are listed as follows:

- We propose a dual-level ownership list and key update mechanism to implement ownership management in the proposed BC-Dedu scheme.
- We construct a novel block-level deduplication protocol to alleviate the side information leakage issue, and this protocol also alleviates the security caused by duplicate-faking attack.
- To reduce the costs and latency of data service, we provide a dynamic data storage strategy to achieve efficient resource utilization by storing data according to service demand.

2 Related Works

Although most of the existing deduplication schemes [1, 3, 7, 9, 19, 21, 23] are presented in cloud computing rather than fog computing, the related experience is worth learning. Thus, we will introduce some representative works.

2.1 Secure Client-Side Deduplication

In the original architecture of deduplication, the cloud server deduplicates the plaintext uploaded by users [5, 6], which reveals the privacy of data users to the cloud server. For better privacy, the users encrypt their data before uploading it. Unfortunately, the general cryptographic primitives obstruct the deduplication since the different users will obtain the various ciphertext by encrypting the identical data with distinct encryption keys. To realize the deduplication over encrypted data, Douceur *et al.* [4] proposed a promising solution called Convergent Encryption (CE) that requires different users use the hash value of the data to encrypt the data. As an extension of CE, Bellare *et al.* [1] presented Message-Locked Encryption (MLE) and gave a formal privacy model PRV$-CDA and strict proof for CE. Recently, Chen *et al.* [3] extended MLE to block-level for secure large file deduplication and introduced the corresponding security model PRV$-CDA-B. Zhao *et al.* [23] presented a variant of block-level MLE [3] named updatable block-level MLE that supports the block-level data update.

Considering the side information leakage issue caused by CoF attack in client-side deduplication, Harnik *et al.* [6] proposed a scheme that resists CoF attack by utilizing the client-side and server-side deduplication alternatively according to a random threshold. Based on this work, Lee *et al.* [13] minimize the success probability of CoF attack through optimizing the security parameters to enhance the security of outsourced data. Koo *et al.* [11] proposed a hybrid deduplication protocol in conjunction with client-side and server-side deduplication to alleviate the side information leakage issue in fog computing.

To prevent the malicious users who never own the target data from acquiring valid ownership with a single hash value obtained by eavesdropping, Halevi *et al.* [5] proposed an interactive Proof of Ownership (PoW) protocol based on the Merkle Hash tree (MHT). Xu *et al.* [20] proposed a client-side deduplication scheme based on MHT [5]. However, this scheme is subject to file proof reply attack. Recently, Yang *et al.* [21] proposed a provable method of ownership proof of encrypted data, which can resist the file proof reply attack. Nevertheless, the encryption key is easily leaked. Li *et al.* [14] proposed a significant application of MHT-based PoW in deduplication and auditing scenario, which resists the file proof reply attack and realizes efficient verification by allowing multiple data blocks to be challenged simultaneously. Even so, the malicious user who owns the data may launch a duplicate-faking attack (DFA) by identifying the data and uploads a poison version in the initial upload phase. Then, the subsequent uploader only can access the poison data after uploading the data. Kutylowski *et al.* [12] proposed a novel deduplication scheme, called TrDup, which can trace the malicious user by incorporating traceable signatures with MLE. Kim *et al.* [9] proposed a novel client-side deduplication scheme to prevent data users from losing data under duplicate-faking attack by using a double-tag interaction model, where the second tag is generated by the cloud server in the initial upload phase.

2.2 Ownership Management

Considering the forward and backward secrecy of the outsourced data, Hur *et al.* [7] proposed an ownership management technique that adopts a key-encryption keys tree to realize efficient ownership management. Based on this work, Jiang *et al.* [8] presented a lazy update strategy to reduce the frequency of update. However, the default user-space of this technique does not necessarily satisfy the actual demand, especially when the number of data owners changes dramatically. Besides, this technique will lead to large costs if it is employed in block-level deduplication. Koo *et al.* [10] proposed a novel ownership management technique in file-level deduplication, which achieves the fine-grained access control efficiently without the consideration of default user-space. Unfortunately, it only can be used in file-level deduplication since its block encryption key implies the information of the file.

3 Preliminary

In this section, we introduce some necessary preliminaries for the proposed deduplication scheme.

3.1 Discrete Logarithm Problem (DLP)

For a group $\mathbb{G}$ with prime order p and generator g, given $g^a \in \mathbb{G}$, where $a \in \mathbb{Z}_p$, there is no polynomial time algorithm can compute a with non-negligible probability.

3.2 Notations

We denote the empty string as ε, and let $[i] = \{1, \ldots, i\}$ for $i \in \mathbb{N}$. If x is a vector, we denote $|\mathrm{x}|$ as its dimension, and denote $\mathrm{x}[i]$ as the i-th component of x, and define that $\mathrm{x}[i, j] = \mathrm{x}[i] \ldots \mathrm{x}[j]$ for $1 \le i \le j \le |\mathrm{x}|$. For a finite set S, $|S|$ denotes its size and $s \xleftarrow{r} S$ represents that an element s is uniformly selected in S. An operation that employs the algorithm $\mathcal{A}$ on inputs $x_1, \ldots$ randomly is denoted as $y \xleftarrow{r} \mathcal{A}(x_1, \ldots)$. We define the guessing probability $\mathrm{GP}(X)$ and min-entropy $\mathrm{H}_\infty(X)$ of a random variable X as $\max_x \Pr[X = x] = 2^{-\mathrm{H}_\infty(X)}$. Besides, for a random variable X given a random variable Y, we denote its conditional guessing probability $\mathrm{GP}(X|Y)$ and conditional min-entropy $\mathrm{H}_\infty(X|Y)$ as $\sum_y \Pr[Y = y] \cdot \max_x \Pr[X = x|Y = y] = 2^{-\mathrm{H}_\infty(X|Y)}$.

3.3 Unpredictable Sources

Suppose that a polynomial-time algorithm is a source $\mathcal{M}$, which takes 1^λ as input, outputs $(\mathbf{M}, Z)$, where $\mathbf{M}$ is a message vector in $\{0,1\}^*$ and $Z \in \{0,1\}^*$ denotes some auxiliary information of $\mathbf{M}$. We denote the length of vector $\mathbf{M}$ as $n(\lambda)$, which represents the number of blocks in our context. Besides, we label the i-th block of the message $\mathbf{M}$ with $\mathbf{M}[i]$ for all $i \in [1, n(\lambda)]$. Due to the block-level deduplication architecture of the proposed scheme, we suppose the sources output message vector over $\{0,1\}^B$, where B is the size of data block. Thus, $\mathbf{M}[i] \in \{0,1\}^B$ for all i. Furthermore, we require that $\mathbf{M}[i_1] \neq \mathbf{M}[i_2]$ for all distinct $i_1, i_2 \in [n(\lambda)]$ to bar against trivial adversary. A source $\mathcal{M}$ is unpredictable if $\mathrm{GP}_\mathcal{M} = \max_i\{\mathrm{GP}(\mathbf{M}[i]|Z)\}$ is negligible.

3.4 Block-Level MLE

According to Chen et al's works [3], a block-level MLE scheme is composed of the following algorithms:

- Setup: inputs the security parameter 1^λ and returns the system parameters P.
- KeyGen: Inputs the system parameters P and a file $M = M[1]\| \ldots \|M[n]$, runs the following two sub-algorithms and outputs a master key k_{mas} and block keys $\{k_i\}_{1 \le i \le n}$, respectively.
  1) M-KeyGen: Takes P and M as input, returns the master key k_{mas}.
  2) B-KeyGen: Takes P and $M[i]$ as input, returns the block key k_i.
- Enc: Inputs P, a block $M[i]$ and corresponding block key k_i, outputs the block ciphertext $C[i]$.
- Dec: Inputs P, a block ciphertext $C[i]$ and block key k_i, outputs the block $M[i]$ or $\perp$.
- TagGen: Inputs P and a file M, runs the following sub-algorithms and outputs file tag t and block tags $\{T_i\}_{1 \le i \le n}$ respectively.
  1) M-TagGen: Inputs P and M, outputs the file tag t.
  2) B-TagGen: Takes P and a block $M[i]$ as input, returns the block tag T_i.
- PoWPrf: Inputs the challenge $\mathcal{Q}$ and a file M, outputs a response $\mathcal{P}$
- PoWVer: Inputs the challenge $\mathcal{Q}$, the file tag t, the block tags $T_{i1 \le i \le n}$,and the response $\mathcal{P}$, outputs True or False.

3.5 PRV$-CDA-B Game

Based on the architecture of block-level MLE, Chen *et al.* [3] introduced a privacy model for block-level MLE scheme, called PRV$-CDA-B, which argues that a block-level MLE scheme is secure under chosen distribution attacks if no polynomial-time adversary $\mathcal{A}$ can win the following chosen distribution attack game PRV$-CDA-B with non-negligible advantage:

Setup: An adversary $\mathcal{A}$ sends the challenger $\mathcal{C}$ the description of an unpredictable block-source $\mathcal{M}$, and $\mathcal{C}$ generates and returns the system parameter P to $\mathcal{A}$.

Challenge: $\mathcal{C}$ selects $b \leftarrow \{0,1\}$ randomly. If $b = 0$, $\mathcal{C}$ runs $\mathcal{M}$ as $(\mathbf{M}^0, Z) \leftarrow \mathcal{M}(\lambda)$. Otherwise, $\mathcal{C}$ chooses $\mathbf{M}^1$ from $\{0,1\}^{|\mathbf{M}^0|}$ uniformly and randomly, and set $\mathbf{M} = \mathbf{M}^b$. We denote n as the number of blocks. For each $i \in [1, n]$, $\mathcal{C}$ computes: block keys $k_i \leftarrow \mathrm{B-KeyGen}(\mathbf{M}_i)$, ciphertexts $C_i \leftarrow \mathrm{B-Enc}(k_i, \mathbf{M}_i)$, and block tags: $T_i \leftarrow \mathrm{B-TagGen}(C_i)$, as well as the file tags: $t \leftarrow \mathbf{M-TagGen}(\mathbf{M})$. Finally, $\mathcal{C}$ returns auxiliary information Z, tags $T = \{t, T_1, \ldots, T_n\}$, and the ciphertexts $C = \{C_1, \ldots, C_n\}$ to $\mathcal{A}$.

Output: The adversary $\mathcal{A}$ outputs his guess b' according to (C, T, Z). If $b' = b$, then $\mathcal{A}$ wins the game.

We regard $\mathcal{A}$ as a PRV$-CDA-B adversary and define the advantage of $\mathcal{A}$ by

$$\mathrm{Adv}^{\mathcal{A},\mathcal{M}}_{\mathrm{PRV\$-CDA-B}}(\lambda) = \left| \Pr[b = b'] - \frac{1}{2} \right|.$$

Definition 1. *A block-level MLE scheme is PRV$-CDA-B-secure if for any $\mathcal{M}$ and any PRV$-CDA-B adversary $\mathcal{A}$, $\mathrm{Adv}^{\mathcal{A},\mathcal{M}}_{\mathrm{PRV\$-CDA-B}}$ is negligible.*

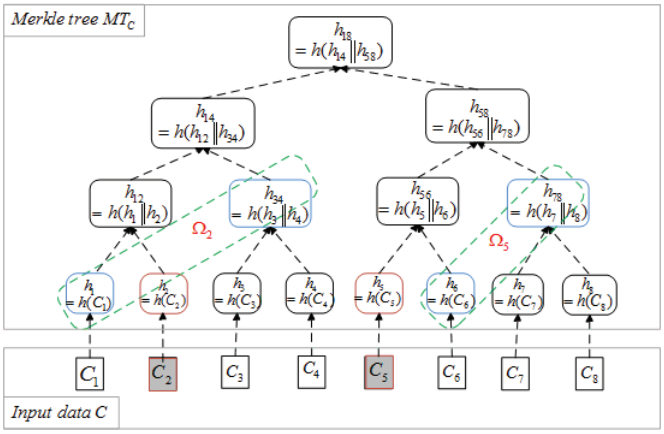


Figure 1: A Merkle Hash tree for input data with 8 blocks

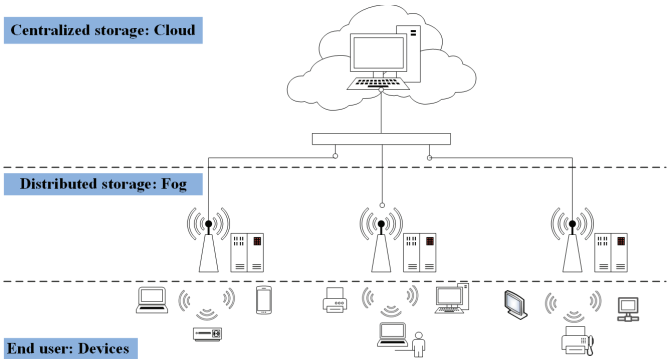


Figure 2: Fog storage system

3.6 Proof of Ownership

A demonstration of Li *et al.*'s [14] PoW protocol: suppose that a file C contains eight blocks. As is shown in Figure 1, the storage server constructs the MHT of C and triggers PoW process with a random challenge set $I_c = \{2, 5\}$. The prover computes h_2, h_5 of C_2, C_5 and corresponding auxiliary information $\Omega_2 = (h_1, h_{34})$, $\Omega_5 = (h_6, h_{78})$ (highlighted by green) as the ownership proof, and returns to storage server. The storage server reconstructs the root node of the MHT to verify proof:

$$h'_{18} = h(h(h(h_1 \parallel h_2) \parallel h_{34}) \parallel h(h(h_5 \parallel h_6) \parallel h_{78})).$$

If the proof is accepted, the prover is authorized to access this stored file. We employ this PoW protocol in the proposed scheme, and use the root value of MHT as the second file tag T_0.

4 System Model and Design Goals

In this section, we introduce the architecture of fog storage and define some security requirements.

4.1 Fog Storage System

The fog storage system consists of three system entities: Cloud, Fog, and End user (Figure 2).

- **Cloud**: Centralized service provider, which provides data storage and retrieval service to users, and manages the fog devices.
- **Fog**: Distributed entities, which are used as the proxy of the cloud to provide fast services.
- **End user**: Data outsourcing/retrieving entities, which are divided into initial and subsequent uploader based on whether their data has been uploaded.

We regard initial and subsequent uploader as data owners. The local fog device of a data owner is denoted as F_0, and the data storage fog device is denoted as F_s.

4.2 Adversarial Model

We consider the following adversaries [8] in the proposed scheme.

- **Outside adversary**: The outside adversary may acquire some knowledge (eg., a hash value) of the file by eavesdropping, and pretend as a common user to interact with the remote server.
- **Inside adversary**: The insider adversary executes the assigned tasks honestly but would like to learn as much information of file as possible. Such as the cloud server or fog devices.

We assume that all service devices are honest-but-curious, and do not collude with outside adversary.

4.3 Security Requirements

Considering the aforementioned adversarial model, we propose the following security requirements [8]:

- **Privacy**: The proposed scheme should provide the outsourced data with PRV$-CDA-B security [3], and prevent the plaintext of outsourced data from any illicit access.
- **Integrity**: End users should be allowed to verify the integrity of data during the data retrieval phase, and the proposed scheme should prevent users from losing data under duplicate-faking attack.
- **Leakage resilience**: Information leakage of data should be minimized as possible during data outsourcing phase.
- **Forward and backward secrecy**: In cross-user deduplication, forward secrecy means that the revoked users should be deterred to access the data stored in the remote storage. Backward secrecy means that the user should be prevented from accessing the data stored in the remote storage before she/he obtains the valid ownership.

Algorithm 1

Setup(1^λ)
Select a hash function $H : \{0,1\}^* \to \{0,1\}^\lambda$, and select a λ-bit prime p that is the description of H.
Select a symmetric encryption scheme with key length λ: SKE.{KeyGen, Enc, Dec}.
Select a multiplicative group $\mathbb{G}$ with prime order p such that $\mathbb{G} = \langle g \rangle$ where g is the generator of $\mathbb{G}$.
Return(p,g,H,$\mathbb{G}$)

KeyGen($M = M_1 \parallel \cdots \parallel M_n$)
for each $i \in [1,n]$
$k_i = H(M_i)$
$k_{mas} = H(M)$
Return(k_{mas}, $\{k_i\}_{1\le i\le n}$)

Update($C_i^{(j)}$, $Rk_i^{(j)}$)
$r_{j+1} \in_R \mathbb{Z}_p^*$
$Rk_i^{(j+1)} \leftarrow \left(Rk_i^{(j)}\right)^{r_{j+1}}$
$T^{(j)\to(j+1)} = Rk_i^{(j+1)} / Rk_i^{(j)}$
$C_i^{(j+1)} \leftarrow C_i^{(j)} \cdot T^{(j)\to(j+1)}$
Return($C_i^{(j+1)}$,$Rk_i^{(j+1)}$)

Enc($\{M_i\}_{1\le i\le n}, \{k_i\}_{1\le i\le n}$)
for each $i \in [1,n]$
$C_i = \text{SKE.Enc}(k_i, M_i)$
$Ck = \text{SKE.Enc}(k_{mas}, k_1 \parallel \cdots \parallel k_n)$
Return($\{C_i\}_{1\le i\le n}$, Ck)

RkeyDrv($\{Rk_i^{(*)}\}_{1\le i\le n}$, id_s)
$C_{Rk} = id_s \cdot (Rk_1^{(*)} \parallel \cdots \parallel Rk_n^{(*)})$
Return(C_{Rk})

TagGen($\{C_i\}_{1\le i\le n}$, k_{mas})
$T_i = H(C[i])$, for each $i \in [1,n]$
$t = g^{k_{mas}}$
Return(T_i, t)

ReEnc(C_i)
$r_1 \in_R \mathbb{Z}_p^*$
$Rk_i^{(1)} = g^{r_1}$
$C_i^{(1)} = C_i \cdot Rk_i^{(1)}$
Return($C_i^{(1)}$, $Rk_i^{(1)}$)

Dec($\{C_i^{(*)}\}_{1\le i\le n}$, Ck, C_{Rk}, id_s, k_{mas})
$Rk_1^{(*)} \parallel \cdots \parallel Rk_n^{(*)} = \text{C}_{Rk}/id_s$
$k_1 \parallel \cdots \parallel k_n = \text{SKE.Dec}(k_{mas}, Ck)$
for each $i \in [1,n]$
$C_i \leftarrow C_i^* / Rk_i^*$
$M_i = \text{SKE.Dec}(k_i, C_i)$
Return($M = M_1 \parallel \cdots \parallel M_n$)

5 The Proposed Scheme

In this section, the proposed scheme is described in detail. The necessary algorithms are defined in Algorithm 1.

5.1 Overview

In general, our scheme first runs the file-level deduplication, and performs block-level deduplication when the data does not exist in fog computing. In the upload phase, the initial uploader is required to upload the unduplicated blocks as well as some duplicate blocks selected randomly, the subsequent uploader is requested for some random duplicate blocks. All of the random blocks will be discarded. In this way, data users cannot infer the existence of the file through the existence of the data block since they implement data outsource through the similar processes. After the data outsourcing, the outsourced data should be updated to ensure forward and backward secrecy while the ownership changes. To realize the ownership management in BC-Dedu scheme, we construct a dual-level ownership list to maintain the connections between files and updated blocks. Meanwhile, we design a corresponding update algorithm for a low-cost update operation. Moreover, we propose a dynamic data storage strategy that requires storage devices store blocks based on service demand to reduce the service costs and latency in fog computing.

5.2 Main Construction

5.2.1 System Setup

A trust initializer runs the **Setup** algorithm to obtain and publish the system parameters. The data owners are allowed to transfer data to fog storage.

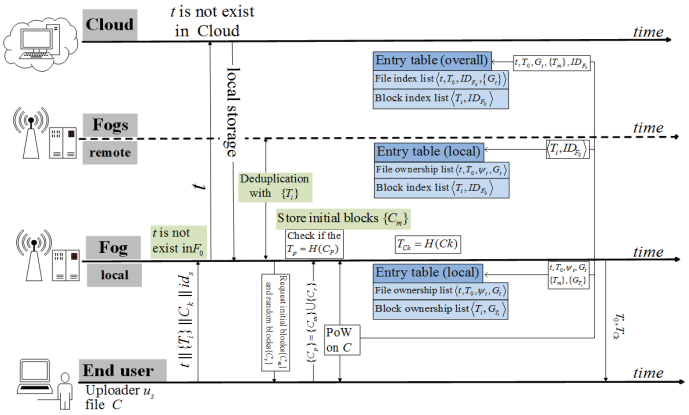


Figure 3: Initial upload

5.2.2 Data Outsourcing and Deduplication

When a data owner u_s tries to upload file M to fog storage, u_s invokes the **KeyGen, Enc** and **TagGen** algorithm to compute the file tag t, block keys ciphertext Ck, block ciphertexts $\{C_i\}_{1\le i\le n}$ and tags $\{T_i\}_{1\le i\le n}$. Then, u_s sends the upload message $Upload \parallel t \parallel \{T_i\} \parallel Ck \parallel id_s$ to the local fog device F_0, where id_s is the identity of u_s. Notably, we regard the file data block that does not exist in the fog storage as initial block C_m, and turn the existing file block into subsequent block C_s, such that: $\{C_m\} \bigcup \{C_s\} = \{C_i\}_{1\le i\le n}$.

Initial Upload: As is illustrated in Figure 3, If t is not in F_0 and the central cloud, F_0 searches the initial blocks $\{C_m\}$ with $\{T_i\}_{1\le i\le n}$ in the block index list maintained by the cloud, and deduplicates the subsequent blocks with related storage devices. Besides, F_0 picks some duplicate blocks $\{C_r\}(\in \{C_s\})$ ran-

domly, where the number of the random file blocks is determined by F_0 according to the security requirements. Then, F_0 requests u_s to return $\{C_m\}\bigcup\{C_r\}$. For each block C_i' returned by u_s, for each block C_i', F_0 checks the $H(C_i')$ with T_i, and retains the initial blocks $\{C_m\}$ if all the checks are passed. After that, F_0 triggers **PoW** protocol. If F_0 accepts u_s as a valid data owner, F_0 computes the tag of keys ciphertext $T_{Ck} = H(Ck)$. Finally, F_0 returns T_{Ck} and the root value T_0 of MHT to u_s for subsequent file retrieval.

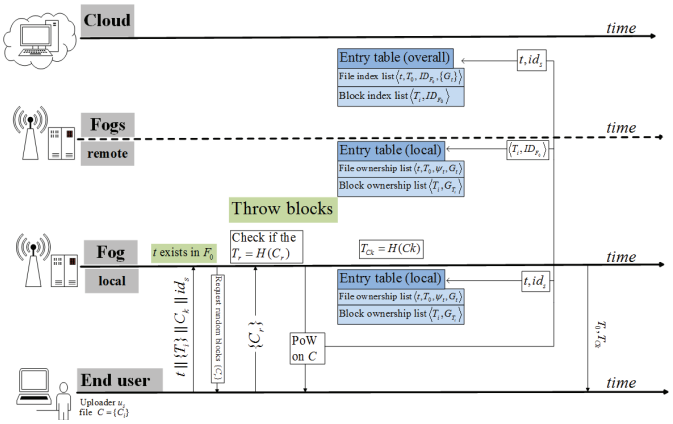


Figure 4: Subsequent upload

Subsequent Upload: If t exists in F_0, the detailed process performed by F_0 is described in Figure 4. F_0 checks the consistency between t and $\{T_i\}_{1\le i\le n}$ with ψ_t. If all the checks are passed, F_0 pretends to perform initial upload by requesting some random blocks $\{C_r\}(\in\{C_s\})$ from u_s. For each block C_i' returned by u_s, F_0 checks the $H(C_i')$ with T_i. If all the checks are passed, F_0 discards the random blocks and triggers **PoW** protocol. If u_s's proof is accepted, F_0 computes $T_{Ck} = H(Ck)$ and performs server-side deduplication over the key ciphertext. Finally, F_0 returns T_{Ck} and T_0 to u_s.

Note that when the consistency checks between t and $\{T_i\}_{1\le i\le n}$ are not all passed, F_0 performs the remainder procedures of initial upload, which alleviates the duplicate-faking attack by allowing the subsequent uploader to outsource the data that has been suffered the duplicate-faking attack. Besides, when t exists in another fog device F_s, F_s will employ F_0 as a proxy to perform the similar subsequent upload procedures as that executed by F_0 when t exists in F_0.

5.2.3 Onwership Management

For forward and backward secrecy, the storage devices should update the data during the following three case.

Data Upload: In this case, the data storage devices may perform the following four operation:

- For the file C uploaded by u_s initially, F_0 creates a file-level ownership list $L_F : \langle t, T_0, \psi_t, G_t\rangle$, where ψ_t is a map from t to $\{T_i\}_{1\le i\le n}$. Then, F_0 inserts u_s into the file ownership group $G_t = \{id_s\}$.
- For an initial block C_m, F_0 creates a block-level ownership list $L_B : \langle T_m, G_{T_m}\rangle$, where the block ownership group $G_{T_m} = \{ID_F\}$ consists of the valid fog devices. Besides, F_0 runs the **ReEnc** algorithm to re-encrypts C_m, and stores the re-encrypted ciphertext and re-encryption key $\left\langle C_m^{(1)}, Rk_m^{(1)}\right\rangle$ locally. Finally, F_0 informs the central cloud to update both file-level and block-level indexes.
- For a subsequent block C_s, F_0 informs the related block storage device F_s to run **Update** algorithm to update re-encrypted block.
- For the file C uploaded by u_s subsequently, F_0 inserts u_s into G_t, and informs related block storage devices to run the **Update** algorithm to update re-encrypted blocks. Then, F_0 requests the central cloud to update the overall file-level ownership list.

Data Deletion: When $u_s(\in G_t)$ wants to delete the file C, u_s sends the file deletion message with $Delete \parallel t \parallel T_0 \parallel id_s$ to F_0. F_0 removes id_s from G_t and informs related block storage devices to run the **Update** algorithm to update re-encrypted blocks. Then, F_0 informs the central cloud to update the overall file-level ownership list.

Data Modification: We consider the following two case of the data modification.

- *Block-deletion*: when u_s wants to obtain the file C' by deleting the i-th block of C, u_s computes the t', Ck' of C' and sends $Modify \parallel t \parallel T_0 \parallel T_i \parallel t' \parallel Ck' \parallel id_s$ to F_0. Then, F_0 removes the identity id_s from G_t and informs the block storage device who stores C_i to run **Update** algorithm to update re-encrypted blocks. Subsequently, F_0 performs initial upload or subsequent upload according to whether t' exists in the system. Finally, F_0 informs the central cloud to update the overall file-level ownership list.
- Block-modification: when u_s wants to obtain the file C' by modifying the i-th block C_i of file C with C_i'. u_s computes the new t', T_i', C_i', Ck' , then sends the modification message $Modify \parallel t \parallel T_0 \parallel T_i \parallel t' \parallel T_i' \parallel C_i' \parallel Ck' \parallel id_s$ to F_0, Finally, F_0 performs the similar procedures as *block-deletion*.

5.2.4 Retrieval

When $u_s(\in G_t)$ wants to retrieves the file M that is stored in F_0, u_s sends data retrieval message $Retrieval \parallel$

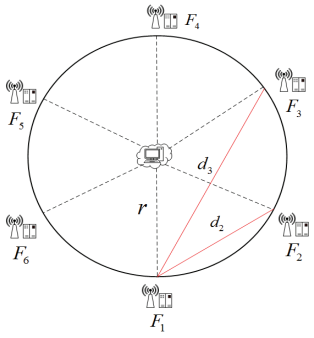


Figure 5: Ideal system model of fog storage

$t \parallel T_0 \parallel T_{Ck} \parallel id_s$ to F_0. F_0 validates the validity of the u_s through the file-level ownership list L_F, and requests all related storage devices F_s for corresponding file blocks and re-encryption key. Then, F_s validates the validity of F_0 through the block-level ownership list L_B, and returns the file blocks. F_0 constructs a file ciphertext $C^* = C_1^{(*)} \parallel \cdots \parallel C_n^{(*)}$, and computes a re-encryption key ciphertext C_{Rk} via **RkDrv** algorithm. Then, F_0 returns $\langle C^*, Ck, C_{R_k} \rangle$ to u_s. u_s takes the C^*, Ck, and C_{R_k} as input to run the **Dec** algorithm to obtain the file M.

5.2.5 Dynamic Data Storage

In fog computing, the central cloud offers a wide range of low-latency computing services by using fog device adjacent to the users, which means that the distance from local fog device to end user (*intra-network*) is much less than that from the central cloud to this fog device (*inter-network*). Therefore, the data access costs and latency is mainly caused by *inter-network* transmission. To reduce the service costs and latency, we propose a novel dynamic data storage strategy, which can periodically store data blocks in the devices of the fog computing according to the service demand.

For a concise illustration, we establish a ideal fog storage system as shown in Figure 5, where the six fog devices are distributed uniformly. Suppose that the distance from each fog device to cloud is r, the distance from j-th fog device to F_1 is d_j, and $\{d_j\}_{1 \le j \le 6} = \{0, r, \sqrt{3}r, 2r, \sqrt{3}r, r\}$. For a block of size S_{C_i} that stored in F_1 periodically, we assume that the number of access from j-th fog device to C_i is num_j. Then, we introduce the following notions:

$$\begin{aligned} cost_{C_i} &= \sum_{j=2}^{n} d_j \cdot num_j \cdot S_{C_i} \\ cost'_{C_i} &= \sum_{j=1}^{n} num_j \cdot r \cdot S_{C_i} \\ Rate_{C_i} &= cost_{C_i} / (cost_{C_i} + cost'_{C_i}). \end{aligned}$$

Where $cost_{C_i}$ is the total access costs of block C_i stored in F_1, the $cost'_{C_i}$ is the total access costs of block C_i stored in the central cloud. Notably, when $Rate_{C_i} > 0.5$, it is cost-efficient for F_1 to move local block C_i to the central cloud. Without loss of generality, we assume that each fog device accesses C_i with an identical possibility. Then, the mean distance for fog devices to access block C_i is $(4 + 2\sqrt{3})r/5$. Thus, $Rate_{C_i} > 0.5$ means that:

$$rate_{F_1} = \frac{num_1}{\sum_{j=1}^{n} num_j} < \frac{2\sqrt{3} - 1}{4 + 2\sqrt{3}} = rate_0$$

Where $rate_{F_1}$ is the percentage of the accesses launched by local device in total accesses. Obviously, the $rate_0$ is a suitable criterion for data storage devices to determine the next storage location of C_i, since $rate_0$ causes less computational overhead than $Rate_{C_i}$.

In the proposed dynamic data storage strategy, the central cloud computes the $rate_0$ as a system parameter. The fog device F_s will move a local block to the central cloud if $rate_{F_s} < rate_0$. Similarly, a block stored in cloud will be moved to fog device F_j if $rate_{F_j} > rate_0$. The decrease of data access costs is mainly caused by the decrease of mean access distance of fog devices in this dynamic data storage strategy, which also reduces the latency of data service. Besides, we allow the central cloud to adjust $rate_0$ according to service demand.

6 Security Analysis

6.1 Privacy

In the proposed scheme, each data block of the unpredictable file is re-encrypted with re-encryption key $Rk_i^{(1)}$ from C_i to $C_i^{(1)}$ in the initial upload phase. When the file ownership changes, $\langle C_i^{(j)}, Rk_i^{(j)} \rangle$ is updated with $\langle C_i^{(j+1)}, Rk_i^{(j+1)} \rangle$. The re-encryption key is retained secretly in the block storage device, and is distributed securely to a valid user u_s only when u_s retrieves the corresponding file. Therefore, the outside adversary cannot decrypt the ciphertext without the valid re-encryption key.

For a stronger security demonstration, we make a compromise to assume that the outside adversary obtains the corresponding identity and decrypts all blocks from $\{C_i^{(*)}\}$ to $\{C_i\}$. In this case, bounded by the hardness assumption of **DLP**, the outside adversary plays an identical role with the inside adversary since they cannot obtain the master key k_{mas} from $t = g^{k_{mas}}$. Besides, both the inside and the outside adversary are unable to learn any information from the ciphertext blocks $\{C_i\}$. We follow give an outline of security proof based on the existing works [1, 3, 23].

Theorem 1. *Let* SKE.{KeyGen, Enc, Dec} *be a symmetric encryption scheme with key length* λ*, and model* $H(\cdot)$ *as a random oracle. If there exists an adversary* $\mathcal{B}'$ *that can break the KR-security with advantage* $\mathrm{Adv}_{\mathrm{KR}}^{\mathcal{B}'}(\lambda)$*, and exists an adversary* $\mathcal{D}'$ *that breaks the ROR-security with advantage* $\mathrm{Adv}_{\mathrm{ROR}}^{\mathcal{D}'}(\lambda)$*. There exists a PRV$-CDA-B adversary* $\mathcal{A}$ *in the proposed scheme such that:*

$$\mathrm{Adv}_{\mathrm{PRV\$-CDA-B}}^{\mathcal{A},\mathcal{M}}(\lambda) \le \mathcal{O}(qn) \cdot \mathrm{Adv}_{\mathrm{KR}}^{\mathcal{B}'}(\lambda)$$

$$+\mathrm{Adv}^{\mathcal{D}'}_{\mathrm{ROR}}(\lambda) + \frac{n^2}{2^{B+1}} + \frac{qn}{2^{\mu}}$$

where $\mathcal{M}$ is a block-source with min-entropy μ, B is the length of a block, n is the number of block messages, and q is the number of queries to $H(\cdot)$ by $\mathcal{A}$.

Proof. We introduce a sequence of PRV$-CDA-B games to prove the privacy of our scheme by transiting the game from the world of hidden bit 0 to the world of bit 1. Meanwhile, we demonstrate that each transition is indistinguishable from the security of the underlying primitive.

Game G_0: An initial game that holds a hidden bit 0.

Game G_1: Except that a table is created by the challenger $\mathcal{C}$ to track the random oracle queries when encrypting the file M, this game is identical to G_0. On the condition that all M_i are distinct, $\mathcal{C}$ will abort the game if the result of a query X has been defined as $H(X)$ during an earlier query. Thus, $\Pr[\mathrm{G}^{\mathcal{A}}_0] \leq \Pr[\mathrm{G}^{\mathcal{A}}_1] + \Pr[\mathrm{G}^{\mathcal{A}}_1 \text{ sets bad}]$. Note that the total number of random oracle queries is bounded by the number of ciphertexts. By union bound, we conclude that $\Pr[\mathrm{G}^{\mathcal{A}}_1 \text{ sets bad}] = \sum_{\mathrm{i}=0}^{\mathrm{n}-1} \frac{\mathrm{i}}{2^{\mathrm{B}}} < \frac{\mathrm{n}^2}{2^{\mathrm{B}+1}}$.

Game G_2: During the challenge phase, this game is identical to G_0 until $\mathcal{A}$ makes a "bad" query of $H(X)$, and $\mathcal{C}$ aborts game due to the "bad" query. Bounded by the KR-security of the symmetric-key encryption scheme, we argue that this occurrence is negligible.

Note that the hash value of data is used as an encryption key in symmetric encryption, and the corresponding ciphertexts are sent to $\mathcal{A}$. Suppose that an adversary $\mathcal{B}$ makes such bad queries with non-negligible probability, we can break the KR-security by building an adversary $\mathcal{B}'$. $\mathcal{B}'$ just guesses hash query j^* and the encryption index i^*. Besides, $\mathcal{B}'$ plants its own key-recovery challenge c^* in the i^*-th encryption and outputs j^*-th hash query. Thus, $\Pr[\mathrm{G}^{\mathcal{A}}_1] \leq \Pr[\mathrm{G}^{\mathcal{A}}_2] + \Pr[\mathrm{G}^{\mathcal{A}}_2 \text{ sets bad}]$. By a hybrid argument, $\Pr[\mathrm{G}^{\mathcal{A}}_2 \text{ sets bad}] \leq \mathrm{qn}' \cdot \mathrm{Adv}^{\mathcal{B}'}_{\mathrm{KR}}(\lambda)$.

Game G_3: A further transition is to replace all encryptions of message $\mathrm{M}[i]$ or $\mathrm{M}'[j]$ with encryptions of random messages of the same length. This is possible due to ROR-security of the symmetric-key encryption scheme. Suppose that an adversary $\mathcal{D}$ who can distinguish this game from G_2, so an adversary $\mathcal{D}'$ can be built to breaks the ROR-security as follows. In ROR game, $\mathcal{D}'$ computes the ciphertext for $\mathcal{D}$ by querying its encryption oracle, then $\mathcal{D}'$ outputs the output of $\mathcal{D}$. $\Pr[\mathrm{G}^{\mathcal{A}}_3 \text{ sets bad}] \leq \mathrm{Adv}^{\mathcal{D}'}_{\mathrm{ROR}}(\lambda)$. Thus, $\Pr[\mathrm{G}^{\mathcal{A}}_2] \leq \Pr[\mathrm{G}^{\mathcal{A}}_3] + \Pr[\mathrm{G}^{\mathcal{A}}_3 \text{ sets bad}]$.

Game G_4: When $\mathcal{A}$ queries $H(M[i])$ for some i, $\mathcal{C}$ aborts. Recall in G_3, for the adversary $\mathcal{A}$, all the ciphertexts are independent of the true ciphertext C. So we can bound the above probability by applying the min-entropy of $\mathcal{M}$. By union bound, we have $\Pr[\mathrm{G}^{\mathcal{A}}_4 \text{ sets bad}] = \sum_{\mathrm{i}=1}^{\mathrm{n}} \frac{\mathrm{q}}{2^{\mu}} \leq \frac{\mathrm{qn}}{2^{\mu}}$. Therefore, $\Pr[\mathrm{G}^{\mathcal{A}}_3] \leq \Pr[\mathrm{G}^{\mathcal{A}}_4] + \Pr[\mathrm{G}^{\mathcal{A}}_4 \text{ sets bad}]$. Moreover, **Game 4** implements exactly the case where $b = 1$ such that:

$$\mathrm{Adv}^{\mathcal{A},\mathcal{M}}_{\mathrm{PRV\$-CDA-B}}(\lambda) = \Pr[\mathrm{G}^{\mathcal{A}}_0] - \Pr[\mathrm{G}^{\mathcal{A}}_4].$$

□

6.2 Integrity

When the valid data owner u_s obtains outsourced file M' during **Retrieval** phase, u_s can verify the integrity of outsourced data based on whether the equation $t = g^{H(M')}$ holds. Furthermore, the proposed scheme alleviates the duplicate-faking attack by allowing the subsequent uploader to outsource the data that has been suffered the duplicate-faking attack. In this way, the correct version is stored in fog storage.

6.3 Leakage Resilience

In the proposed scheme, the outside adversary cannot learn the existence of outsourced data by launching CoF attack, since they cannot accurately distinguish whether they are performing an initial upload or a subsequent upload according to the respond of F_0. Specifically, when a data owner u_s tries to upload file C to local fog device F_0, four cases may occur as follows:

Case (1): All blocks of C are not in fog storage;

Case (2): Partial blocks of C are not in fog storage;

Case (3): All blocks of C are in fog storage, but C is not;

Case (4): C exists in fog storage.

During the data outsourcing phase, F_0 requests data owner to upload blocks that consist of initial blocks and some random blocks, where the number of random data blocks is determined by the F_0 according to the security requirement. When the number of the requested blocks is equal to the sum of the target file, Case (2) is indistinguishable from Case (1); When the number of initial blocks is 0, Case (2), Case (3) and Case (4) are indistinguishable. Then, all blocks returned by u_s will be checked. If all of the checks are passed, the PoW protocol will be triggered. In the actual service scenario, the Case (2), Case (3) and Case (4) are occurring in data outsourcing phase frequently, where the uploader cannot learn the existence of outsourced data from the respond of F_0. Therefore, our scheme protects the outsourced data from the side information leakage.

6.4 Forward and Backward Secrecy

In the proposed scheme, when a data owner $u_s(\in G_t)$ deletes or modifies the outsourced file C, the file storage device F_s removes u_s from file-level ownership list, and

informs the related block storage devices to update the data blocks and re-encryption key. Specifically, for a block $C_i^{(j)}$, a random exponent r_{j+1} is chosen to generate the new re-encryption key $Rk_i^{(j+1)}$ that will be securely stored in the block storage device. Then, the block ciphertext $C_i^{(j)}$ is re-encrypted to $C_i^{(j+1)}$ with $Rk_i^{(j+1)}/Rk_i^{(j)}$. When a valid data owner u_s requests the outsourced file, all block re-encryption keys will be integrated with the identity of u_s and returned to u_s. In this way, our scheme ensures the forward secrecy of outsourced data since the revoked users cannot decrypt the updated ciphertext without a valid re-encryption key.

When a subsequent uploader u_s uploads file to fog storage, the file storage device F_s inserts u_s into the file-level ownership list and informs the related block storage devices to update the data blocks and re-encryption keys. All of the re-encryption keys are distributed to u_s in a secure manner during the **Retrieval** phase. Thus, the unauthorized users are unable to decrypt the updated ciphertext since they have no re-encryption key before they obtain valid ownership by uploading the data. The backward secrecy of the outsourced data is guaranteed.

7 Performance Analysis

In this section, we analyze the proposed scheme and compare it with some state-of-the-art deduplication schemes in theoretical and practical aspects.

7.1 Comparisons

Table 1: Comparison of deduplication schemes

	Ownership management	**Leakage resilience**	**Dynamic storage**
BKR [1]	×	×	×
HKSK [7]	File-level	√	×
KH [10]	File-level	×	×
Ours	Block-level	√	√

Table 1 shows the comparison results of some deduplication schemes in terms of ownership management, leakage resilience, and dynamic data storage. In addition to BKR, the remainders provide ownership management for the outsourced data by updating key. Specifically, KH and HKSK achieve ownership management in file-level deduplication, and our scheme achieves ownership management in block-level deduplication, which supports more space savings in fog storage. As regards the leakage resilience in secure deduplication, HKSK can prevent data from the side information leakage based on its server-side deduplication architecture, and the BKR and KH are vulnerable to the side information leakage since the outside adversary could learn the existence of the data by launching the CoF attack. Due to our unique deduplication protocol, the proposed scheme can resist the side information leakage efficiently.

Both BKR and HKSK do not support dynamic data storage due to their architecture of single server cloud storage. KH alleviates the service pressure on the central cloud by storing the outsourced data in fog device in a period. However, all of the outsourced data will be moved to the central cloud finally. Our scheme requires the block storage devices periodically store blocks based on the service demand to reduce service costs and latency while alleviating the pressure on the cloud server.

Table 2: Notations used in theoretical analysis

Notation	Description
C_G	Bitlength of an element in $\mathbb{G}$
C_Z	Bitlength of an element in $\mathbb{Z}_p^*$
C_H	Bitlength of a hash value
C_M	Bitlength of a file M
C_k	Bitlength of a block keys ciphertext
C_{Rk}	Bitlength of a ReEnc keys ciphertext
n	Number of blocks in file M
m	Number of unduplicate blocks of M
r	Number of requested random blocks
u	Number of challenge blocks
$\mathcal{O}$	Number of data owners of file M
e	Evaluation of bilinear map
H	Evaluation of hash function
Exp	Evaluation of exponentiation
Mul	Evaluation of multiplication
Enc	Evaluation of symmetric key encryption/decryption
E_k	Evaluation of the encryption/decryption of block keys

7.2 Efficiency Analysis

We define the notations in Table 2, which are used in the following efficiency analysis in terms of computation costs, communication overheads and storage overheads.

Computation costs: Table 3 shows the computation costs of different deduplication schemes in different phases. The initial upload includes the costs of all operations for the initial uploader to outsource a new data to the remote storage. Similarly, the subsequent upload includes the costs of the subsequent uploader to regenerate the data which exists in remote storage. Verification includes the costs caused by the PoW process, Update invokes the costs for the data update, and Retrieval invokes the costs for the decryption of outsourced data on client-side.

With regard to initial upload, subsequent upload, and retrieval phases, the computation costs of our scheme are similar to HKSK and BKR, where the additional symmetric encryption operation is caused by block keys management, and it is acceptable in large file deduplication. Besides, the computation of HK is higher than other schemes, which will be illustrated with corresponding simulation experiment in the following subsection.

Only the KH and our scheme support the PoW and ownership management. As regards the verification, the computation costs of our scheme are less than KH, since the PoW protocol employed in our scheme supports multiple blocks verification simultaneously.

Table 3: Comparison of computation costs

	Initial upload	Subsequent upload	Verification	Update	Retrieval
BKR [1]	$nEnc + 2H$	$nEnc + 2H$	-	-	$nEnc$
KHSK [7]	$nEnc + 2H$	$nEnc + 2H$	-	-	$nEnc$
KH [10]	$3e + 5Exp$ $+(n+5)Mul + 1H$	$3e + Exp$ $+(n+3)Mul + 1H$	$u(\log n + 1)H$	$(\mathcal{O}+3)Exp$ $+(n+1)Mul$	$2e + Exp$ $+(n+3)Mul$
Ours	$nEnc + E_k + 2H$	$nEnc + E_k + 2H$	$\sum_{i=1}^{u}(\log n + 2 - i)H$	$nExp + 2nMul$	$nEnc + E_k$ $+nMul$

Table 4: Comparison of communication and storage overhead

	Communication overhead			Storage overhead	
	Initial upload	Subsequent upload	Retrieval	Service provider	Data owners
BKR [1]	$C_M + 3C_H$	$2C_H$	$C_M + C_H$	$C_M + 2C_H$	$2C_H$
HKSK [7]	$C_M + C_H$	$C_M + C_H$	$C_M + 2C_H$	$C_M + 2C_H$	$2C_H$
KH [10]	$C_M + C_H + 3C_G$	$C_G + u(\log n + 1)C_H$	$C_M + 4C_G + C_H$	$C_M + \lvert\mathcal{O}\rvert C_G$	$C_H + C_Z$
Ours	$\frac{m+r}{n}C_M$ $+(n+2)C_H + C_k$	$\frac{r}{n}C_M + C_k$ $+\sum_{i=1}^{u}(\log n + 2 - i)C_H$	$C_M + 2C_H$ $+C_k + C_{Rk}$	$\frac{m}{n}C_M + (n+2)C_H$ $+C_k + C_{Rk}$	$3C_H$

Considering the Update, with the increasing volume of the data owners, our scheme consumes less time than that of the KH.

Table 4 summarizes the comparison results of different schemes in terms of communication and storage overhead. The proposed scheme supports block-level client-side deduplication, and other schemes are file-level deduplication scheme, which means that our scheme saves more storage space and bandwidth.

Communication overheads: Compared with other schemes, only the partial data blocks are requested in our scheme during the initial upload phase, which saves more bandwidth than other schemes. Despite some random blocks are requested in the subsequent upload, the additional overheads can be regarded as a tradeoff since our scheme obtains a better leakage resilience than the common client-side deduplication and more bandwidth savings than the server-side deduplication.

Storage overheads: Based on the architecture of block-level deduplication, our scheme provides more efficient space savings than other schemes. Furthermore, considering the implementation of ownership management in the proposed block-level deduplication scheme, the increase in space caused by block tags, block key ciphertext, and re-encryption keys ciphertext can be seen as a trade-off, that is negligible.

7.3 Simulation

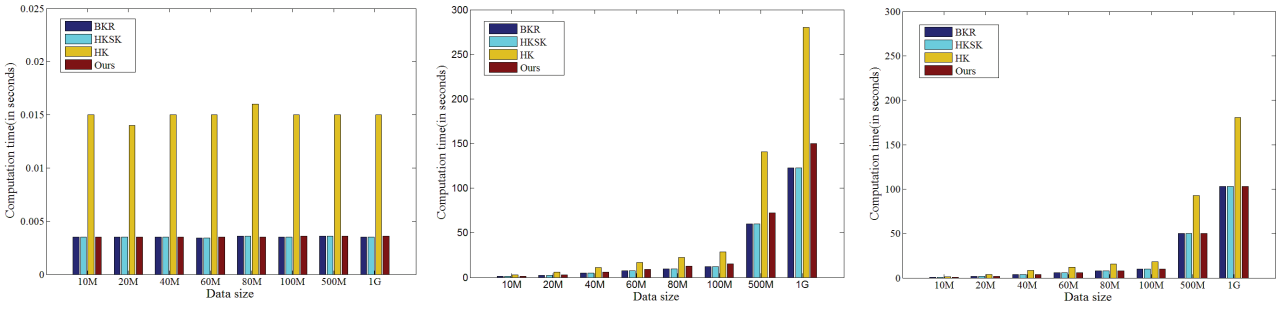
We conduct a series of simulation experiments to analyze the performance of the proposed scheme in terms of computation costs and resource utilization.

Primarily, we measure the computation costs of MLE-based schemes during different phases by using the Crypto++ library ver.5.6.2, where the SHA-256 is used as a cryptographic hash function to generate encryption key and tags, and the AES-128 with Electronic Code Book (ECB) mode is employed as an encryption/decryption function. Besides, we use the Pairing-Based Cryptography (PBC) library (Version 0.5.14) built upon the GNU Multiple-Precision (GMP) library (Version 6.0.0a) to implement HK's scheme. The size of the blocks is 1MB. All experiments are performed on a laptop with the 2.5 GHz Intel(R) Core(TM) i5-3210M CPU and 8GB memory. Note that each experimental data was obtained from the average of 20 repeated samples.

The computation costs of different schemes are shown in Figure 6. As is illustrated in Figure 6(a), the computation costs of each scheme in **Setup** phase will not change with the size of the file, and our scheme consumes almost $3.5ms$ in this phase, which is the same as that of BKR and HKSK. Besides, the KH's computation costs is $15ms$ and is higher than other schemes. During the outsourcing and retrieval phases, the computation costs of each scheme are shown in Figure 6(b) and Figure 6(c), which are proportional to the size of the outsourced data.

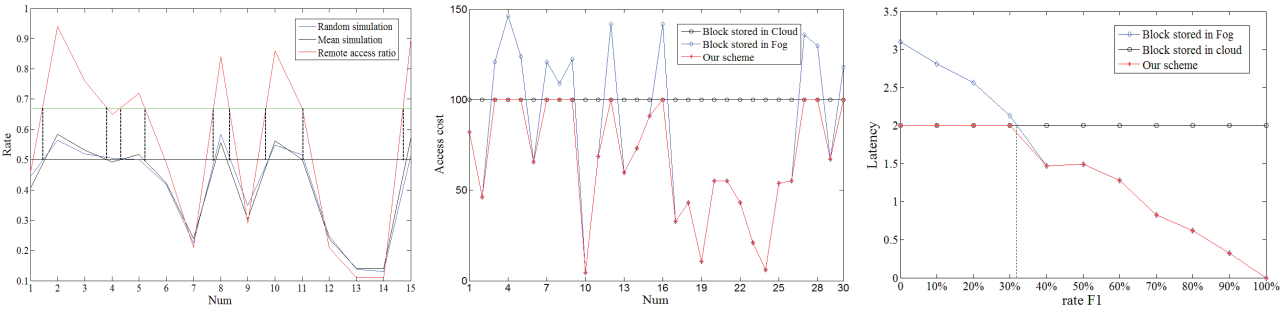
During the data **Outsourcing** phase, when the size of the outsourced file is 10MB, the outsourcing time is $1.2s$ for BKR and HKSK, $2.8s$ for KH, and $1.5s$ for our scheme. While the file size increases to 1024MB, the corresponding time is $122.9s$, $280.4s$ and $151.2s$. Compared with BKR and HKSK, the additional computation costs of our scheme is caused by the block tags generation, which is necessary for the proposed block-level deduplication.

During the **Retrieval** phase, when the size of outsourced data is 10MB, the computation time is $1.02s$ for BKR, HKSK, $1.46s$ for KH, $1.1s$ for our scheme. While the file size increases to 1024MB, the corresponding time is $102.9s$, $180.4s$, $103.5s$. Note that the computation time of our scheme is almost 57.4% as that of KH.



(a) Setup (Server side) (b) Outsourcing (Client side) (c) Retrieval (Client side)

Figure 6: Comparison of computation time



(a) Comparison of access cost ratio $Rate_{C_i}$ (b) Comparison of access costs (c) Comparison of access latency

Figure 7: Data access of inter-network

Next, we analyze the rationality of our dynamic data storage strategy under ideal fog storage model mentioned before. For simplicity, the number of total access from fogs to blocks is assumed as 100, the distance between fogs and cloud is assumed as 1 unit, the data transmission speed is 2s/unit and the size of block C_i is assumed as 1Mb. We conduct a large number of simulation experiments, and select some samples randomly for analysis.

As is depicted in Figure 7(a), a block is suitable to store in its current fog device if its $Rate_{C_i} < 0.5$. Notably, the trend of actual distance simulation (highlighted by blue) is similar to that of the mean distance simulation (highlighted by black). Besides, the value of actual distance simulation and mean distance simulation are close to 0.5 when the remote access ratio ($1 - rate_{F_1}$, highlighted by red) approaches a specific value (highlighted by green). Distinctly, the proportion $rate_{F_1}$ of the local accesses in total accesses can be used as the criterion to determine the next storage location of the block.

Under the same block access situation, we demonstrate the performance of our dynamic storage strategy in Figure 7(b) and Figure 7(c). Notably, Both the access costs and latency of our scheme are always equal to the minimum of the other two situation. Thus, our dynamic data storage strategy achieves efficient resource savings and low-latency services.

8 Conclusion

This paper has proposed a secure and efficient BC-Dedu scheme in commercial fog computing, which provides a comprehensive privacy-preservation for the outsourced data, especially in leakage resilience, forward and backward secrecy. Besides, we proposed a dynamic data storage strategy to obtain low-cost and low-latency data access services by utilizing the fog storage resource efficiently. Both the security and performance analysis demonstrate that the proposed scheme is suitable for the deduplication of large encrypted data in fog storage where ownership changes frequently.

Acknowledgment

This work is supported by the Fundamental Research Funds for the Central Universities (XJS17053, JBF181501).

References

[1] M. Bellare, S. Keelveedhi, and T. Ristenpart, "Message-locked encryption and secure deduplication," in *Advances in Cryptology*, pp. 296–312, 2013.

[2] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the internet of things," in

ACM Edition of the Mcc Workshop on Mobile Cloud Computing, pp. 13–16, 2012.

[3] R. Chen, Y. Mu, G. Yang, and F. Guo, "Bl-mle: Block-level message-locked encryption for secure large file deduplication," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 12, pp. 2643–2652, 2015.

[4] J. R. Douceur, A. Adya, W. J. Bolosky, S. Dan, and M. Theimer, "Reclaiming space from duplicate files in a serverless distributed file system," in *The 22nd International Conference on Distributed Computing Systems*, pp. 617–624, 2002.

[5] S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg, "Proofs of ownership in remote storage systems," in *The 18th ACM Conference on Computer and Communications Security*, pp. 491–500, 2011.

[6] D. Harnik, B. Pinkas, and A. Shulmanpeleg, "Side channels in cloud services: Deduplication in cloud storage," *IEEE Security and Privacy*, vol. 8, no. 6, pp. 40–47, 2010.

[7] J. Hur, D. Koo, Y. Shin, and K. Kang, "Secure data deduplication with dynamic ownership management in cloud storage," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 11, pp. 3113–3125, 2016.

[8] S. Jiang, T. Jiang, and L. Wang, "Secure and efficient cloud data deduplication with ownership management," *IEEE Transactions on Services Computing*, pp. 1–14, 2017.

[9] K. Kim, T. Youn, N. Jho, and K. Chang, "Client-side deduplication to enhance security and reduce communication costs," *Etri Journal*, vol. 39, no. 1, pp. 116–123, 2017.

[10] D. Koo and J. Hur, "Privacy-preserving deduplication of encrypted data with dynamic ownership management in fog computing," *Future Generation Computer Systems*, vol. 78, no. 2, pp. 739–752, 2018.

[11] D. Koo, Y. Shin, J. Yun, and J. Hur, "A hybrid deduplication for secure and efficient data outsourcing in fog computing," in *IEEE International Conference on Cloud Computing Technology and Science*, pp. 285–293, 2016.

[12] M. Kutylowski, J. Li, K. Kluczniak, X. Chen, and J. Wang, "Trdup: Enhancing secure data deduplication with user traceability in cloud computing," *International Journal of Web and Grid Services*, vol. 13, no. 3, pp. 270–288, 2017.

[13] S. Lee and D. Choi, "Privacy-preserving cross-user source-based data deduplication in cloud storage," in *International Conference on ICT Convergence*, pp. 329–330, 2012.

[14] J. Li, J. Li, D. Xie, and Z. Cai, "Secure auditing and deduplicating data in cloud," *IEEE Transactions on Computers*, vol. 65, no. 8, pp. 2386–2396, 2016.

[15] L. Liu, Z. Cao, and C. Mao, "A note on one outsourcing scheme for big data access control in cloud," *International Journal of Electronics and Information Engineering*, vol. 9, no. 1, pp. 29–35, 2018.

[16] M. Mukherjee, R. Matam, S. Lei, L. Maglaras, M. Ferrag, N. Choudhury, and V. Kumar, "Security and privacy in fog computing: Challenges," *IEEE Access*, vol. 5, pp. 19293–19304, 2017.

[17] P. Puzio, R. Molva, M. Onen, and S. Loureiro, "Cloudedup: Secure deduplication with encrypted data for cloud storage," in *IEEE 5th International Conference on Cloud Computing Technology and Science*, pp. 363–370, 2013.

[18] J. Singh, "Cyber-attacks in cloud computing: A case study," *International Journal of Electronics and Information Engineering*, vol. 1, no. 2, pp. 78–87, 2014.

[19] Z. Wang, Y. Lu, and G. Sun, "A policy-based deduplication mechanism for securing cloud storage," *International Journal of Electronics and Information Engineering*, vol. 2, no. 2, pp. 70–79, 2015.

[20] J. Xu, E. C.Chang, and J. Zhou, "Weak leakage-resilient client-side deduplication of encrypted data in cloud storage," in *The 8th ACM SIGSAC Symposium on Information, Computer and Communications Security*, pp. 195–206, 2013.

[21] C. Yang, J. Zhang, X. Dong, and J. Ma, "Proving method of ownership of encrypted files in cloud deduplication deletion (in chinese)," *Journal of Computer Research and Development*, vol. 52, no. 1, pp. 248–258, 2015.

[22] B. Yin, W. Shen, Y. Cheng, L .X. Cai, and Q. Li, "Distributed resource sharing in fog-assisted big data streaming," in *IEEE International Conference on Communications*, pp. 1–6, 2017.

[23] Y. Zhao and S. S. M. Chow, "Updatable block-level message-locked encryption," in *2017 ACM on Asia Conference on Computer and Communications Security*, pp. 449–460, 2017.

Biography

Hua Ma received her B.S. and M.S. degrees in Mathematics from Xidian University, China, in 1985 and 1990, respectively. She is a professor of Mathematics and statistics. Her research includes security theory and technology in electronic commerce design and analysis of fast public key cryptography theory and technology of network security.

Guohua Tian received his B.S. degree in 2016 from School of Mathematics and Information Science, Shaanxi Normal University. Now, he is a master degree student in Mathematics at Xidian University. His research focuses on network and information security.

Linchao Zhang received his B.S. degree in 2015 from College of Mathematics and Applied Mathematics, Hunan Institute of Science and Technology. Now, he is a master degree student in Mathematics at Xidian University. His research focuses on Proxy re-encryption and data security.

A Note On One Secure Data Self-Destructing Scheme in Cloud Computing

Lihua Liu[1], Yang Li[1], Zhengjun Cao[2], and Zhen Chen[2]
(Corresponding author: Zhengjun Cao)

Department of Mathematics, Shanghai Maritime University, China[1]
Department of Mathematics, Shanghai University[2]
No.99, Shangda Road, 200444, Shanghai, China
(Email: caozhj@shu.edu.cn)
(Received May 19, 2018; Revised and Accepted Nov. 9, 2018; First Online June 14, 2019)

Abstract

Recently, Xiong *et al.* have proposed a secure data self-destructing scheme [IEEE TCC, vol. 2, no. 4, pp. 448-458, 2014] in cloud computing. The scheme aims to solve some important security problems by supporting user-defined authorization period and by providing fine-grained access control during the period. The sensitive data will be securely self-destructed after a user-specified expiration time. In this note, it shows that the scheme is flawed because its decryption mechanism is incorrect. The consistency between encryption mechanism and decryption mechanism is not kept. We also show that it seems difficult to revise its decryption mechanism.

Keywords: Attribute-based Encryption; Cloud Computing; Data Self-destructing Scheme; Fine-grained Access Control; Time-Specific Encryption

1 Introduction

Cloud computing greatly benefits data mining, computational financing, and many other data-intensive activities by supporting a paradigm shift from local to network-centric computing and network-centric content. It enables customers with limited computational resources to outsource large-scale computational tasks to the cloud [20–22, 27, 28, 31].

Attribute-based encryption (ABE), introduced by Sahai and Waters, is a type of fuzzy identity-based encryption. In the scenario, a user's identity is composed of a set of strings which serve as descriptive attributes of the user, and the sender only needs to know the receivers' description in order to determine their public key. ABE has attracted much attention [14]. For example, Lewko, Waters, Pirretti, Goyal, Yamada, *et al.* [1, 25, 37] studied the construction of ABE systems and its shortcomings. Ostrovsky, Sahai, and Waters [29] investigated some non-monotonic access structures of ABE. Bethencourt, Sahai, Waters, and Goyal, *et al.* proposed some ciphertext-policy ABE schemes [2, 16, 33]. Chase and Chow [8, 9] introduced the setting of multi-authority in ABE. Hohenberger and Waters [17] discussed oline/offline ABE. In 2018, Cao *et al.* [4] discussed an inherent shortcoming of the cryptographic primitive of ABE. Notice that these ABE schemes do not support user-defined authorization period and secure self-destruction after expiration for privacy-preserving of the data lifecycle in cloud computing, because of the lacking of time constraints.

The cryptographic primitive of data self-destructing, introduced by Geambasu *et al.* [15], enables users to control over the lifecycle of the sensitive data. Recently, Xiong *et al.* [36] employed identity-based timed release encryption algorithm [6] and the distributed hash table network and proposed a full lifecycle privacy protection scheme for sensitive data. The time-specific encryption [30] is an extension of timed release encryption (TRE) [6]. In TRE, a piece of protected data can be encrypted in such a way that it cannot be decrypted (even by a legitimate receiver who owns the decryption key for the ciphertext) until the time (called the release-time) that was specified by the encryptor. Most of the previous TRE schemes do not consider the sensitive data privacy after expiration [23, 24].

In 2013, Chen *et al.* [13, 38] investigated on achieving secure role-based access control on encrypted data in cloud storage. In 2014, Chen *et al.* proposed two computation outsourcing schemes for linear equations and for linear programming [10,11]. But the schemes are insecure because the technique of masking a vector with a diagonal matrix is vulnerable to statistical analysis attacks [5]. The Wang *et al.*'s scheme for outsourcing linear equations is flawed [3], too. Hsien *et al.* [7, 12, 18, 26, 32] have presented some good surveys on public auditing for secure data storage in cloud computing.

In 2014, Xiong *et al.* [35] proposed a data self-destructing scheme in cloud computing by using key-policy attribute-based encryption with time-specified attributes. In the scheme, every ciphertext can only be

decrypted if both the time instant is in the allowed time interval and the associated attributes satisfy the key's access structure. The scheme aims to provide an encryption mechanism with multipurpose, such as confidentiality, data self-destructing function, and flexible control on legitimate receivers. In this note, we would like to stress that Xiong *et al.*'s scheme is flawed because the user cannot finish the calculations in the decryption phase. Furthermore, we want to point out that it is difficult to simply revise the decryption mechanism, because it requires that the authority should share the secret exponents with the user, which enables the user to decrypt any ciphertext.

The remainder of this paper is organized as follows. It reviews Xiong *et al.*'s scheme in Section 2, and then points out that the scheme has three drawbacks in Section 3. The first is that its consistency between encryption mechanism and decryption mechanism is not kept, which means a legitimate receiver cannot successfully recover the plaintext. We then point out that the scheme cannot be simply revised because the authority has to share the session exponents with any legitimate user. We also explain the reason for setting lots of parameters in Xiong *et al.*'s scheme.

2 Review of Xiong *et al.*'s Scheme

The entities in the scheme [35] comprises data owner, the authority, time server, cloud servers, users, potential adversary. It consists of four phases: Setup, Encryption, KeyGeneration and Decryption.

Setup. Let G be a bilinear group of prime order p, $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}'$ be a bilinear map, where $g \in \mathbb{G}$ is a generator. Let T be the maximum time in the system, where $|T| = n'$. Let $\mathcal{U} = \{1, \cdots, n\}$ be the universe of all attributes. The authority picks $y \in \mathbb{Z}_p$ and sets $g_1 = g^y$. Choose

$$g_2, u'_{1,1}, \cdots, u'_{n,1}, u'_{1,2}, \cdots, u'_{n,2}, u_1, \cdots, u_T \in \mathbb{G}.$$

Set public parameters as

$$g, g_1, g_2, u'_{1,1}, \cdots, u'_{n,1}, u'_{1,2}, \cdots, u'_{n,2}, u_1, \cdots, u_T.$$

The master key is set as $MSK = g_2^y$.

Encryption. To encrypt a message M under a set of attributes S_{att} with every attribute $i \in S_{att}$, where i is constrained by a time interval $T'_i \in [t_{m_{L,i}}, t_{m_{R,i}}]$ (the double-subscript notation $t_{m_{L,i}}$ indicates that the time is associated with the attribute hierarchy m and the concrete attribute i), the data owner picks $s \in \mathbb{Z}_p$, defines $c_{L,i} = n' - m_{L,i}$ and sets the ciphertext as

$$\begin{aligned} &S_{att}, C = g^s, C_M = M \cdot e(g, g_2)^{sy}, \\ &\{E = \left(u'_{i,1}\Pi_{j=1}^{m_{R,i}+1} u_j^{t_j}\right)^s, \\ &E' = \left(u'_{i,2}\Pi_{j=1}^{c_{L,i}} u_j^{T-t_j}\right)^s, T'_i\}_{i \in S_{att}} \end{aligned}$$

$\diamond$ *Notice that the encryption mechanism is well-defined because of*

$$C_M = M \cdot e(g_1, g_2^s) = M \cdot e(g^y, g_2^s) = M \cdot e(g, g_2)^{sy}.$$

That is, the encryptor can complete the phase by invoking the system's parameters and the picked exponent s.

KeyGeneration. For non-leaf node x in access tree Υ, the authority sets the degree d_x of the polynomial q_x and its threshold value k_x such that $d_x = k_x - 1$. For the root node r, set $q_r(0) = y$ and choose other d_r points to completely define the polynomial q_r. For any other node x, set

$$q_x(0) = q_{parent(x)}(index(x))$$

and pick d_x other points to define the polynomial q_x completely. Define a leaf node $x \in S_Y$ in the tree as an attribute which is constrained by a time instant t'_{n_x}, where S_Y denotes the leaf node set of Υ. Set the index $n_x = n' - c_x$.

The authority picks $r_x, r'_x \in \mathbb{Z}_p$, computes and sends the following secret key d to the user:

$$\begin{aligned} d = \{&D_{x,1}, D_{x,2}, g^{r_x}, g^{r'_x}, u_{n_x+2}^{r_x}, \cdots, u_T^{r_x}, \\ &u_{c_x+1}^{r'_x}, \cdots, u_T^{r'_x}, t_{n_x}\}_{x \in S_Y}, \end{aligned}$$

where

$$\begin{aligned} D_{x,1} &= g_2^{q_x(0)+\tau_x}\left(u'_{i,1}\Pi_{j=1}^{n_x+1} u_j^{t_j}\right)^{r_x} \\ D_{x,2} &= g_2^{-\tau_x}\left(u'_{i,2}\Pi_{j=1}^{c_x} u_j^{T-t_j}\right)^{r'_x} \end{aligned}$$

Decryption. This is a recursive algorithm from bottom to up, performed by the user. For a leaf node x: If $t_{n_x} \notin [t_{m_{L,x}}, t_{m_{R,x}}]$, the algorithm simply outputs $\perp$. Otherwise, it picks $r''_x, r'''_x \in \mathbb{Z}_p$ and computes

$$\begin{aligned} &\{a_0, g^{r_{R,x}} \cdot g^{r''_x}, u_{m_{R,x}+2}^{r_{R,x}} \cdot u_{m_{R,x}+2}^{r''_x}, \cdots, u_T^{r_{R,x}} \cdot u_T^{r''_x}\} \\ &\{b_0, g^{r_{L,x}} \cdot g^{r'''_x}, u_{c_{L,x}+1}^{r_{L,x}} \cdot u_{c_{L,x}+1}^{r'''_x}, \cdots, u_T^{r_{L,x}} \cdot u_T^{r'''_x}\} \end{aligned}$$

where

$$\begin{aligned} a_0 =& D_{x,1}(u'_{i,1}\Pi_{j=n_x+1}^{m_{R,x}+1} u_j^{t_j})^{r_{R,x}}(u'_{i,1}\Pi_{j=1}^{m_{R,x}+1} u_j^{t_j})^{r''_x} \\ =& g_2^{q_x(0)+\tau_x}(u'_{i,1}\Pi_{j=1}^{m_{R,x}+1} u_j^{t_j})^{r_{R,x}+r''_x} \\ b_0 =& D_{x,2}(u'_{i,2}\Pi_{j=c_x}^{c_{L,x}} u_j^{T-t_j})^{r_{L,x}}(u'_{i,2}\Pi_{j=1}^{c_{L,x}} u_j^{T-t_j})^{r'''_x} \\ =& g_2^{-\tau_x}(u'_{i,2}\Pi_{j=1}^{c_{L,x}} u_j^{T-t_j})^{r_{L,x}+r'''_x} \end{aligned}$$

It then calculates

$$DN = \frac{e(g^s, a_0) \cdot e(b_0, g^s)}{e(E, g^{r_{R,x}+r''_x}) \cdot e(g^{r_{L,x}+r'''_x}, E')} = e(g, g_2)^{sq_x(0)}.$$

For a non-leaf node x with all nodes z that are the children of x, use Lagrange's interpolation method to compute

$$\begin{aligned}F_x &= \prod_{c\in S_x}(e(g,g_2)^{sq_c(0)})^{\triangle_{i,S'_x}(0)}\\ &= \prod_{c\in S_x}(e(g,g_2)^{sq_{parent(c)}(0)})^{\triangle_{i,S'_x}(0)}\\ &= \prod_{c\in S_x}e(g,g_2)^{sq_x(i)\cdot\triangle_{i,S'_x}(0)} = e(g,g_2)^{sq_x(0)}\end{aligned}$$

Finally, for the root node r,

$$e(g,g_2)^{sq_r(0)} = e(g,g_2)^{sy}$$

can be recovered. It then computes

$$M = C_M/e(g,g_2)^{sy}.$$

3 Cryptanalysis

The Xiong *et al.*'s scheme involves lots of parameters and secret exponents. It tries to link time intervals to attributes and provides flexible access control strategy. But we find the scheme is flawed.

3.1 The Consistency Between Encryption Mechanism and Decryption Mechanism is not Kept

It is easy to find that

- The true *master key* is y, not g_2^y. In KeyGeneration phase, the authority has to directly invoke y and set $q_r(0) = y$. However, g_2^y is not invoked at all.
- It fails to check the consistency between encryption mechanism and decryption mechanism. Concretely, the user cannot finish the calculations of a_0, b_0 and DN. In fact,

$$\begin{aligned}a_0 =&D_{x,1}(u'_{i,1}\Pi_{j=n_x+1}^{m_{R,x}+1}u_j^{t_j})^{r_{R,x}}(u'_{i,1}\Pi_{j=1}^{m_{R,x}+1}u_j^{t_j})^{r''_x}\\ =&g_2^{q_x(0)+\tau_x}\left(u'_{i,1}\Pi_{j=1}^{n_x+1}u_j^{t_j}\right)^{r_x}\\ &\cdot(u'_{i,1}\Pi_{j=n_x+1}^{m_{R,x}+1}u_j^{t_j})^{r_{R,x}}(u'_{i,1}\Pi_{j=1}^{m_{R,x}+1}u_j^{t_j})^{r''_x}\\ \neq&g_2^{q_x(0)+\tau_x}(u'_{i,1}\Pi_{j=1}^{m_{R,x}+1}u_j^{t_j})^{r_{R,x}+r''_x},\end{aligned}$$

$$\begin{aligned}b_0 =&D_{x,2}(u'_{i,2}\Pi_{j=c_x}^{c_{L,x}}u_j^{T-t_j})^{r_{L,x}}(u'_{i,2}\Pi_{j=1}^{c_{L,x}}u_j^{T-t_j})^{r'''_x}\\ =&g_2^{-\tau_x}\left(u'_{i,2}\Pi_{j=1}^{c_x}u_j^{T-t_j}\right)^{r'_x}\\ &\cdot(u'_{i,2}\Pi_{j=c_x}^{c_{L,x}}u_j^{T-t_j})^{r_{L,x}}(u'_{i,2}\Pi_{j=1}^{c_{L,x}}u_j^{T-t_j})^{r'''_x}\\ \neq&g_2^{-\tau_x}(u'_{i,2}\Pi_{j=1}^{c_{L,x}}u_j^{T-t_j})^{r_{L,x}+r'''_x},\end{aligned}$$

$$\begin{aligned}DN =&\frac{e(g^s,a_0)\cdot e(b_0,g^s)}{e(E,g^{r_{R,x}+r''_x})\cdot e(g^{r_{L,x}+r'''_x},E')}\\ \neq&e(g,g_2)^{sq_x(0)},\end{aligned}$$

3.2 The Scheme cannot be Simply Revised

To revise the above equations, in KeyGeneration phase $D_{x,1}, D_{x,2}$ should be replaced by

$$\begin{aligned}D_{x,1} &= g_2^{q_x(0)+\tau_x}\left(u'_{i,1}\Pi_{j=1}^{n_x}u_j^{t_j}\right)^{r_x},\\ D_{x,2} &= g_2^{-\tau_x}\left(u'_{i,2}\Pi_{j=1}^{c_x-1}u_j^{T-t_j}\right)^{r'_x}.\end{aligned}$$

Besides, it should specify that

$$r_{R,x} = r_x, \quad r_{L,x} = r'_x.$$

◇ *Notice that in the simple revision the authority has to share the session exponents r_x, r'_x with the user.*

We now want to stress that the session exponents r_x, r'_x cannot be exposed to the user [19]. Otherwise, g_2^y will be exposed to the user (inner adversary) and the user can freely recover any ciphertext. In fact, the adversary can recover the *session key* $g_2^{q_x(0)}$ by calculating

$$\begin{aligned}g_2^{q_x(0)} =&D_{x,1}D_{x,2}\left(u'_{i,1}\Pi_{j=1}^{n_x+1}u_j^{t_j}\right)^{-r_x}\\ &\cdot\left(u'_{i,2}\Pi_{j=1}^{c_x}u_j^{T-t_j}\right)^{-r'_x}\end{aligned}$$

Consequently, the *secret key* $g_2^{q_r(0)} = g_2^y$ will be recovered. Once the adversary obtains g_2^y, he can recover the plaintext by computing

$$C_M/e(C,g_2^y) = M\cdot e(g,g_2)^{sy}/e(g^s,g_2^y) = M.$$

3.3 The Reason for Setting Lots of Parameters in the Scheme

In the past years, the general instruction for designing a new cryptographic scheme is to build the new on some preliminary schemes. Consequently, the method to introduce more parameters in a new scheme is broadly adopted. To achieve different purposes, it is usual to set different parameters *separately*. As a result, the whole scheme becomes gross and the consistency between different phases becomes difficult to check.

The Xiong *et al.*'s scheme combined many techniques developed in [15, 24, 34, 36]. It has to set lots of parameters, including that for representing the universe of all attributes, time intervals, access tree and its nodes, session key, secret key, and master key. Thus, it becomes more difficult to check the consistency as the quantity of parameters increases. Moreover, the security argument becomes gloomy, intricate and unintelligible. We would like to remark that designing a cryptographic scheme with all-sided characters is inadvisable in practice.

4 Conclusion

In this note, we show that Xiong *et al.*'s scheme is flawed. We want to stress that the concepts of session key, secret

key, and master key should be accurately specified. Moreover, the consistency in a cryptographic scheme must be checked carefully.

Acknowledgements

We thank the National Natural Science Foundation of China (Project 61303200, 61411146001). The authors gratefully acknowledge the reviewers for their valuable suggestions.

References

[1] N. Attrapadung and *et al.*, "Attribute-based encryption schemes with constant-size ciphertexts," *Theoretical Computer Sciences*, no. 422, pp. 15–38, 2012.

[2] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *Proceedings of IEEE Symposium on Security and Privacy (S&P'07)*, pp. 321–334, May 2007.

[3] Z. J. Cao and L. H. Liu, "Comment on 'harnessing the cloud for securely outsourcing large-scale systems of linear equations'," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 5, pp. 1551–1552, 2016.

[4] Z. J. Cao, L. H. Liu, and Z. Z. Guo, "Ruminations on attribute-based encryption," *International Journal of Electronics and Information Engineering*, vol. 8, no. 1, pp. 9–19, 2018.

[5] Z. J. Cao, L. H. Liu, and O. Markowitch, "Comment on 'highly efficient linear regression outsourcing to a cloud'," *IEEE Transactions on Cloud Computing*, pp. 1-1, 2017. DOI: 10.1109/TCC.2017.2709299

[6] A. F. Chan and I. F. Blake, "Scalable, server-passive, user-anonymous timed release cryptography," in *Proceedings of 25th International Conference on Distributed Computing Systems (ICDCS'05)*, pp. 504–513, June 2005.

[7] W. Y. Chao, C. Y. Tsai, and M. S. Hwang, "An improved key-management scheme for hierarchical access control," *International Journal of Network Security*, vol. 19, no. 4, pp. 639–643, 2017.

[8] M. Chase, "Multi-authority attribute based encryption," in *Proceedings of 4th Theory of Cryptography Conference (TCC'07)*, pp. 515–534, Feb. 2007.

[9] M. Chase and S. Chow, "Improving privacy and security in multi-authority attribute-based encryption," in *Proceedings of ACM Conference on Computer and Communications Security (CCS'09)*, pp. 121–130, Nov. 2009.

[10] F. Chen, T. Xiang, X. Lei, and J. Chen, "Highly efficient linear regression outsourcing to a cloud," *IEEE Transactions on Cloud Computing*, vol. 2, no. 4, pp. 499–508, 2014.

[11] F. Chen, T. Xiang, and Y. Y. Yang, "Privacy-preserving and verifiable protocols for scientific computation outsourcing to the cloud," *Journal of Parallel and Distributed Computing*, vol. 74, pp. 2141–2151, 2014.

[12] J. S. Chen, C. Y. Yang, and M. S. Hwang, "The capacity analysis in the secure cooperative communication system," *International Journal of Network Security*, vol. 19, no. 6, pp. 863–869, 2017.

[13] T. Y. Chen, C. C. Lee, M. S. Hwang, and J. K. Jan, "Towards secure and efficient user authentication scheme using smart card for multi-server environments," *The Journal of Supercomputing*, vol. 66, no. 2, pp. 1008–1032, 2013.

[14] P. S. Chung, C. W. Liu, and M. S. Hwang, "A study of attribute-based proxy re-encryption scheme in cloud environments", *International Journal of Network Security*, vol. 16, no. 1, pp. 1-13, 2014.

[15] R. Geambasu, T. Kohno, A. Levy, and H. M. Levy, "Vanish: Increasing data privacy with self-destructing data," in *Proceedings of 18th USENIX Security Symposium*, pp. 299–315, Aug. 2009.

[16] V. Goyal and *et al.*, "Bounded ciphertext policy attribute based encryption," in *Proceedings of 35th International Colloquium on Automata, Languages and Programming (ICALP'08)*, pp. 579–591, July 2008.

[17] S. Hohenberger and B. Waters, "Online/offline attribute-based encryption," in *Proceedings of 17th International Conference on Practice and Theory in Public-Key Cryptography (PKC'14)*, pp. 293–310, Mar. 2014.

[18] W. F. Hsien, C. C. Yang, and M. S. Hwang, "A survey of public auditing for secure data storage in cloud computing," *International Journal of Network Security*, vol. 18, no. 1, pp. 133–142, 2016.

[19] L. C. Huang, T. Y. Chang, and M. S. Hwang, "A conference key scheme based on the diffie-hellman key exchange," *International Journal of Network Security*, vol. 20, no. 6, pp. 1221–1226, 2018.

[20] M. S. Hwang, C. C. Lee, T. H. Sun, "Data error locations reported by public auditing in cloud storage service," *Automated Software Engineering*, vol. 21, no. 3, pp. 373–390, Sep. 2014.

[21] M. S. Hwang, T. H. Sun, "Using smart card to achieve a single sign-on for multiple cloud services," *IETE Technical Review*, vol. 30, no. 5, pp. 410–416, 2013.

[22] M. S. Hwang, T. H. Sun, C. C. Lee, "Achieving dynamic data guarantee and data confidentiality of public auditing in cloud storage service," *Journal of Circuits, Systems, and Computers*, vol. 26, no. 5, 2017.

[23] K. Kasamatsu and *et al.*, "Time-specific encryption from forwardsecure encryption," in *Proceedings of International Conference on Security and Cryptography for Networks*, pp. 184–204, Sep. 2012.

[24] R. Kikuchi, A. Fujioka, Y. Okamoto, and T. Saito, "Strong security notions for timed-release public-key encryption revisited," in *Proceedings of 14th International Conference on Information Security and Cryptology (ICISC'11)*, pp. 88–108, Nov. 2012.

[25] A. Lewko and B. Waters, "New proof methods for attribute-based encryption: Achieving full security through selective techniques," in *Proceedings of 32nd Annual Cryptology Conference, Advances in Cryptology (CRYPTO'12)*, pp. 180–198, Aug. 2012.

[26] C. W. Liu, W. F. Hsien, C. C. Yang, and M. S. Hwang, "A survey of attribute-based access control with user revocation in cloud data storage," *International Journal of Network Security*, vol. 18, no. 5, pp. 900–916, 2016.

[27] C. W. Liu, W. F. Hsien, C. C. Yang, and M. S. Hwang, "A survey of public auditing for shared data storage with user revocation in cloud computing", *International Journal of Network Security*, vol. 18, no. 4, pp. 650–666, 2016.

[28] L. Liu, Z. Cao, C. Mao, "A note on one outsourcing scheme for big data access control in cloud," *International Journal of Electronics and Information Engineering*, vol. 9, no. 1, pp. 29–35, 2018.

[29] R. Ostrovsky, A. Sahai, and B. Waters, "Attribute-based encryption with non-monotonic access structures," in *Proceedings of the ACM Conference on Computer and Communications Security (CCS'07)*, pp. 195–203, Oct. 2007.

[30] K. G. Paterson and E. A. Quaglia, "Time-specific encryption," in *Proceedings of 7th International Conference on Security and Cryptography for Networks (SCN'10)*, pp. 1–16, Sep. 2010.

[31] S. Rezaei, M. Ali Doostari, and M. Bayat, "A lightweight and efficient data sharing scheme for cloud computing," *International Journal of Electronics and Information Engineering*, vol. 9, no. 2, pp. 115–131, 2018.

[32] Y. L. Wang, J. J. Shen, and M. S. Hwang, "A survey of reversible data hiding for vq-compressed images," *International Journal of Network Security*, vol. 20, no. 1, pp. 1–8, 2018.

[33] B. Waters, "Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization," in *Proceedings of International Conference on Practice and Theory in Public-Key Cryptography (PKC'11)*, pp. 53–70, Mar. 2011.

[34] S. Wolchok and *et al.*, "Defeating vanish with low-cost sybil attacks against large dhts," in *Proceedings of the Network and Distributed System Security Symposium (NDSS'10)*, pp. 1–15, Mar. 2010.

[35] J. Xiong and *et al.*, "A secure data self-destructing scheme in cloud computing," *IEEE Transactions on Cloud Computing*, vol. 2, no. 4, pp. 448–458, 2014.

[36] J. Xiong and *et al.*, "A full lifecycle privacy protection scheme for sensitive data in cloud computing," *Peer-to-Peer Networking and Applications*, vol. 8, no. 6, pp. 1025–1037, 2015.

[37] S. Yamada and *et al.*, "A framework and compact constructions for non-monotonic attribute-based encryption," in *Proceedings of International Conference on Practice and Theory in Public-Key Cryptography (PKC'14)*, pp. 275–292, Mar. 2014.

[38] L. Zhou, V. Varadharajan, and M. Hitchens, "Achieving secure role-based access control on encrypted data in cloud storage," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 12, pp. 1947–1960, 2013.

Biography

Lihua Liu is an associate professor with the Department of Mathematics at Shanghai Maritime University. She received her Ph.D. degree in applied mathematics from Shanghai Jiao Tong University. Her research interests include combinatorics, cryptography and information security.

Yang Li is currently pursuing his M.S. degree from Department of Mathematics, Shanghai Maritime university. His research interests include combinatorics and cryptography.

Zhengjun Cao is an associate professor with the Department of Mathematics at Shanghai University. He received his Ph.D. degree in applied mathematics from Academy of Mathematics and Systems Science, Chinese Academy of Sciences. He served as a post-doctor in Computer Sciences Department, Université Libre de Bruxelles, from 2008 to 2010. His research interests include cryptography, discrete logarithms and quantum computation.

Zhen Chen is currently pursuing his M.S. degree from Department of Mathematics, Shanghai university. His research interests include information security and cryptography.

The Forensics of DDoS Attacks in the Fifth Generation Mobile Networks Based on Software-Defined Networks

Shahrzad Sedaghat

(Corresponding author: Shahrzad Sedaghat)

Faculty of Engineering Department, Jahrom University
Jahrom 7413188941, Iran
(Email: shsedaghat@jahromu.ac.ir)

(Received Apr. 3, 2018; Revised and Accepted Apr. 4, 2019; First Online Sept. 21, 2019)

Abstract

The number of devices connected to the Internet has been increasing with the emergence of the Internet of things technology. Although it has many advantages, the weak configuration of Internet of things devices and the higher number of such devices provide a good potential for DDoS (Distributed Denial-of-Service) attacks. In this study, an approach based on SDN (Software Defined Network) and NFV (Network Functions Virtualization) technologies were presented for the purpose of network forensics and DDoS attack detection. In this approach, the entropy-based methods were used as a warning for DDoS attacks. The methods of detecting the fake IP address of the message source and a method based on correlation coefficient were used for separating the legal traffic from allowed traffic from non-allowed traffic. In addition, NFV technology was used for allocating more resources dynamically.

Keywords: 5G; DDoS Attack; Forensic; SDN

1 Introduction

The emergence of the Internet of things has caused the communication range of mobile to spread from personal communications to smart communications among different things and people. The bursty growth of data traffic due to applications such as a smart house, smart vehicles, smart environmental monitoring, and the like require a huge number of devices connected to the Internet and continuous emergence of new services, requirements, and capacity beyond the presented facilities in the current generation of mobile networks. The fifth generation of mobile networks was aimed at removing the time and place limitation and providing an interactive user experience.

The bursty growth of data size caused some problems to their management. Thus, changing the architecture of traditional mobile networks to the fifth generation of software-defined mobile networks is a new technology, which is expected to meet the needs of users in the future. This type of network is designed by integrating the software-defined network and network functionality virtualization. These two technologies can complete each other, keep the network servicing at the busy time of network, control and manage the network, and remove the exclusive problem of network solutions. Thus, their application is highly significant for future networks.

Despite such new technologies and concepts, network security is considered as a big challenge for future networks. Network threats can be related to mobile networks and potential technologies, which should be used in the fifth generation of mobile networks. By considering the bursty size of data and a high number of the devices connected to the network, detecting the suspicious activities and making a decision about whether this activity is malicious on behalf of the attacker or has another reason is very difficult. If the occurred suspicious activity is a type of attack, detecting the attacker and his tool and method is problematic. Computer forensics is a service in the area of computer collecting data from computer equipment and digital media processing the collected data.

Thus, network forensics can be highly useful as detecting the attacks in the early steps is regarded as a very important factor for preventing action in the early steps and detecting the cause of attack, On the other hand, even if this issue cannot be effective in preventing the attack, having some evidence on the attack can be useful for the future allowed pursuits and the design of a mechanism to cope with similar attacks in the future [15].

Despite new technologies and concepts, network security is a major challenge for future networks. Network threats can also be due to the nature of mobile networks, and also due to potential technologies that should be used in fifth-generation mobile networks [6, 19]. Distributed denial-of-service attacks are one of the real threats to these networks, which can lead to a lot of destruction

in the IT infrastructure and communications. Therefore, any financial and governmental organization with vast infrastructure and information and communications resources is potentially exposed to this attack and it is necessary to find a way to respond to this type of attack by implementing a new and effective mechanism.

With this introduction, as well as considering the explosion of data volumes and the large number of devices connected to the network, it would be very difficult to detect suspicious activity and to decide whether this activity is an offensive act by an attacker or another reason. It is also difficult to determine who and by whom and with what method the crime was committed If suspicious activity is a kind of attack. Computer criminology is a science in the field of computer, in order to detect crime, collects evidence from proven scientific techniques for collecting, identifying, reviewing, combining, Correlation, analyzing and documenting evidence obtained uses processing, or Transfers digital resources [15]. Since the identifying attacks in the initial steps can be a very important factor in stopping it in the very first steps and identifying the cause of the attack, Therefore, the discovery of network crime can be very useful in this regard.

In this paper, our intention is to use criminology to detect the distributed denial-of-service attacks. The public aspect of the denial-of-service attacks (distributed) is sending large volumes of traffic to the network and saturation of its resources, which leads to the emergence of changes such as a sudden increase in traffic, delays in service delivery, excessive use of the processor and possibly reduced efficiency in the network activity pattern. Therefore, having evidence of changing the pattern of the network, analyzing this evidence and understanding its origins can be a great step to counteract or prevent this kind of attack [18]. Therefore, in the following, a solution is proposed for the purpose of the criminology of distributed denial-of-service attacks on fifth-generation mobile networks, with three general steps and follows the following goals:

1) Data collection: Detection of denial-of-service attacks be in different layers of software-based distributed networks, and therefore the processing burden resulting from this goal is not imposed on a specific layer.

2) Combine and correlate: In order to obtain the correct information, ensure the reliability of the data collection nodes.

3) Follows the analysis: More types of distributed denial-of-service attacks are covered.

This paper is organized as follows: The second section presents the significance of forensics. Section three explains the different types of DDoS attacks and the new types of attack in SDN networks. Section four explains the previous studies on discovering, preventing, and repairing the DDoS attacks. Section five describes the proposed strategy and section six presents the conclusion.

2 The Significance of Forensics and DDoS Attack

Network forensics is using the proved scientific technologies for collecting, identifying, studying, combining, analyzing, and documenting the evidence obtained from digital sources process or transfer. It is aimed at discovering the planned facts, measuring the success of allowed activities including sabotage or abuse of system components, and having sufficient information for responding to the malicious activity or improving the mechanisms and systems after each activity [15].

The present study aimed to use the forensics for DDoS attack criminology. The DDoS attack is a real threat for the network, digital, and security infrastructures, which can cause much destruction in the infrastructures of information technology and communications. These attacks are made due to financial- political benefits, or destruction. In the list of recent DDoS attacks, there are the websites of Greece bank, Ireland government, Polish Airlines, Thailand government, and Canada government. Thus, each organization including the financial and governmental with broad information and communications are potentially exposed to this attack looking for a way to respond to this type of attack by implementing a new and effective mechanism [5].

The general aspect of DDoS attacks is sending a bulk size of traffic to the Internet and saturating its sources, leading to some changes such as the sudden increase of traffic, delay in service delivery, excessive use of the processor, and reduction of efficiency in the network activity pattern. Thus, having some evidence for changing the network pattern, analyzing such evidence, and understanding its origin can be considered as a big step for coping with or preventing this attack [18].

By presenting this definition, the functionality of a forensic system can be expressed in three steps: Data collection, combination and correlation, and analysis.

The forensic network is mainly introduced as an observation point to a forensic system. In other words, the forensic system uses the network nodes for observing and recording the functionality and events of the network. Thus, the complete and accurate collected data depends on the reliability of the used nodes. In other words, the unreliable nodes cannot meet the complete observation of the network, which is the primary and important step of forensic systems. Thus, this case is one of the forensic challenges in the networks. An increase in analysis methods is regarded as another challenge in forensic systems. Trusted or not network nodes that in designed architecture called the virtual explorers, using the challenge-response algorithm, the network monitoring module is located. For this purpose, the module sends packets of challenge-response packets to data-level virtual explorers randomly or at certain times. These challenges can be small requests sent to the virtual explorers. The speed of the reaction of the virtual explorers to these pack-

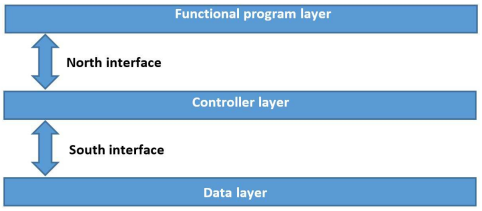


Figure 1: The layers of SDN networks

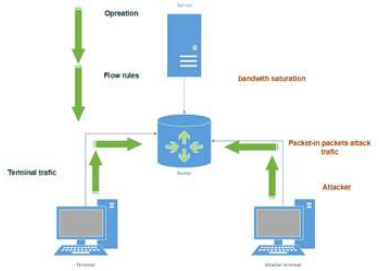


Figure 2: Broadband saturation attack in SDN

ages compared to the speed of this node to send recent reports as well as the completeness of the information received can provide an indication of the reliability of the virtual explorers.

3 Describe Different Steps of DDoS Attack and Introduce Similar Attacks

DDoS attacks prevent the access of allowed users to the computer, network, or information sources of the victim by using the distributed sources. Such distributed sources called "Zombie" are controlled by a manager and send some packets to the victim randomly or regularly when they receive an order from the manager. A type of classification for these types of attacks is displayed in Table 1.

3.1 New Types of DDoS Attack in SDN and 5G Networks

Since the software-defined networks are the inseparable parts of the fifth generation mobile networks, the attacks can use the wireless networks' nature and potential attractions of software-defined networks for making the DDoS attacks.

By considering the different layers of software-defined networks Figure 1, the data layer, communication interfaces among the controller, the surface and the control layer are the attractive parts for attackers to make DDoS attacks. The switches in the data layer adapt the header of each packet to the available rules in their rule table when they receive a packet. In the case of adaption, the packet is processed based on that rule, otherwise, a message called "packet-in" is sent to the controller through the south interface by using the fields inside the packet header and the controller defines a new rule to send to the switch.

In a type of DDoS attack, the attacker produces many packet-INS and saturates the south interface broadband by producing the new flow (*e.g.*, by randomizing a part of fields in packets' header) Figure 2. In addition, when the number of fake packets arrived to switch is excessive, the total switch memory is filled with the non-useful flows rule leading to discarding the new input packets.

The controller can have a special attraction for DDoS attack because it is related to other levels by using the north and south interfaces. Thus, the attacker can make the controller to define new rules by sending new flows to the data level or the functional programs can prevent the controller from addressing the important and allowed affairs by sending unnecessary feedbacks.

The new type of DDoS attacks in wireless networks is called "jamming" which reduces the signal rate to the noise in the receiver by sending the intervening wireless signals. The attack, which is an interventional interaction, can prevent or modify communications by intervening in the network physical equipment, network status, and network configuration.

4 Review of Literature

In this section, the mechanisms used for discovering, preventing, and restoring the DDoS attacks are explained. A review papers by Boani *et al.* [3] Classified the conducted studies into three groups of discovery, prevention, and restoration. Tables 2, 3, and 4 represent the classifications. Then, some studies were explained in more details, which may be placed in more than several groups. The review of these studies could provide us with a good perception of this subject.

4.1 Detailed Expression Some of Researches

Zhin *et al.* [8] suggested the use of TTL in the IP packets header for detecting the fake traffic. In the proposed algorithm, the number of steps between the sender and receiver was calculated by extracting the TTL field and the sender's IP address. Then, the calculated steps were studied to see whether they are correct or not. In case of a correct number of steps, the source IP address was considered as real and in case of incorrect number, the IP address was fabricated. In this study, a challenge was guessing the initial value of the adjusted TTL in the message source, which was derived in the sender by considering some hypotheses.

Tapengam *et al.* [16] proposed a method for recognizing the DDoS attacks in allowed congested networks. Based on the results of this study, the DDoS attacks today are mainly based on botnets, which are considered as automatic or semi-automatic methods running a program and

Table 1: Different types of DDoS attacks

Flood attacks based on feedback	
attack Smurf	A large number of ICMP packets with a fake IP address move towards the network leading to traffic congestion in the network.
attack Fraggle	This attack is like Smurf, but it uses UDP traffic instead of ICMP traffic to achieve the same goal.
Flood attacks based on protocol abuse	
Attack based on SYN packets	An attacker, by setting the SYN bit, sends a packet to the attacker in the TCP handshake. The victim responds to the attacker with the SYN-ACK packet and adds a record to his memory and assigns this connection. The attacker will never respond to this packet. Repeating these communications will result in the full memory of the victim and lack of response to allowed users.
Attack based on UDP fragmentation	In this attack, the attacker sends a few large UDP packets (over 1,500 bytes) in order to add more bandwidth. The victim resources are used to rebuild and collapse packages that cannot be assembled again.
Attack based on feedback and amplification	
The attack is based on DNS amplification	In the attack, the DMS small Zombies send a fake source IP address which generates a large amount of traffic to the victim
The attack is based on NTP amplification	The attack is also like DNS amplification but the packets are sent to NTP servers instead of being sent to the DNS server

acting based on a predictable model. On the other hand, the allowed users usually spend their time responding after their first request. For example, after showing the webpage to the user, he takes some time to respond by clicking on a link. In other words, the request rate of allowed users during a time period is unpredictable.

These studies divided the attacks into two groups of predictable and non-predictable rates. The attacks with predictable rate were divided into the constant rate, incremental rate, and periodic rate. Finally, this study tested the packet sending algorithm by using the packet receiving rate as the studied parameter and using the Pearson correlation coefficient as a mathematical model and then classified the network traffic into malicious and uncertain groups. Tapengam *et al.* studied the correlation coefficient between the packet input rate and their input time and analyzed the correlation between the packet input rates with each other. Presenting a good solution for detecting the botnets is regarded as one of the valuable aspects of this study. In addition, such a method can be used in any analytical module.

Tapengam *et al.* [17] presented an approach based on supervised machine learning which learns the behavioral pattern of the network sources by observing the input packets. The features of the data input rate were processed by using the Pearson correlation coefficient and Shannon entropy.

Then, these two features were classified by using the LDA (Linear discriminant analysis) into attack or allowed traffic groups. Accordingly, the traffic derived from the attack was filtered and the allowed attack was passed. The data provided to the LDA classifier were grouped into training data and test data. The detection power of the classifier was depended on training data. On the other hand, waiting for the training phase can cause delay and problem for using this method at real time.

Shin *et al.* [14] presented an approach called AVANT-GUARD for managing the flows in the software-defined networks based on the open flow. The communication immigration aimed to separate the sources enabling to complete the TCP handshaking protocol and the sources which could not perform it. The second module for activating without rule flow delay can be used under special conditions, which help the control level manage the network flows without delay. For this purpose, the controller specifies the conditions according which a warning should be issued and registers them in the switch. Each switch produces a warning based on this event and sends it to the controller in case of facing such conditions. By receiving this warning, the controller can change the rules of flow registered in the switches to control the current conditions. The main advantage of this study was separating the allowed traffic from the non-allowed traffic. However, in this study, only a specific type of DDoS was addressed. In addition, two additional modules should be added to SDN architecture for implementing the proposed method.

In another study, Lim *et al.* [9] presented a model for blocking the DDoS attacks, based on software-defined networks. In the architecture of this proposed model, they presented a module having a pool of IP addressed. Based on the packet-ins sent from data to the controller, this module monitored the number of flows in each switch. Simultaneously, the server monitored the parameters showing a DDoS attack. When the server found an attack, it

Table 2: DDoS discovery methods

Solution	Discovery methods
Entropy-based methods for detecting abnormal behavior depending on the distribution of network characteristics. Probabilistic distributions of various network feature such as the source and destination IP address and port number are used to calculate entropy. In order to determine whether current traffic behavior is normal or non-normal, a predefined threshold on entropy changes is used.	Entropy
Machine-based methods use techniques such as Bayesian networks, SOM and fuzzy logic to identify abnormal behavior. These algorithms pay attention to the different characteristics of the network and traffic to discover the abnormal behavior of the network.	Machine learning
These techniques work with the assumption that infected hosts behave differently from healthy hosts. Generally, botnets and infected machines (bats) are controlled by a single bat. The same traffic pattern is the result of sending a command to a large number of botnet host members, which results in the same behavior (sending illegal packets, starting to scan).	Traffic Pattern Analysis
These techniques are divided into two categories: 1) The successful communication rate 2) The rate of communication refers to the number of communications in a given time window.	Communication rate
These techniques use a combination of the intrusion detection system (such as SNORT) and Open Flow to detect attacks and re-configure the network dynamically. An intrusion detection system monitors network traffic to monitor suspicious activities, and Open Flow switches dynamically reconfigure the network based on real-world discoveries.	Integration of Open Flow with SNORT

Table 3: DDoS prevention methods

Solution	Prevention methods
In these solutions, a profile of users is kept, and the information header of each packet is checked and, and packages are prioritized to respond depending on the service previously agreed with the client.	Customizing the customer resources
In the mechanisms of this category, load balancing algorithms and virtual machines or virtual network functions are used to deal with a DDOS attack. In a number of studies, input traffic using load balancing algorithms is divided among several (virtual) factors. These virtual agents are allocated as needed and dynamically. The operating factor of the load balancing algorithm can also be a virtual factor.	Load balance
This mechanism is used to deal with DDOS attacks which generate fake traffic using fake IP addresses as the source address. In order to counteract this attack, they use the challenge-response mechanism. In this method, the server sends the packet to the originating IP address (sender of the message). If the sender is unable to respond to, it will be authorized as a fake source.	Factor detection mechanisms

Table 4: DDoS restoration methods

Solution	Restoration methods
Network traffic which satisfies the defined rules is sent and the rest of the traffic is discarded.	Discarding the packet
The network traffic sent by the attacked port is completely blocked.	Closing the port number
The allowed traffic changes to a new IP address.	Changing the route
The controller limits the flow rate by allocating the average bandwidth to each interface.	Controlling the broadband
The network controller changes the flow table of each switch to change the network topology.	Changing the network topology
When an attack is detected, the victim's MAC or IP address is changed, resulting in allowed traffic to the new address and the blocked traffic will be blocked.	Changing the MAC address or IP address

sent a message to the module through a secure channel and the module gave a new IP address to the server so that the server had to continue its work from that address. By using this module, the IP address of services was turned into a new address and a DDoS attack made by fabricating the source IP address failed. However, the allowed clients can search the new IP address from the server to achieve the service. In case of DDoS attacks based on bots which failed to fabricate the IP address, Lim *et al.* suggested the information message of new addresses in a form causing a huge computation to the client for understanding. For this purpose, Captcha was used in this model. However, each model causing difficulty for the bots to understanding the route change message will realize the objective. In the proposed model, it is assumed that the new addresses are all attributed to a similar physical machine by the SDN controller. When a client detects the bot or its fake IP address, a rule corresponding to the removal of current packets in the relevant flow should be sent to the switches.

This study could destroy the predetermined planning of the attacker by providing the strategy of changing the IP address. In addition, the address fabrication attacks were coped with the proposed model. These two mentioned factors could be considered as the strengths of this study.

Wang *et al.* [20] presented a model for coping with DDoS attacks to saturate the broadband between the data and controller. When the saturation attack was discovered, two new modules of flow rule analyst and data level cache were activated. All packet-ins were sent to cache in terms of attack detection, kept there temporarily, and sent gradually to the controller. The analyst module produced some rules as hyperactive based on the received packet-ins and installs them on the switches. Although this solution could help at the time of DDoS attacks, it failed to provide any solution for separating the allowed traffic from the non-allowed traffic.

Paydrahita *et al.* [12] presented the defensive system for DDoS attacks in software-defined networks. They identified the network traffic status by monitoring the output interfaces to inform the controller, which studied the network topography while receiving these warnings and ask the other routers to send their information. The routers responded to the controller by sending the source and destination IP addresses. Then, the controller avoided the saturation of broadband and lack of servicing the virtual flows between the attack flows by sharing the broadband fairly and fining the flows, which did not use this broadband appropriately. When the network was congested, the controller sent the necessary orders to the congested router and the routers, which were on the route of sending the packet to the congested router. In the case of non-congested routers, if the bit rate of sending the packet was less than the bit rate based on a fair system, that flow would be classified as good-behavior flow. However, if a flow used the broadband more than the specified threshold, the broadband would be less than the fair broadband and this reduced rate would be imposed as fine to this flow. This study prevented the inactivation of network services for allowed flows by sharing the broadband fairly while it failed to present any solution for separating the allowed flows from non-allowed flows. In addition, it could delay data collection and request the data from routers when the network was congested.

Turwald *et al.* [18] presented an entropy-based approach for discovering DDoS attacks in the software-defined networks of VANET (Vehicular ad-hoc networks). Entropy is a concept in the field of data flow measuring uncertainty about a random variable. If a random variable occurs more than the other variables, its entropy (uncertainty) becomes less. In this study, the investigated data were packet header. Some fields from packet header were considered as a random variable and the emergence of these variables and their entropy were calculated in a certain time period called "window". In this study, the source IP address was one of the random variables. The controller had to calculate the entropy of this variable by counting the number of sent packets towards a specific source. If the entropy exceeded the threshold, the packets would be random which can be due to a DDoS attack. Although using the entropy could be a good warning for the congested network, determining a threshold for entropy was challenging. In addition, entropy could provide us with useful information about the distribution of variables. In other words, the variables with different distribution but similar uncertainty were considered the same in this solution. This solution could separate the allowed traffic from non-allowed traffic.

In addition, Yan *et al.* [21] presented a plan for dealing with DDoS attacks, which led to excessive traffic load to switch level. In this plan, the switches of data level played a cooperative role for each other. In the proposed method, a controller monitored the status of switch flows table in terms of the used and non-used part. If the switch memory space was completely filled, the traffic sent to this switch would be guided towards other switches. Thus, the traffic was distributed throughout the network and new rules were included in the tables of networks switches flow. By this method, the empty sources of total network flow tables were used for reducing the attack.

Suggesting the status of switch memory status and guiding the traffic towards other switches can be performed by the controller at the time of studying the packet-ins. The status of switches could be studied periodically and actively in terms of memory consumption rate and then develop some rules by the controller. In addition, attempting for installing route change rules could not necessarily be made at the time of full switch memory and could be made sooner. This study kept the quality of service by distributing the sent traffic to the whole network but dealt similarly with flows including allowed or non-allowed.

Chawlo *et al.* [4] distinguished the traffic related to DDOS attacks from the huge but allowed traffic by using the Pearson correlation coefficient method. This method

was composed of discovery module and the traffic separation module. In the discovery module, the input traffic was sampled and the features of source and destination IP address, the number of packets in each time period, and a number of packets sent from each IP address were extracted.

If the number of packets at a time period of T was more than a certain amount, a warning signal would be sent to the separation traffic module. Otherwise, the traffic would be considered as allowed traffic. In the traffic separation module, the traffic correlation coefficient was calculated and the created traffic was classified into two groups of DDoS attack or the congested network due to allowed users depending on the calculated value. This study introduced the instant Pearson coefficient as the best option for detecting DDoS attacks by testing different correlation coefficients.

Alhabi *et al.* [2] used the NFV technology to present a smart method for discovering the DDoS attack. The present study aimed to identify all DDoS attack and a deep investigation were used to divide the traffic into allowed traffic and attack traffic. The approach presented in the two steps mentioned the input traffic in the two steps of quick observation and deep investigation. The deep investigation was only presented when the first step showed positive results or received the message from the source allocation protocol. ARP protocol was used for allocating new sources or reducing the allocated sources. This protocol sent a message to the coordinator module when the consumed memory was more than the specified value creating a virtual machine to keep servicing. In the quick observation step, the status was recognized suspicious and the deep investigation module was activated when the input traffic was more than the threshold value or a message was sent for detecting more sources. This module detected the type of DDoS attack by receiving the data like the used protocol, packet size, destination port number, and source IP address. For this purpose, some algorithms were designed in which the attack was detected depending on the value of protocols and packet size exponentially or linear increase of the flow.

Shang *et al.* [13] presented a plan for coping with DDoS attacks imposed on data and controller. In this plan, the network status including the packet, in message rate, memory, and processor were monitored continuously and dynamically. In case of detecting the traffic in a switch, a module called table-miss module sent some flow rules to the victim switch, upon which the switch should distribute a part of table-miss traffic among its neighboring switches. Protective rules were only applied to the changed traffic due to the probability of interference between these protective rules and flow rules inside the switch. The next module filtered the message in two steps. In the first step, the flows with the frequency rate more than the defined threshold were separated. In the second filter, some features such as the number of packets in a flow within a time interval, the number of bytes of a flow within a time interval, the total number of packets in the opposite flow in an interval, and the number of bytes in the opposite flow in an interval were extracted from these data and sent to the SVM (support-vector machines) for classifying this flow into two groups of normal and attack. This learner was resistant to learning the data with high noise. Thus, all packets were divided into two groups of attack and non-attack. Based on the two described filters, some rules were defined for including in the victim switch. The two-step filter was one of the strengths in this plan and separating the traffic in the second filter added to this strength. However, one of the concerns in the proposed approach was related to SVM learning to see if it can be used timely in the real world and its error detection rate.

Further, Jakarya *et al.* [7] presented a model for coping with DDoS attacks, in which the DDoS attacks based on the fabrication of source IP address during the implementation of handshaking protocol, was prevented by using the NFV (network function virtualization) technology. In this plan, there were two main modules each one allocated dynamically by using the NFV. In the first module, the received packets were distributed among some factors (the second module) responsible for filtering the packets. In the case of high traffic, the number of such factors increased. A load balance algorithm was used for distributing the packets. These factors test the clients implementing the TCP handshaking protocol with the server by sending the challenge-response packets. The strength of this study was using NFV technology for allocating the sources dynamically. This study only dealt with a specific type of DDoS attacks.

In another study, Afek *et al.* [1] evaluated a variety of methods for making fake packets and presented an approach based on the software-defined network to cope with the huge size of packets sent to the network. Verifying each packet was performed in each switch. In the proposed approach, using the total sources of the network was suggested to cope with this type of attack. For this purpose, some thresholds were defined and the network status was defined for the two parameters of rules memory capacity and processing parameter. If one or two parameters exceed the defined threshold in a switch, the controller will guide the network to the parts with fewer loads by defining the traffic rules sent to the bust part of the network. This study used the total network sources optimally by using the load balance algorithms to respond to the input traffic, which was one of the strengths of this study. However, it only dealt with a specific type of attacks and used the controller as the monitor of the network status, which was equal to imposing more processing loads to the controller.

Lyanaj *et al.* [10] presented a plan for monitoring the fifth generation mobile network based on software-defined networks. The architecture of this plan was in line with the architecture of SDN networks developing the SDN architecture at three levels for the network monitoring functionality. The monitor controller could perform the abilities of traffic monitoring, load balancing, and monitored

data accumulating. Thus, it could optimize data analysis based on security needs. This monitoring controller can be implemented by P2P hierarchical model. The south interface of this controller sent the controlling messages related to monitoring the monitor explorers at the data level. The explorers were the virtual machines collecting the behavioral information of network and could be dynamically allocated. Explorers could operate passively (only for data collection) or actively (for prevention, reduction, or correction). These explorers were managed by the module, which was a part of NFV responsible for their dynamic implementation and configuration. The dashboard was a functional program considered as the functional programs of SDN networks enabling the user to specify the objectives of the monitoring system. By assuming that the 5-the generation mobile networks will be integrated with SDN architecture, the strength of this plan was that this architecture could be exclusively used for security goals. In addition, the abilities of network virtualization could cause flexibility in this plan under different network situations.

Lopez *et al.* [11] presented a plan based on network status awareness to manage and respond to the fifth generation mobile networks' events. This module which was designed as layered and could be adapted to SDN and NFV technologies monitored the low-level parameters of the network behavior by the explorers based on virtual infrastructures. Such explorers were the NFV programs, which could be personalized for monitoring and placed in different network infrastructures based on the need. The data collected from these explorers were provided to the module in the higher layer, i.e. the monitoring and correcting module. This module could collect the network traffic data precisely with low processing overhead at the real time by using the described explorers. This module could control the allocation and access to explorers. Such explorers were accessible based on two scenarios: they sent a report in case of discovering an event or the monitoring module requested the data from the explorers. The higher level module and the analysis module identified the network status by using the parameters provided by the monitoring module. In fact, this module monitored the events and extracted the risks exposed to the network by receiving the reports sent from the monitoring module. A part of this module predicted the future events by using the collected data, discarded the collected data after a time interval (defined), and used the new data for prediction for compatibility with network status dynamism. In order to detect the errors and attacks, different methods of machine learning like Bayesian networks were suggested to correct the risk detection methods. The last module was responsible for defensive measures against the probable risks such as load balance and network traffic management. This module could potentially use a big reservoir of NFV programs to operate its measures.

Furthermore, Zhang *et al.* [22] represented a plan based on packet prioritization coping with the overhead attack of rule table in the data switches in SDN architecture. In this study, the switches periodically sent the messages including the new packets input size and the number of packets in each flow to the module. In this study, the attacker sent a lot of new flows with a few packets toward the victim to reduce the attack cost. After data collection, the flows were classified into two groups of good and bad based on the number of packets sent by a client and the number of flows in this client. Then, two values of high and low threshold were considered for this parameter. The value of this parameter for each flow belonged to one of the groups of higher than the high threshold, a value between the two thresholds, and a value less than the low threshold. Accordingly, this study received services based on its label. Thus, the switches serviced some packets with high priority in the congested network and they discarded the packets with low priority in case of excessive congestion.

4.2 A Summary of Previous Studies and Presenting Some Challenges

In the previous section, some studies were stated in more details to understand the proposed solutions precisely. Furthermore, their strengths and weaknesses were mentioned at the end of each study. Some necessities of the proposed method which were considered in the fifth generation mobile networks, as explained in the weakness part of previous studies, are described below:

1) Separating the allowed traffic from the non-allowed traffic is very difficult because of too much similarity between the non-traffic traffic behavior and allowed traffic. For example, complicated botnets can pass the DDOS attacks discovery mechanisms by imitating the allowed traffic pattern when the requests are high.

2) Keeping the error detection rate low may lead to low discovery rate so that the DDoS attacks which are slow can easily pass the solution of attack prevention.

3) The proposed plans are usually software and are not integrated into the proposed solutions for DDoS attacks of interference type.

4) The proposed methods only consider the external attacks and do not regard the internal malicious factors of the network, as one of the forensic challenges in the network.

5 The Proposed Plan Architecture

The present study aimed at using the forensics to cope with DDoS attacks occurring in the future generation mobile networks. Thus, the proposed method should include three steps of data collection, analysis, and decision-making.

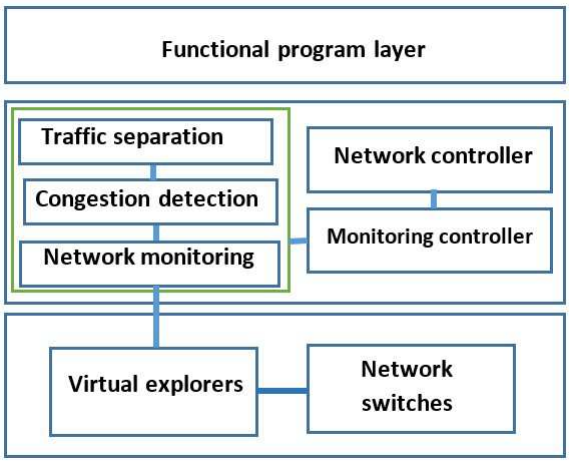


Figure 3: The proposed architecture

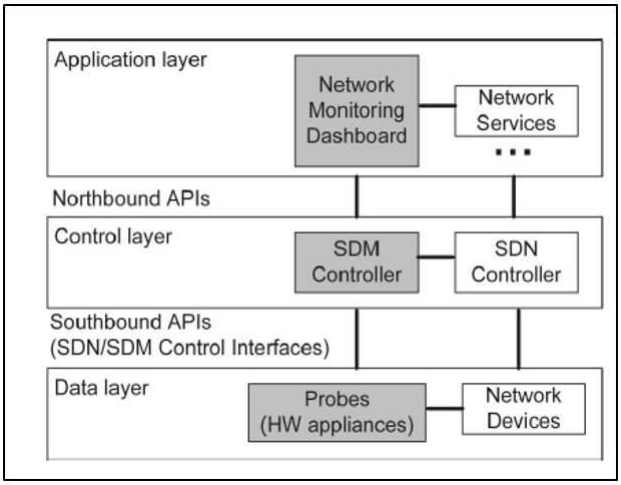


Figure 4: Research architecture

Since the SDN architecture is the inseparable part of the future generation networks, the proposed method is based on SDN and NFV technologies.

Here are the main expectations of a comprehensive solution for discovering, preventing, or restoring this type of attack:

1) DDoS protection mechanism should not disrupt the allowed traffic activities.

2) DDoS discovery mechanism should prevent both attacks inside or outside the networks.

3) The proposed mechanism should have efficiency and scalability.

4) The proposed mechanism should be implemented at a minimum cost.

5) It should include more types of these attacks.

This solution is based on the architectures [10, 11]. This architecture is composed of four functional layers: Virtual and real explorers, monitoring, and congestion detection, and allowed and non-allowed traffic separation. The last three modules were controlled by the high level of controller. The monitor controller shared its results and decisions to the network controller. In fact, the proposed method could be implemented as a security service in SDN architecture.

In the proposed architecture Figure 3, the statistical data were collected by the explorers. These data collected by the virtual explorer includes the packet arrival rate to the network, the origin, and destination IP address, the package type. This information is sent to the monitoring module. Each of the congestion detection and traffic separation modules use some of these information items to perform their task. The statistical information was sent to network monitoring, congestion detection, and traffic analysis modules. Congestion detection module studied the congestion in the network based on the threshold-based methods. If the input traffic exceeding a certain rate or the switch memory consumption is more than a certain limit, these situations will be sent to the monitor controller as a report based on congestion (allowed or non-allowed) which can also be sent to the network controller. Traffic separation module divided the input traffic into two groups of allowed and non-allowed and sent the obtained results to the monitor controller. The network controller could develop the appropriate rules in the network switches by receiving these data.

As described in Section 3, the DDoS attacks identification and monitoring section are divided into two data layers and controllers. The intermediate module, in fact, is a monitoring controller that consists of three network monitoring stations, congestion detection and traffic separation. These three parts are initially organized in sequence and then in parallel with each other. In other words, at first, the congestion detection module waits for information from the monitoring section and the traffic separation module waiting to apply for work from the congestion detection module. But after receiving the first information and requests, all three modules can be simultaneously in operation.

Authors in [1] provide a Software-Defined Networking-based solution that includes only DDoS attacks based on fake packets. However, in this research, it is suggested that along with using the method presented in [1], the Pearson coefficient method [4] used to Detection of distributed denial of service attacks Which is done using real clients (and not fake ones). In addition, in [1], monitoring and network identification are assigned to controllers. In the present study, the monitoring and security monitoring work on network inputs is assigned to separate modules from the switch controller, thus reducing the processor load imposed on the controller. For example, the monitoring module is in the form of a separate module in the controller layer and it receives its information from the nodes of the explorer in the data layer.

Authors in [10] with aims to provide more sophisticated and dynamic management functions, an SDN-based design for monitoring Figure 4 of the fifth-generation mobile networks. In the present study, it is suggested that architecture [10] In particular, it is used to identify DDoS attacks. For this purpose, the middle section of this architecture (SDM Controller), which is called monitoring controller in the present, is composed of three monitoring modules, congestion detection and traffic separation based on its specific task, namely identification of DDoS attacks. Data surface virtual explorers are also used in both surveys to collect information and send data to the middle section. In the present study, the assumption of the inadmissibility of these virtual explorer is also proposed, and it is suggested that the mid-section of the network, using the challenge-response algorithms, measure their accuracy.

5.1 The Variables Describing the Network Status

The present study aimed at detecting the DDoS attacks. The different methods of detecting such an attack considered different features of network flows to detect DDoS attacks. In this study, the features presented in Table 5 were suggested by considering some aspects of DDoS attacks.

5.2 Data Collection and Virtual Explorers

The virtual explorers continuously collect traffic log data into the network, source IP address, package type, and packet arrival rate, and send it to the monitoring module. Explorer nodes are the virtual or real machines at data level monitoring a small limit. These machines send the features of packets or input flows to the network in the form of some reports to the monitoring module. These reports were suggested to be sent to the monitoring module after entering a certain amount of packets. The questions raised in this module are as follows:

1) Where the explorers should be placed? In other words, how should the explorers be distributed? As mentioned, these explorers can be both real and virtual. Implementing the real explorers is optimal only in congested places near the entry gates or the servers providing important or interesting services to the client. In addition, at least one virtual explorer should be used in such places to ensure the accuracy of collected data and be considered as an auxiliary source at the time of network congestion.

2) What and when should the explorer's report? These explorers must monitor the described variables in Section 5.1 and after collecting the data about a certain amount of packets, they can prepare a report and send to the monitoring module. Such reports can be on packet or deduction of flow and can be prepared after a time period or certain number and sent to the monitoring module at the controller layer. The monitoring packets sent by the explorer must include the time cache which can provide the monitoring module with updated information and reflect the dynamic changes of the network.

5.3 Network Monitoring and Data Collection

The network monitoring module sends information from virtual explores to the congestion detection module. This module also has the task of organizing the virtual explores. More precisely, the organization of the virtual explorers includes two parts of the new virtual explorers' allocation and their verification.

About task new virtual explorers' allocation, the monitoring module can Do this in each of the following conditions:

1) Declare network congestion by congestion detection module;

2) Increase the number of reports received from virtual explorers within a specified time period;

3) Specific times the network administrator guesses that traffic congestion will be higher.

Regarding the task of verifying the virtual explorer nodes, the monitoring module is organized regularly or at random times, by sending a packet to a virtual explorer node that contains a challenge based on a challenge-response algorithm and receiving a response and a review of the malice/non-malice of these nodes Makes sure. This challenge can be to send multiple small random packets to the virtual explorers and then examine the reports received by the virtual explorers from these packages.

In the other words, the main responsibility of the network monitoring module is related to collecting data from explorers and sending them to the data combination module. The monitoring module can organize the explorers by involving the task of allocating a new explorer and verifying the explorer nodes. Obviously, the number of packets increases at the time of network congestion and the full monitoring of the network needs more monitoring sources. Detecting the network congestion status can be announced by the congestion detection module or show the network congestion through increasing the number of reports received from explorers at a certain time period. In addition, the network manager can predict more traffic congestion at certain times, which can announce the need for more allocation to the monitoring module through the monitoring controller.

The second task of this module is related to ensuring the accuracy of virtual or real nodes of an explorer, which could be used for responding to the fourth requirements in Section 2. For this purpose, this module could send some challenge-response packets to the data level explorers randomly or at certain times. Here, because the goal is

Table 5: The variables used for describing the network status

The features describing the aspects of DDOS attack	**The considered aspect**
The amount of memory used in the rules table and the queue memory per switch, the packet-in packet rates, the traffic log rate to the network, the entropy of the origin and destination IP address, the type of packet entropy	Congestion in the network and memory overflow in the switch
The IP address and package type	Fake source IP address
Packet Entry Rate and Source IP Address	Botnets

to ensure the accuracy of the virtual explorer and to send as fast and complete information as possible beforehand, these packets can pack small packages with a random IP address and add the legal authority to report these packages to the monitoring module. Then monitoring module can conclude on the accuracy of the virtual explorers by examining the complete report of packet data as well as the speed of this report compared to recent reports. These challenges should be designed in such a way to apply low processing overhead to the explorers. These challenges include the implementation of authentication protocols or requesting a report on a switch. Comparing the new report to the current information could ensure us on the accuracy of the explorer.

5.4 Congestion Detection

The congestion detection module is a research-based [18] that uses entropy to detect congestion. For this purpose, the destination IP addresses of the packets are considered as random variables. It also obtains information about the amount of memory consumed through the monitoring controller that it receives from the network controller. If the variable entropy value of the IP address of the packet destination exceeds a specified value or the amount of memory utilized by the switch is greater than a certain amount, this situation is considered as a bust condition. In case of congestion detection, this module also informs the controller and then informs the network controller and sends the information received from the monitoring module to the traffic separation module to decide whether a DDoS attack occurs/does not occur.

Congestion detection uses two methods to examine the memory consumption of controller-level switches as well as random variable entropy using the IP address of the destination of packet inputs to the network. If the variable entropy value of the IP address of the packet destination exceeds a specified value or the amount of memory utilized by the switch is greater than a certain amount, this situation is considered as a bust condition. In case of congestion detection, this module also informs the controller and then informs the network controller and sends the information received from the monitoring module to the traffic separation module to decide whether a DDoS attack occurs/does not occur.

5.5 Traffic Separation

If requested by the congestion detection module, this module will begin its work and uses the following two methods to identify the streams associated with the DDoS attack. (The explanation of these two methods is explained in more detail below.)

1) Investigating the fakes of source IP addresses [1]: To detect DDoS attacks that are falsified by source IP addresses.

2) Using the Pearson Correlation Coefficient [4]: To detect botnet-based DDoS attacks with real clients.

Packet information is tested in two ways, and if one of the above methods detects inbound inputs, then the current stream is reported as an attack on the controller and then to the network controller.

The Traffic Separation Module will work if the congestion detection module detects congestion on the network. As described above, the congestion detection module, if the variable entropy of the source IP of the packets exceeds a certain rate, or if the use of the switch memory exceeds a certain limit, consider the network status congestion and by sending information Receiving a traffic separation module from the monitoring module to the traffic separation module requires a breakdown of traffic.

The method of examining the falsification of source IP addresses in a handshaking phase by sending a cookie that is not kept on the server side and after completing the handshaking phase by sending specific messages to the client and examining his reaction to these two challenges about the actual or fake address origin of IP decisions. Pearson's method also examines features such as the origin and destination IP address, the number of packets per time interval, and the number of packets sent from each IP address, and if the volume of traffic sent in a stream exceeds the threshold value for a normal traffic flow, this flow is considered a suspicious flow. Then, the correlation coefficient between the currents is calculated and if this value is greater than the threshold of the expected difference for the currents, it is concluded that the denial of the distribution service has occurred and that the flows x and y are due to the similarity of each other among the strikes they take.

Fake checking method for source IP addresses: This research uses the SYN Cookie method with one of the following ways to ensure the authenticity of the source IP address. In the SYN Cookie, server or client-interacting device, in the SYN-ACK response message in the TCP handwriting process, a challenge that is not kept on the server side encodes in the sequence number section. If the client succeeds with an ACK response message containing the correct ACK number, then the server will continue to communicate with the client. After successfully completing this step, the server verifies that the client is the one who is in the following ways (Of course, in the original research, there are four methods out of them, two methods below are ours).

HTTP Redirect: In this way, during the TCP handshaking process, the client request receiver records the client IP address as a legitimate address, but in the response message, the header activates the redirect server address and then terminates the connection. Sending a response packet with the redirect header to the client will cause the real client to re-connect to the server, in which case it will communicate directly with the server. Sending a response packet with the redirect header to the client will cause the real client to re-connect to the server, in which case it will communicate directly with the server.

TCP Reset: This method is similar to the HTTP-Redirect method, with the difference that the server responds to the client with an RST packet, and thus the real client starts a new connection when the connection is disconnected, in which case the real client is detected.

Method of using the pearson correlation] coefficient: The research, using the Pearson correlation coefficient, attempts to detect the traffic associated with distributed denial-of-service denials from high-traffic but legal traffic. The strategy of this research consists of two detection modules and a traffic separation module. In the discovery module, the input traffic is sampled and features such as the origin and destination IP address, the number of packets per time interval, and the number of packets sent from each IP address are extracted. If the number of packets in a T interval is greater than a specified value, an alert signal will be sent to the traffic separation module. Otherwise, traffic is considered legal traffic. In the traffic separation module, the traffic correlation coefficient is calculated and, depending on the calculated value, the traffic generated in the two categories of DDoS attack or the bustle of legal users is divided. This study, by examining different correlation coefficients, introduced the momentary Pearson coefficient as the best option for detecting distributed denial-of-service attacks.

Then, the key variables of the network status, which could be a sign of DDoS attacks, were detected and the tasks of each layer were explained. We will also make suggestions to meet any of the requirements described in this section. The strengths of the suggested solution include:

1) In order to Separate legal traffic from illegal traffic has been used in two different ways in the Traffic Separation module. With the goal, identifying fake packages from real traffic, identifying botnets by examining the pattern of packet access to the network, have been used the proposed methods in [1,4]. Using these two methods simultaneously can cover a variety of attacks.

2) The nodes that work for data-level monitoring are validated using the challenge-response solutions by the monitoring module. By doing so, you can identify malicious internal agents and ensure the accuracy of the information collected.

3) The use of entropy-based methods in the congestion detection Module is much faster than traffic separation methods, and are considered as good alerts for the likelihood of an attack.

6 Conclusion

In the present study, a solution based on SDN and NFV technologies was presented to perform network forensic and detect the DDoS attacks. In this strategy, the entropy-based methods were used as a warning for DDoS attacks and the two methods presented in previous studies were used for separating the allowed traffic from the non-allowed traffic. In addition, NFV technology was used for allocating more sources dynamically. These sources were used for monitoring the network activities and extracting the data for congestion detection and traffic separation modules.

References

[1] Y. Afek, A. Bremler-Barr and L. Shafir, "Network anti-spoofing with SDN data plane," in *Proceedings of IEEE Conference on Computer Communications*, 2017. (`http://www.deepness-lab.org/pubs/infocom17_spoofing.pdf`)

[2] T. Alharbi, A. Aljuhani, H. Liu, "Smart and lightweight DDoS detection using NFV," in *Proceedings of ICCDA*, 2017.

[3] N. Bawany, J. Shamsi and K. Salah, "DDoS attack detection and mitigation using SDN: Methods, practices, and solutions," *Computer Engineering and Computer Science*, vol. 42, no. 2, pp. 425-441, 2017.

[4] S. Chawla, M. Sachdeva and S. Behal, "Discrimination of DDoS attacks and flash events using pearson's

product moment correlation methods," *International Journal of Computer Science and Information Security*, vol. 19, no. 5, pp. 734-741, 2016.

[5] M. S. Hwang, S. K. Chong and T. Yu. Chen, "DoS-resistant ID-based password authentication scheme using smart cards," *Journal of Systems and Software*, vol. 83, pp. 163-172, Jan. 2010.

[6] S. Islam, H. Ali, A. Habib, N. Nobi, M. Alam, and D. Hossain, "Threat minimization by design and deployment of secured networking model," *International Journal of Electronics and Information Engineering*, vol. 8, no. 2, pp. 135–144, 2018.

[7] A. Jakar, B. Rashidi, M. Rahman, C. Fung and W. Yang, "Dynamic DDoS defense resource allocation using network function virtualization," in *Proceedings of the ACM International Workshop on Security in Software Defined Networks & Network Function Virtualization*, 2017. DOI: 10.1145/3040992.3041000.

[8] C. Jin, H. Wang and K. Shang, "Hop-count filtering: An effective defense against spoofed traffic," in *Proceedings of the 10th ACM Conference on Computer and Communications Security*, pp. 30-41, 2003.

[9] S. Lim, J. Ha, H. Kim, Y. Kim and S. Yang, "A SDN-Oriented DDoS blocking scheme for botnet-based attacks," in *Proceedings of Sixth International Conference on Ubiquitous and Future Networks*, 2014. DOI: 10.1109/ICUFN.2014.6876752.

[10] M. Liyanage, J. Okwuibe, I. Ahmed and M. Yliant-tila, "Software defined monitoring (SDM) for 5G mobile backhaul networks," in *Proceedings of IEEE International Symposium on Local and Metropolitan Area Networks*, 2017. (`http://jultika.oulu.fi/files/nbnfi-fe2018080733468.pdf`)

[11] L. López, A. Caraguay, J. Vida, M. Monge and L. Villalba, "Towards incidence management in 5G based on situational awarness," *Future Internet*, vol. 9, no. 1, pp. 3, 2017.

[12] A. Piedrahita, S. Rueda, D. Mattos and O. Carlos, "Flowfence: A denial of service defense system for software defined networking," in *Proceedings of Global Information Infrastructure and Networking Symposium*, 2015. DOI :10.1109/GIIS.2015.7347185.

[13] G. Shang, P. Zhe, X. Bin, H. Aiqun and R. Ku, "FloodDefender: Protecting data and control plane resources under SDN-aimed DoS attacks," in *Proceedings of IEEE Conference on Computer Communications*, 2017. DOI: 10.1109/INFOCOM.2017.8057009.

[14] S. Shin, V. Yegneswaran, P. Porras and G. Gu, "AVANT-GUARD: Scalable and vigilant switch flow management in software-defined networks," in *Proceedings of the ACM SIGSAC Conference on Computer & Communications Security*, 2013. (`http://faculty.cs.tamu.edu/guofei/paper/AvantGuard-CCS13.pdf`)

[15] D. Spiekermann and T. Eggendorfer, *Challenges of Network Forensic Investigation in Virtual Networks*, 2017. DOI: 10.13052/jcsm2245-1439.522.

[16] T. Thapngam, S. Yu, W. Zhou and G. Beliakov, "Discriminating DDoS attack traffic from flash crowd through packet arrival patterns," in *Proceedings of IEEE Conference on Computer Communications Workshops*, 2011. (`http://cse.unl.edu/~byrav/INFOCOM2011/workshops/papers/p969-thapngam.pdf`)

[17] T. Thapngam, S. Yu and W. Zhou, "DDoS discrimination by linear discriminant analysis (LDA)," in *International Conference on Computing, Networking and Communications (ICNC'12)*, 2012. DOI: 10.1109/ICCNC.2012.6167480.

[18] M. Todorova and S. Tomova, *DDoS Attack Detection in SDN-based VANET*, 2016. (`https://projekter.aau.dk/projekter/en/studentthesis/ddos-attack-detection-in-sdnbased-vanet-architectures(11020a55-4287-4d8c-a603-b85c8969d9ca).html`)

[19] A. Tayal, N. Mishra and S. Sharma, "Active monitoring & postmortem forensic analysis of network threats: A survey," *International Journal of Electronics and Information Engineering*, vol. 6, no. 1, pp. 49–59, 2017.

[20] H. Wang, L. Xu and G. Gu, "FloodGuard: A DoS attack prevention extension in software-defined Networks," in *Proceedings of the 45th Annual IEEE/IFIP International Conference on Dependable Systems and Networks*, 2015. DOI: 10.1109/DSN.2015.27.

[21] B. Yuan, D. Zou and S. Yu, "Defending against flow table overloading attack in software-defined networks," *IEEE Transactions on Services Computing*, pp. 231-246, 2016.

[22] M. Zhang, J. Bi, J. Bai, Z. Dong, Y. Li and Z. Li, "FTGuard: A priority-aware strategy against the flow table overflow attack in SDN," in *Proceedings of the SIGCOMM Posters and Demos*, 2017. (`http://netarchlab.tsinghua.edu.cn/~junbi/SIGCOMM2017-3.pdf`)

Biography

Shahrzad Sedaghat received her B.Sc. and M.Sc. degrees in Computer and Information Technology engineering from Yazd University, Iran in 2008 and 2010 respectively, and is currently pursuing her Ph.D. in computer engineering in Sharif University of Technology, Iran. In 2011, she joined the department of computer and information technology engineering, Jahrom University. Her research interests are computer network and security, quality of service and reliability modeling.

A Formal Framework of Shielding Systems by Stepwise Refinement

Jiabin Zhu[1], Wenchao Huang[1], Fuyou Miao[1], Cheng Su[1], Baohua Zhao[2], and Yan Xiong[1]
(Corresponding author: Wenchao Huang)
School of Computer Science and Technology, University of Science and Technology of China[1]
The Library of West Campus of USTC, Huang Shan Road, Hefei, Anhui Province, China
Global Energy Interconnection Research Institute[2]
Global Energy Interconnection Research Institute, Binhe Road, Beijing, China
(Email: huangwc@ustc.edu.cn)
(Received May 31, 2018; Revised and Accepted Oct. 18, 2018; First Online July 16, 2019)

Abstract

The shielding systems, *e.g.*, special-purpose hypervisor, provide more secure environments for security-critical applications (SCAs), compared with traditional computer systems. In this paper, we propose a general framework of formally modeling and verifying the shielding systems for enhancing the security. The framework supports multiples types of shielding systems based on different technologies, such as Intel TXT or TrustZone. It is implemented by stepwise refinement, in which the early steps model the common states, events and security properties among the systems. Then the shielding systems are modeled in latter steps, where all the events are refined from the ones in the previous steps without the requirement of reproving soundness of security properties, *e.g.*, memory isolation, data confidentiality, upon the occurrence of each event. Therefore, the complexity of formally verifying new shielding systems is reduced. We implement the framework in the Coq proof assistant, and find potential security threats in using the shielding systems.

Keywords: Formal Methods; Framework; Security Analysis; Shielding Systems

1 Introduction

Recently, many shielding systems [5, 10, 11, 14, 15, 18, 20, 21], *e.g.*, special-purpose hypervisor, have been proposed to enhance security upon traditional computer systems. It is achieved by leveraging instructions of modern CPUs, *e.g.*, Intel TXT [17], TrustZone [1], for launching security-critical applications (SCAs), which run in isolation with legacy OS. For instance, a shielding system may keep the OS from accessing memory being used by SCAs. As a result, the security of SCAs no longer relies on the OS, but only the shielding system. Since the OS is error-prone in design due to its large size, it greatly reduces risks of security breaches by adopting a much smaller shielding system as the substitute of Trusted Computing Base (TCB).

It also requires to formally prove that a shielding system satisfies the security properties, *e.g.*, memory isolation and confidentiality, to achieve enhanced security. As the shielding system provides calling interfaces to the OS and SCAs, it should formally guarantee that both the caller, *i.e.*, SCA or OS, and the callee, *i.e.*, shielding system, run at expected states without any security breaches. The designers of SCAs may be unfamiliar with the shielding system, thus the properties may be broken when the interfaces are called at inappropriate states. On the other hand, since few source codes of shielding systems can be obtained online along with the literatures, it is urgent to re-implement the systems without flaws.

We propose and implement a general framework for formally verifying shielding systems. The framework supports various types of shielding systems, which may use different techniques, *e.g.*, hypervisor or TrustZone. To reduce the complexity of developing different systems, we model and verify the systems by stepwise refinement. Each refinement step is composed of states, events with action and guard, and invariants. The invariants should be preserved whenever any event occurs. In early steps of refinement, we model the internal processes of shielding systems, and prove the invariants of security properties, *e.g.*, memory isolation and data confidentiality. In latter steps, the processes are refined and merged into interfaces provided to the SCAs and the OS. The properties in the steps are still preserved by proving that the interfaces are correctly refined from previous processes. Therefore, the early steps can be reused when verifying new shielding systems, which reduce the workload of modeling and verifying.

The framework contains the following refinement steps as the general model for all shielding systems:

S0 *Abstract specification of memory isolation*: The

model's state only contains sufficient structure for proving memory isolation in shielding systems.

S1 *Memory isolation with multi-core support*: The model considers the case that multiple entities, *e.g.*, SCAs, use different cores of CPU simultaneously. We prove the soundness of memory isolation in this case.

S2 *Data confidentiality*: The states and events are refined for proving data confidentiality. Besides the addition of relevant states and events, we introduce an adversary model in Dolev-Yao style [12], and then analyze whether the private data may be leaked under the model.

In the case studies, we model two typical shielding systems, TrustVisor [21] and OSP [11], which use different technologies, *i.e.*, virtualization extensions and TrustZone, respectively. Moreover, TrustVisor only uses a single CPU core, while OSP may use several CPU cores. Faced with the differences, both systems can be successfully refined from our general framework. Specifically, as the shielding systems provide calling interfaces, each callee is divided into several events, which are refined from the events in **S2**.

We implement the framework and verify the cases by the Coq proof assistant [13] with the theory of refinement that borrows elements from [3, 7]. The results show that by using the framework, the complexity in modeling and verifying shielding systems is reduced. We also find potential security threats in using TrustVisor and OSP.

The paper is organized as follows. We provide preliminaries in Section 2. We introduce the refinement framework in Section 3. We propose the general model of the framework in Section 4, and discuss how to use the model to analyze different shielding systems in Section 5. We review some related work in Section 6. Finally, we conclude the paper in Section 7.

2 Preliminaries

2.1 Virtualization Extensions and TrustZone

We summarize recent technologies used for memory isolation, including virtualization extensions [29, 30] and the TrustZone [1], which are modeled and verified in our refinement.

2.1.1 Virtualization Extensions

Virtualization extensions enable an OS to execute in a virtual machine which offers virtual system resources. Specifically, the OS in a virtual machine executes on a virtual version (called host virtual memory) of real physical memory. Hardware components called MMUs in the CPUs that support the extensions can be configured to provide host virtual memory, and data structures called nested page tables are used in the MMUs to translate the host virtual memory addresses to host physical memory addresses.

2.1.2 TrustZone

ARM TrustZone technology [1] offers an isolated execution environment for security programs. Specifically, it partitions system resources into a normal world and a secure world, and prevents programs, *e.g.*, the untrusted OS, executed in the normal world to access the resources in the secure world. For partitioning memory, which is one of the resources, ARM TrustZone provides SoC peripherals called TrustZone Address Space Controllers (TZASCs). Configurations of TZASCs determine memory regions in the secure world, and the configurations can be changed by privileged software executed in the secure world.

2.2 Notations

We use standard notation for equality and logical connectives [3]. We extensively use record types and enumerated types. Record types are defined with the form $rec \stackrel{def}{=} \{l_1: T_1, \ldots, l_n: T_n\}$, and therefore elements of the types are of the form $\langle t_1, \ldots, t_n \rangle$. On extending *rec* with a component $l_{n+1} : T_{n+1}$, we define $erec = rec + l_{n+1} : T_{n+1} = \{l_1: T_1, \ldots, l_{n+1}: T_{n+1}\}$. Accordingly, on reducing *rec* with the component l_k, we define $rrec = rec - l_k$ ($1 \leq k \leq n$). Enumerated types are defined by using Haskell-like notation; for example, we define for every type T the type $option\ T \stackrel{def}{=} NONE \mid Some\ (t: T)$. The *option* T is the extension of T with an element *None*. Note that *NONE* has a polymorphic type, and a detailed explanation can be found in the manual of Coq [13] for polymorphic type. We define $T\ set$ as the type of sets over T. Then, we make an extensive use of maps: the type of maps from objects of type A into objects of type B is written $A \mapsto B$. Application of a map m on an object of type a is denoted as $m(a)$ and map update is written as $m(a) := b$, where b overwrites the value associated to a. Finally, we use the notation $\mathsf{let}\ a = b\ \mathsf{in}\ c$ to simplify our expressions of events, invariants, and so on. For example, the $(s.k).t1 + (s.k).t2$ may be expressed as $\mathsf{let}\ b = (s.k)\ \mathsf{in}\ b.t1 + b.t2$.

3 Refinement in Coq

Before we introduce the refinement framework, we summarize our theory of refinement that we developed in the Coq proof assistant [13]. The theory borrows elements of refinement from Event-B [7].

3.1 Models for Refinement

Our models are state machines which consist of states and events that result in the transition of states. The goal of machines is to prove soundness of invariants, where the

invariants preserve security properties for all states in the machines.

3.1.1 States

We model states of a machine by a record, which is a tuple of state variables. The machine starts at an initial state, and the state changes when an event occurs.

3.1.2 Events

We define an event by using 2 propositions: A conjunction of guards and an action.

$$\mathbf{Grds}_e(p_e, s) \wedge \mathbf{Act}_e(p_e, s, s') \tag{1}$$

Here, the state is transformed from s to s', if the event occurs. Denote p_e as the set of parameters of the event. The event can occur if and only if the conjunction $\mathbf{Grds}_e$ is true. The action $\mathbf{Act}_e$ describes the relation between s and s'. Hence, we can use $\mathbf{Act}_e$ to indicate the changes of states after the occurrence of the event, *e.g.*, changes of the value of a state variable. For convenience, we use the notation $s \sim_{c_1,\ldots,c_n=v_1,\ldots,v_n} s'$ for indicating that values of state variables $c_1, c_2, \ldots, c_n$ of s' is $v_1, v_2, \ldots, v_n$ respectively, while values of other state variables in s' remain unchanged.

3.1.3 Invariants

Invariants are propositions for states of a model. For each event which incurs transitions of states, the invariants should still be preserved. The property can be defined as follows:

$$\forall s\ s'\ p_e.\mathrm{I}(s) \wedge \mathbf{Grds}_e(p_e, s) \wedge \mathbf{Act}_e(p_e, s, s') \rightarrow \mathrm{I}(s'). \tag{2}$$

Here, $\mathrm{I}(s)$ is the conjunction of all invariants of a model at a state s. The property states that for an event e whose parameters are p_e and that leads a state s to a state s', if the invariants are valid in the state s, the invariants are valid in the state s'.

3.2 Refinement Method

The refinement is a process of constructing a more concrete model from the previous abstract model. The concrete model should preserve invariants of the abstract model so that we can model a complex object, *e.g.*, a shielding system, by using several models and each of the model is constructed by focusing on individual design aspects. In practice, the concrete model preserves a refined form of the invariants. The refined form and the conjunction of invariants are mutually implicated, and therefore we should prove the following theorem.

$$\forall s_a\ s_c.\mathrm{I}_a(s_a) \wedge \mathrm{SR}^{ac}(s_a, s_c) \leftrightarrow \mathrm{I}_{r,c}(s_c) \wedge \mathrm{SR}^{ac}(s_a, s_c). \tag{3}$$

I_a is the conjunction of all invariants of the abstract model and $\mathrm{I}_{r,c}$ is the refined form of I_a. SR^{ac} is the conjunction of all propositions that specify relations between values of state variables of states of the concrete model and the abstract model. The theorem specifies $\mathrm{I}_{r,c}$ is the refined form of I_a. Now, we prove that $\mathrm{I}_{r,c}$ is preserved in the concrete model by using the preservation of I_a in the abstract model for reducing workload in auditing each event in the concrete model for $\mathrm{I}_{r,c}$. Specifically, for each event e_c in the concrete model, we first manually identify a corresponding abstract event e_a in the abstract model. We assume e_c leads a state s_c to a state s'_c, and e_a leads a state s_a to a state s'_a. Then, we first prove the following theorem.

$$\forall s_c\ p_{e_c}.\mathrm{I}_c(s_c) \wedge \mathbf{Grds}_{e_c}(p_{e_c}, s_c) \rightarrow\ \exists p_{e_a}\ \exists s_a.\mathrm{SR}^{ac}(s_a, s_c) \wedge \mathbf{Grds}_{e_a}(p_{e_a}, s_a). \tag{4}$$

The theorem that states that a concrete event can only occur when the corresponding abstract event occurs. Then, we prove the following theorem.

$$\begin{aligned} &\forall s_c\ s'_c\ p_{e_c}.\mathrm{I}_c(s_c) \wedge \mathbf{Act}_{e_c}(p_{e_c}, s_c, s'_c) \rightarrow \\ &\exists p_{e_a}\ \exists s_a\ \exists s'_a.\mathrm{SR}^{ac}(s_a, s_c) \\ &\wedge \mathrm{SR}^{ac}(s'_a, s'_c) \wedge \mathbf{Act}_{e_a}(p_{e_a}, s_a, s'_a). \end{aligned} \tag{5}$$

The theorem that states that if a concrete event occurs, the abstract event can occur in such a way that the resulting states correspond again. With the two theorems, we can ensure the preservation of $\mathrm{I}_{r,c}$ on each concrete event by using the preservation of I_a on the corresponding abstract event.

4 A General Model of Shielding Systems

In this section, we construct a general model of shielding systems by stepwise refinement. The model contains general operations used for allocating and deallocating storage of the OS and SCAs and for managing data of the OS and SCAs.

4.1 Requirements and Assumptions

We start by specifying general requirements of shielding systems and by making our assumptions about the system explicit.

Requirement 1 (Shielding System). A shielding system, *e.g.*, a hypervisor, executes in a higher privilege mode against **guests**, *i.e.*, OS or SCAs.

Requirement 2 (Memory Isolation). Memory that can be accessed by each guest is pairwise disjoint.

Requirement 3 (Data Confidentiality). Critical data, *e.g.*, private keys, of an SCA cannot be leaked to other guests.

The compromised guests may operate in following ways.

Assumption 1 (Dolev-Yao Adversary). Similar to the Dolev-Yao adversary model [12], the guests may perform data analyzing, (*e.g.*, decomposing data, decrypting data using obtained keys), and data synthesizing according to the analyzed result. However, the adversary cannot perform any crypto-analysis.

Assumption 2 (Calls). A guest may call any interface of the shielding system exposed to the guest when the guest is executing. Moreover, the guest may call the interfaces with arbitrary parameter values.

4.2 Memory Isolation (S0)

The abstract machine (*S0*) models the property of memory isolation.

States S^0: Denote S^i as the record for states at the ith refinement. We define S^0 in *S0* as follows:

$$S^0 \stackrel{def}{=} \{loc : tguests \mapsto tloc\ set\}. \tag{6}$$

Here, $loc(id)$ represents memory locations that can be accessed by a guest whose identifier is id. Generally, we define $t*$ as the type of some variables. For example, *tloc*, *tguests* is the type of memory locations and identifiers of guests respectively. $tguests \stackrel{def}{=} Tisca(id : tident_{sca})|OSID$. Therefore, a guest may be an SCA with the identifier id or an OS with the identifier $OSID$.

Invariant: We model Memory Isolation (*i.e.*, Requirement 2) by the following invariant.

$$\begin{aligned}&\text{Isolation}^0_{mem}(s) \stackrel{def}{=}\\ &(\forall\ i\ j.i \neq j \rightarrow (s.loc(i) \cap s.loc(j)) = \phi).\end{aligned} \tag{7}$$

Events: According to the invariant Isolation^0_{mem}, we model an event ChLoc^0 in which the memory that can be accessed by a guest id is changed to l. Specifically,

$$\begin{aligned}&\mathbf{Grd}_{\mathsf{ChLoc}^0}\ l\ i\ s \stackrel{def}{=}\\ &\quad \textbf{(i)}\ (\forall j, j \neq i \rightarrow l \cap s.loc(j) = \phi)\\ &\mathbf{Act}_{\mathsf{ChLoc}^0}\ l\ i\ s\ s' \stackrel{def}{=}\\ &\quad newl = (s.loc(i) := l) \wedge s \sim_{loc=newl} s'.\end{aligned} \tag{8}$$

In the guard, we require that l should not intersect with memory that can currently be accessed by other guests, including the OS and other SCAs. In the action, the state variable *loc* is changed into l, where l is built by leveraging *loc* in original state s, *i.e.*, $s.loc$. Hence, we can prove that the invariant $\text{Isolation}^0_{mem}(s)$ is preserved after the occurrence of the event.

4.3 Memory Isolation with Changing Modes (S1)

In the first refinement (*S1*), we additionally model events when guests are changing their internal modes. Specifically, the shielding system determines whether a guest can execute or not, which correspond to executing mode and suspending mode. Therefore, for any guest i, $loc(i)$ turns into ϕ when guest i is switched into suspending mode. Moreover, the shielding system may allocate new memory of a guest when a guest is in suspending mode, and the value of $loc(i)$ becomes more complicated when the guest is switched back to executing mode. The above situations motivate us to model new events and prove memory isolation in *S1*.

States S^1: The record for states in *S1* is shown in the following.

$$\begin{aligned}S^1 \stackrel{def}{=}\ &\{mode : tguests \mapsto tmode,\\ &mem : tguests \mapsto tloc\ set\}.\end{aligned} \tag{9}$$

The internal mode of a guest i is denoted as $mode(i)$. $mode(i) = EXE$ or SUS, when i is executing or suspending respectively. Denote $mem(i)$ as locations of memory allocated for guest i by the shielding system. Then, we define the invariant for modeling the relation between S^0 and S^1:

$$\begin{aligned}&\text{SR}^{01}(s0, s1) \stackrel{def}{=} (\forall i.s1.mode(i) = EXE \rightarrow\\ &s0.loc(i) = s1.mem(i)) \wedge\\ &(\forall i.s1.mode(i) = SUS \rightarrow s0.loc(i) = \phi).\end{aligned} \tag{10}$$

The invariant means that for any guest i, if it is in executing mode, then the memory that can be accessed by i, *i.e.*, $s0.loc(i)$, is $s1.mem_{aloc}(i)$. Otherwise, if i is in suspending mode, no memory can be accessed by i.

Invariant: Based on the refined state S^1, we refine $\text{Isolation}^0_{mem}(s)$ by the following invariant.

$$\begin{aligned}&\text{Isolation}^1_{mem}(s) \stackrel{def}{=} \forall i\ j.i \neq j \rightarrow\\ &s.mode(i) = EXE \rightarrow s.mode(j) = EXE \rightarrow\\ &s.mem(i) \cap s.mem(j) = \phi\end{aligned} \tag{11}$$

The invariant states that there is no shared memory between memory that can be accessed by two guests that are executing simultaneously. We notice the invariant because we use refinements of *S1* to model shielding systems [11, 18] which support multi-core CPU. In this case, several SCAs may run on different cores simultaneously.

Events: There are 2 types of events in *S1*:

1) The event when a guest is in suspending mode. Without loss of generality, the action in this case is modeled as arbitrary changes of *mem*. We further refine the action according to different implementations in the case studies.

2) Transition of the modes. The events used for executing a guest or suspending a guest, and the events are refined from ChLoc^0_{guest}. For example, we model an event in the following:

$$\begin{aligned}&\mathbf{Grd}_{\mathsf{StoE}^1}\ i\ s \stackrel{def}{=}\\ &\quad\textbf{(i)}\ mode(i) = SUS \wedge\\ &\quad\textbf{(ii)}\ (\forall j.j \neq i \wedge mode(j) = EXE \rightarrow\\ &\quad s.mem(i) \cap s.mem(j) = \phi)\\ &\mathbf{Act}_{\mathsf{StoE}^1}\ \ i\ s\ s', \stackrel{def}{=}\\ &\quad newm = (s.mode(i) := EXE) \wedge s \sim_{mode=newm} s'.\end{aligned} \tag{12}$$

In the event, the mode of the guest i is transited from *SUS* to *EXE*. Specially, the guard requires that $mem(i)$ does not intersect with memory that can be accessed by other guests. The guard helps proving the theorem of refinement (4). Note that we do not model the event when a guest is executing, since *mem* remains unchanged according to current systems.

4.4 Data Confidentiality (*S2*)

The goal of *S2* is to prove data confidentiality as illustrated in Requirement 3. We leverage property of memory isolation in *S1*, and model more state variables that may affect the goal. For example, a potential attack occurs when a guest is switched into suspending mode. In this case, the shielding system may save the guest's data on the memory and registers, and then allocates the memory to other guests. If the memory or registers are not cleared before they are allocated, the data may be leaked since other guests are assumed to be adversarial. Hence, we formulate the data stored on the memory and refine the events in *S1*.

$tdata \stackrel{def}{=}$	*Hash tdata*	Hashes of data
|	*Key tkey*	Keys
|	*Enc tkey tdata*	Ciphertexts
|	*Id tguest*	Identifiers of guests
|	*Cons tdata tdata*	Concatenation of data
|	*Others*	Other types of data

Figure 1: The sub-types of data defined in *S2*

For preventing more complicated attacks, we model the adversary in Dolev-Yao style. The adversary may perform analysis on obtained data, *e.g.*, decrypting a cypher using obtained encryption key, and then forge data. In Figure 1, we firstly define the type *tdata* for formulating data, and divide *tdata* into several sub-types. Then, we model the ability of adversary by defining functions *analz* and *synth* and the corresponding axioms [6]. Both functions share the same declarations:

$$analz, synth : tdata\ set \mapsto tdata\ set$$

The function *analz* outputs the set of data that can be analyzed from the input. For example, if $Key(invKey(k)) \in s$ and $Enc(k,m) \in s$, then $m \in analz(s)$. *invKey* is a function that leaves a key unaltered if the key is symmetric, or turns a key into its corresponding asymmetric half if the key is asymmetric. The function *synth* outputs the set of data that can be composed by using the input. For example, if $m, k \in s$, then $Enc(k, m) \in synth(s)$. The properties in both examples can be proved by using the defined axioms. For simplicity, we use the notation δ that $\delta(d) = synth(analz(d))$.

States S^2 : The record for states in *S2* is shown in the following.

$$\begin{aligned}S^2 \stackrel{def}{=}\ & S^1 + data_{mem} : tloc \mapsto tdata\ set\\ & + pdata : tguests \mapsto tdata\ set\\ & + know : tguests \mapsto tdata\ set\\ & + gset : tdata\ set\\ & - mode\\ & + core : tguests \mapsto tcores\ set\\ & + data_{regs} : tcores \mapsto tdata\ set\end{aligned} \tag{13}$$

Here, $data_{mem}(i)$ represents the set of data that may currently be stored on memory location *i*. $pdata(i)$ represents the set of private data owned by guest *i*. $know(i)$ represents guest *i*'s **knowledge**, *i.e.*, the set of data that may have been obtained by guest *i*. In our model, the data $d \in know(i)$, if d was in the memory location or register that could be accessed by the guest *i*. Note that the data d may be divide into multiple blocks, *e.g.*, $\{b_1, b_2, \ldots, b_m\}$, which are stored into multiple locations of memory, *e.g.*, $\{d_1, d_2, ..., d_m\}$, respectively. Obviously, an adversary who only reads a single block of d, *e.g.*, b_i in $\{b_1, b_2, \ldots, b_m\}$, does not obtain d. However, the private information, *e.g.*, the encryption key, may happen to be in b_1, which lead to successful attacks. Hence, we simply assume that the adversary may perform attacks by leveraging d, if the adversary obtains any b_i from the location d_i.

The state variable *gset* represents the set of global data that have been generated by the shielding system and guests, and is used to achieve the assumption of Dolev-Yao adversary. Since the adversary cannot perform crypto-analysis, the possibility that newly generated random data equal any historical randomly generated data is assumed to be 0. Therefore, in our model, the data d should satisfy the guard that $(\forall\ k.\ k \in parts(d) \rightarrow k \notin s.gset)$, before d is newly generated in the events, in which $parts(d)$ is added into *gset*. *parts* is a function that outputs the set of data that can be extracted from the input. For example, if $Enc(k,m) \in s$, then $m \in parts(s)$. Axioms for *parts* can be found in [6].

For generality, we also model and prove confidentiality of data stored in the registers, besides the

memory. Recall that the cores of an CPU may be used by multiple guests simultaneously. It should be proved that the data stored on the registers, which belong to the cores used by a guest, are not leaked to other guests. In S^2, we replace *mode* with *core*, where $core(i)$ represents the set of cores currently used by the guest i. Similar to $data_{mem}$, $data_{regs}(i)$ represents the set of data that may currently be stored at the registers of the core i. Therefore, *core* is a refined state of *mode* that a guest is using a core or several cores if it is executing; otherwise, it does not use any cores. Formally, we define the relation as invariant SR^{12} as follows.

$$\begin{aligned}&\mathrm{SR}^{12}(s1, s2) \stackrel{def}{=}\\ &\quad(\forall i.\ s1.mode(i) = EXE \leftrightarrow s2.core(i) \neq \phi)\wedge\\ &\quad(\forall i.\ s1.mode(i) = SUS \leftrightarrow s2.core(i) = \phi).\end{aligned} \tag{14}$$

Invariant: We prove data confidentiality Conf as follows.

$$\begin{aligned}&oknow(s, i) = \{x | j \in GIDS, j \neq i, x \in s.know(j)\}.\\ &\mathrm{Conf}(s) \stackrel{def}{=} \forall i. i \in GIDS \wedge i \neq OSID \rightarrow\\ &\delta(oknow(s, i)) \cap s.pdata(i) = \phi.\end{aligned} \tag{15}$$

The invariant states that the private data owned by any SCA, *i.e.*, $pdata(i)$, cannot be analyzed or synthesized according to the knowledge of other guests. $oknow(s,i)$ represents union of knowledge of guests other than guest i. Here, we assume that the other guests may collude by sharing knowledge with others and perform attacks on guest i. *GIDS* is a constant that represents the set of identifiers of all guests.

Events: We divide events in *S2* into 5 parts.

1) Generating private data. In the events, a guest i generates private data d, *e.g.*, private keys, and saves d into memory or registers. For example, when d is saved into memory location ld, the event is modeled as follows:

$$\begin{aligned}&\mathbf{Grd}_{\mathsf{Gen}^2_{mem}}\ i\ d\ ld\ s \stackrel{def}{=}\\ &\quad\mathbf{(i)}\ i \in GIDS \wedge \mathbf{(ii)}\ s.core(i) \neq \phi\wedge\\ &\quad\mathbf{(iii)}\ ld \subseteq s.mem(i) \wedge \mathbf{(iv)}\ (\forall k.k \in parts(d) \rightarrow k \notin s.gset)\\ &\mathbf{Act}_{\mathsf{Gen}^2_{mem}}\ i\ d\ ld\ s\ s', \stackrel{def}{=}\\ &\quad newd = update_{mem}(s.data_{mem}, ld, d)\wedge\\ &\quad newp = (s.pdata(i) := s.pdata(i) \cup d)\wedge\\ &\quad newk = (s.know(i) := s.know(i) \cup d)\wedge\\ &\quad newe = (s.gset \cup parts(d))\wedge\\ &\quad s \sim_{data_{mem}, pdata, know, gset = newd, newp, newk, newe} s'.\end{aligned} \tag{16}$$

Guard **(ii)**, **(iii)** states that guest i is executing and can access the location ld, respectively. Guard **(iv)** states that the data d is newly generated. In the action, the memory, private data and the knowledge owned by guest i are changed. Here, $(update_{mem}(f, l, d))(x) = \{d, \mathsf{if}\ x \in l; f(x), \mathsf{if}\ x \notin l\}$.

2) Malicious operations. The guests may perform operations according to the Dolev-Yao adversary model. Specifically, a guest may write data to the memory or registers that can be accessed by the guest. The following event represents the case when guest writes data to the memory.

$$\begin{aligned}&\mathbf{Grd}_{\mathsf{Mal}^2_{mem}}\ i\ d\ ld\ s \stackrel{def}{=}\\ &\quad\mathbf{(i)}\ i \in GIDS \wedge \mathbf{(ii)}\ s.core(i) \neq \phi\wedge\\ &\quad\mathbf{(iii)}\ ld \subseteq s.mem(i) \wedge \mathbf{(iv)}\ d \in \delta(s.know(i))\\ &\mathbf{Act}_{\mathsf{Mal}^2_{mem}}\ i\ d\ ld\ s\ s', \stackrel{def}{=}\\ &\quad newd = update_{mem}(s.data_{mem}, ld, \{d\})\wedge\\ &\quad newk = (s.know(i) := s.know(i) \cup \{d\})\wedge\\ &\quad s \sim_{data_{mem}, know = newd, newk} s'.\end{aligned} \tag{17}$$

Guard **(ii)** states that only the executing guest can perform the operations. In the action, data d is written to location ld, where ld is restricted by guard **(iii)**, and d is added to the knowledge of the guest i. Data d is restricted by the guard **(iv)** that the data must be analyzed or synthesized from the knowledge of the guest.

3) Transition of modes. There are two events in this case: (1) StoE^2: the mode of a guest is switched from suspending mode to executing mode, (2) EtoS^2: the mode of a guest is switched to suspending mode. When the mode of guest is switched from executing mode to suspending mode, the guest does not use any core, and vice versa. Therefore, the events are directly refined from events in *S1*. For example, when the guest i starts using with core idc, we model the event as follows.

$$\begin{aligned}&\mathbf{Grd}_{\mathsf{StoE}^2}\ i\ idc\ s \stackrel{def}{=}\\ &\quad\mathbf{(i)}\ i \in GIDS \wedge \mathbf{(ii)}\ s.core(i) = \phi\wedge\\ &\quad\mathbf{(iii)}\ (\forall j.\ j \in GIDS \wedge j \neq i \wedge s.core(j) \neq \phi \rightarrow\\ &\quad s.mem(i) \cap s.mem(j) = \phi)\wedge\\ &\quad\mathbf{(iv)}\ (\forall j.\ j \in GIDS \rightarrow idc \notin s.core(j))\wedge\\ &\quad\mathbf{(v)}\ (\forall j.\ j \in GIDS \wedge j \neq i \wedge j \neq OSID \rightarrow\\ &\quad\delta(oknow(s, j) \cup s.data_{regs}(idc)\\ &\quad\cup\ clt(s.data_{mem}, s.mem(i))) \cap s.pdata(j) = \phi)\\ &\mathbf{Act}_{\mathsf{StoE}^2}\ \ i\ idc\ s\ s', \stackrel{def}{=}\\ &\quad newc = (s.core(i) := s.core(i) \cup \{idc\})\wedge\\ &\quad newk = (s.know(i) := s.know(i) \cup s.data_{regs}(idc)\\ &\quad\cup\ clt(s.data_{mem}, s.mem(i)))\wedge\\ &\quad s \sim_{core, know = newc, newk} s'.\end{aligned} \tag{18}$$

The guard **(ii)** and **(iii)** refines the guard **(i)** and **(ii)** in StoE^1, respectively. The guard **(iv)** states that the core idc has not been used by any guest. The guard

(v) states that for any guest except guest i and the OS, private data of the guest cannot be analyzed or synthesized from the union of the knowledge of other guests and the set of data in memory and registers that can be accessed by guest i. In the action, the state variable *core* is changed and the set of data in memory and registers that can be accessed by guest i is added to the knowledge of guest i. The function *clt* is used for collecting data in memory. Formally, $clt(f, m) = \{d | x \in m, d \in f(x)\}$.

4) Changing the number of used cores. Specifically, a guest may start using more or less cores when the guest is executing. To preserve confidentiality in this event, we add the guard which is similar to the guard **(v)** in StoE^2.

5) Other operations performed by the shielding system. The shielding system may perform other operations for managing the memory or registers. For example, the shielding system may write data to the memory or registers, clear the data in the memory or registers, reallocate the memory for a guest, or generate data, *e.g.*, keys used for encrypting data of guests. We show the event that writes data to memory in the following.

$$
\begin{aligned}
&\mathbf{Grd}_{\mathsf{SWrite}^2_{mem}}\ d\ ld\ s \stackrel{def}{=}\\
&\quad \textbf{(i)}\ \forall\, j_1.j_1 \in GIDS \wedge\\
&\quad s.core(j_1) \neq \phi \wedge ld \cap s.mem(j_1) \neq \phi \rightarrow\\
&\quad (\forall j_2.\ j_2 \in GIDS \wedge j_2 \neq j_1 \wedge j_2 \neq OSID \rightarrow\\
&\quad \delta(oknow(s, j_2) \cup d) \cap s.pdata(j_2) = \phi)\\
&\mathbf{Act}_{\mathsf{SWrite}^2_{mem}}\ d\ ld\ s\ s', \stackrel{def}{=}\\
&\quad newd = update_{mem}(s.data_{mem}, ld, d) \wedge\\
&\quad newk = update_{knwl}(s.know, s.core, s.mem, ld, d) \wedge\\
&\quad s \sim_{data_{mem}, know=newd, newk} s'.
\end{aligned}
\tag{19}
$$

The guard **(i)** states that if the intersection between ld and memory that can be accessed by the guest j_1 is not empty, for any guest except guest j_1 and the OS, private data of the guest cannot be analyzed or synthesized from the union of the set of data to be written, *i.e.*, d, and the knowledge of other guests. In the action, the shielding system writes d to ld and the knowledge of guests that can access a part of memory locations in ld is added with d. Formally, $(update_{knwl}(k,c,m,ld,d))(i)=\{k(i),$ if $(m(i) \cap ld = \phi) \vee (c(i)=\phi);\ k(i) \cup d,$ if $(m(i) \cap ld \neq \phi) \wedge (c(i) \neq \phi)\}$. We omit the definition of other operations in this paper.

5 Case Studies

In this section, we model two shielding systems, *i.e.*, TrustVisor [21] and OSP [11], based upon the refinements of our general model.

5.1 Case 1: TrustVisor

TrustVisor [21] is an open-source hypervisor used for shielding SCAs, which are called Pieces of Application Logic (PALs) by TrustVisor. TrustVisor provides code integrity as well as data integrity and confidentiality for SCAs. Besides, TrustVisor leverages the features of modern processors to reduce the performance overhead caused by protecting SCAs from the OS and its applications. We now explain TrustVisor's functions related to our goals as follows.

1) *registration*: The *registration* interface allows the OS to register SCAs. When the OS calls *registration*, TrustVisor prepares an environment for launching a new SCA, *i.e.*, the memory to be accessed by the SCA. The memory cannot be accessed by the OS and the data on the memory should be prepared. In practice, before using *registration*, the OS sets a region of the memory and the data on the memory, and pass the region as the parameters in *registration*. After *registration* is called, TrustVisor checks the parameters, and sets the corresponding memory unaccessible by the OS. Besides, TrustVisor prepares the context of the SCA, *i.e.*, the data to be loaded to the registers, according to the parameters in *registration*.

2) *invocation*: Following *registration*, the OS may invoke an SCA by calling *invocation*. When the OS calls *invocation*, the OS is switched into suspending mode, and the selected SCA is started to execute. TrustVisor firstly saves the context of the OS, and then loads the context of the SCA. Besides, the OS produces input and passes the input to the SCA. Specifically, the OS firstly puts the input in memory region specified in the parameters of *invocation*. Then, after calling *invocation*, TrustVisor copies the data in the memory region to the memory that can be accessed by the SCA. Finally, the SCA is allowed to use the core of the CPU. Note that the design of TrustVisor only supports using a single CPU core, and TrustVisor can manage multiple SCAs.

3) *termination*: When an SCA has completed executing and returns to the OS, *termination* is called through a return point set by TrustVisor in the stack used by the SCA. When an SCA calls *termination*, the OS and the executing SCA is switched into executing mode and suspending mode respectively. During the process, TrustVisor saves the context of the executing SCA, and copies the context of the OS. Besides, the SCA produces output and passes the output to the OS. Specifically, the SCA firstly puts the output in a memory region specified by TrustVisor. Then, after calling *termination*, TrustVisor copies the data in the memory region to the memory region specified in the parameters of *invocation*.

4) *unregistration*: TrustVisor zeros all execution state associated with the SCA specified by parameters of

unregistration. When the OS unregisters an SCA by calling *unregistration*, TrustVisor deallocates all memory allocated for the SCA, clears data in the deallocated memory, and allocates the memory to the OS.

5) hv_{seal}: When an SCA calls hv_{seal}, TrustVisor encrypts data in the specified blocks of memory using the symmetric key, *e.g.*, k, owned by the TrustVisor, writes the encrypted data to the memory allocated for the SCA, and continues executing the SCA. Here, TrustVisor binds the identifier of the SCA i to the encrypted data d. In other words, the ciphertext outputted by hv_{seal} are formed as $Enc(k, Cons(d, i))$.

6) hv_{unseal}: Corresponding to hv_{seal}, when an SCA calls hv_{unseal} for decrypting $Enc(k, Cons(d, i))$, TrustVisor ensures that d is originally sealed by i, *i.e.*, i is the identifier of the SCA, before writing the decrypted data to the memory that can be accessed by the SCA. Then, TrustVisor continues executing the SCA.

States (S^3_{tv}): We show the record for states in S^3_{tv} in the following.

$$\begin{aligned} S^3_{tv} \stackrel{def}{=} \quad & S^2 + func : option\ tfun \\ & + ctxt : tguest \mapsto (tdata\ set) \\ & + symkey : option\ tkey \end{aligned} \tag{20}$$

The state variable *func* denotes which function provided by TrustVisor, *e.g.*, *registration*, is currently being executed. If $func = NONE$, then no function is running. Note that TrustVisor only uses a single core of the CPU, therefore there is at most one executing function at each state. We divide the process of calling functions into several steps, and each step can be refined from our general model. Therefore, the state variable *func* records not only the executing function's name, but also the function's current step and parameters. For example, $Ivc(s : step_{ivc})(p : prmt_{ivc})$ is a subtype of *tfun*. If $func = Some(Ivc(IVC_{sv}, prm_{ivc}(i)))$, it means that the function *invocation* is being executed, the current step in *invocation* is saving the context of the OS, and the selected SCA is guest i. Here, $step_{ivc}$ is the type of steps of *invocation*, and each of the step is defined as a constant. $prmt_{ivc}$ is the type of parameters of *invocation*, and formally, $prmt_{ivc} \stackrel{def}{=} prm_{ivc}(t : tguest)$. In S^3_{tv}, we also add *ctxt* and *symkey* representing contexts and symmetric key held by TrustVisor, respectively.

Events: As mentioned, the process of calling each function is divided into several steps, and each step is modeled as an event refined from *S2* or a new event in which the state variables in former refinement level do not change. We show the example of modeling functions *invocation* and hv_{seal} as follows.

1) *invocation*: The process is divided into 5 steps. In the first step, the OS is switched into suspending mode. Hence, the event is refined from EtoS^2.

$$\begin{aligned} & \mathbf{Grd}_{\mathsf{EtoS}^3_{ivc}}\ i\ s \stackrel{def}{=} \\ & \quad \mathbf{(i)}\ (s.func = NONE) \wedge \mathbf{(ii)}\ (s.core(OSID) \neq \phi) \wedge \\ & \quad \mathbf{(iii)}\ (i \in GIDS \wedge i \neq OSID) \\ & \mathbf{Act}_{\mathsf{EtoS}^3_{ivc}}\ i\ s\ s' \stackrel{def}{=} \\ & \quad newc = (s.core(OSID) := \phi) \wedge \\ & \quad s \sim_{func, core = Some(Ivc(IVC_{sv}, prm_{ivc}(i))), newc}\ s'. \end{aligned} \tag{21}$$

The guard **(i)** states that there is no function that is executing. The guard **(ii)** states that the OS is occupying the CPU, which means the OS is executing. In guard **(iii)**, guest i is selected to execute. Therefore in the action, the OS no longer uses the CPU so that the mode of the OS is switched into suspending mode. On the other hand, the step turns into IVC_{sv}.

In the second step IVC_{sv}, TrustVisor saves the context of the OS. Since context is saved to the new state *ctxt* in S^3_{tv} and no other state is changed, we model the step as a new event.

$$\begin{aligned} & \mathbf{Grd}_{\mathsf{Sv}^3_{ivc}}\ i\ s \stackrel{def}{=} \\ & \quad \mathbf{(i)}\ s.func = Some(Ivc(IVC_{sv}, prm_{ivc}(i))) \\ & \mathbf{Act}_{\mathsf{Sv}^3_{ivc}}\ i\ s\ s' \stackrel{def}{=} \\ & \quad newc = (s.ctxt(OSID) := s.data_{regs}(UCORE)) \wedge \\ & \quad s \sim_{func, ctxt = Some(Ivc(IVC_{copy}, prm_{ivc}(i))), newc}\ s'. \end{aligned} \tag{22}$$

The guard ensures that the event occurs only when the previous step has been executed, in which the *func* is changed into $Some(Ivc(IVC_{sv}, prm_{ivc}(i)))$. In the action, the data on the registers are assigned to $ctxt(OSID)$, and the step turns into IVC_{copy}. Here, *UCORE* is a constant representing the identifier of the single core used by TrustVisor.

In the third step IVC_{copy}, TrustVisor manages the memory for preparing the input of *invocation* before SCA i is executed. Specifically, the data d are generated by the OS as input, which are stored in *ls*, and then copied to the location *ld*, which belongs to SCA i. The step is refined from the event SWrite^2_{mem} in *S2*.

$$\begin{aligned} & \mathbf{Grd}_{\mathsf{Copy}^3_{ivc}}\ i\ ls\ d\ ld\ s \stackrel{def}{=} \\ & \quad \mathbf{(i)}\ s.func = Some(Ivc(IVC_{copy}, prm_{ivc}(i))) \wedge \\ & \quad \mathbf{(ii)}\ ld \subseteq s.mem(i) \wedge \mathbf{(iii)}\ ls \subseteq s.mem(OSID) \wedge \\ & \quad \mathbf{(iv)}\ d = clt(s.data_{mem}, ls) \\ & \mathbf{Act}_{\mathsf{Copy}^3_{ivc}}\ i\ ls\ d\ ld\ s\ s' \stackrel{def}{=} \\ & \quad newd = update_{mem}(s.data_{mem}, ld, d) \wedge \\ & \quad s \sim_{func, data_{mem} = Some(Ivc(IVC_{load}, prm_{ivc}(i))), newd}\ s'. \end{aligned} \tag{23}$$

Note that the guard **(i)** implies there is no guest that is executing. Therefore when Copy^3_{ivc} occurs, the guard **(i)** in SWrite_{mem} is ensured, and the knowledge of any guest is not changed for SWrite^2_{mem}, *i.e.*, $update_{knwl}(s.know,\ s.core,\ s.mem,\ ld,\ d) = s.know$. The step then turns into IVC_{load}.

In the fourth step, TrustVisor loads the context of the SCA i. It is also refined from the event in *S2*, in which TrustVisor writes data to registers.

$$\begin{aligned}&\mathbf{Grd}_{\mathsf{Load}^3_{ivc}}\ i\ s \stackrel{def}{=}\\ &\quad \mathbf{(i)}\ s.func = Some(Ivc(IVC_{load}, prm_{ivc}(i)))\\ &\mathbf{Act}_{\mathsf{Load}^3_{ivc}}\ i\ s\ s' \stackrel{def}{=}\\ &\quad newd = (s.data_{regs}(UCORE) := s.ctxt(i)) \wedge\\ &\quad s \sim_{func, data_{regs}=Some(Ivc(IVC_{StoE}, prm_{ivc}(i))), newd}\ s'.\end{aligned} \tag{24}$$

The guard simply checks if it is at the fourth step, *i.e.*, IVC_{load}. In the action, the context, which is stored by TrustVisor and modeled as a global state *ctxt*, is assigned to the registers, *i.e.*, $data_{regs}$. Then the step turns into IVC_{StoE}.

In the final step, the mode of SCA i is switched into executing mode, therefore the event is refined from StoE^2.

$$\begin{aligned}&\mathbf{Grd}_{\mathsf{StoE}^3_{ivc}}\ i\ s \stackrel{def}{=}\\ &\quad \mathbf{(i)}\ s.func = Some(Ivc(IVC_{StoE}, prm_{ivc}(i)))\\ &\mathbf{Act}_{\mathsf{StoE}^3_{ivc}}\ i\ s\ s' \stackrel{def}{=}\\ &\quad newc = (s.core(i) := s.core(i) \cup \{UCORE\}) \wedge\\ &\quad newk = (s.know(i) := s.know(i) \cup s.data_{regs}(UCORE)\\ &\quad \cup\ clt(s.data_{mem}, s.mem(i))) \wedge\\ &\quad s \sim_{func, core, know=NONE, newc, newk}\ s'.\end{aligned} \tag{25}$$

Here, the guard is refined that it only checks whether the step is IVC_{StoE}. The guards defined in StoE^2 are true according to the definitions of previous steps. Finally, the state variable *func* turns into *NONE*.

2) hv_{seal}: We divide the process into 3 steps. In the first step, the mode of SCA i is switched from executing mode to suspending mode. Therefore the step is also refined from EtoS^2.

$$\begin{aligned}&\mathbf{Grd}_{\mathsf{EtoS}^3_{seal}}\ i\ d\ s \stackrel{def}{=}\\ &\quad \mathbf{(i)}\ (s.func = NONE) \wedge \mathbf{(ii)}\ (s.core(i) \neq \phi) \wedge\\ &\quad \mathbf{(iii)}\ (i \in GIDS \wedge i \neq OSID)\\ &\mathbf{Act}_{\mathsf{EtoS}^3_{seal}}\ i\ d\ s\ s' \stackrel{def}{=} newc = (s.core(i) := \phi) \wedge\\ &\quad s \sim_{func, core=Some(Seal(SEAL_{enc}, prm_{seal}(i))), newc}\ s'.\end{aligned} \tag{26}$$

The guards state that SCA i is executing. Besides the actions refined from EtoS^2, we add action that the step turns into $SEAL_{enc}$.

In the second step, TrustVisor encrypts data in memory that can be accessed by SCA i, and save the encrypted data. The step is refined from the event SWrite^2_{mem} in *S2*.

$$\begin{aligned}&\mathbf{Grd}_{\mathsf{Seal}^3_{enc}}\ i\ ls\ d\ ld\ k\ s \stackrel{def}{=}\\ &\quad \mathbf{(i)}\ s.func = Some(Seal(SEAL_{enc}, prm_{seal}(i))) \wedge\\ &\quad \mathbf{(ii)}\ ld \subseteq s.mem(i) \wedge \mathbf{(iii)}\ s.symkey = Some(k) \wedge\\ &\quad \mathbf{(iv)}\ ls \subseteq s.mem(i) \wedge \mathbf{(v)}\ d \in clt(s.data_{mem}, ls)\\ &\mathbf{Act}_{\mathsf{Seal}^3_{enc}}\ i\ ls\ d\ ld\ k\ s\ s' \stackrel{def}{=}\\ &\quad newd = update_{mem}(s.data_{mem}, ld,\\ &\quad \{Enc(k, Cons(d, Id(i)))\}) \wedge\\ &\quad s \sim_{func, data_{mem}=Some(Seal(SEAL_{StoE}, prm_{seal}(i))), newd}\ s'.\end{aligned} \tag{27}$$

The guard **(i)** checks whether the step is $SEAL_{enc}$. In the guards, we also make constraints on the parameters of the event. Specifically, the data d in memory ls are encrypted, and the encrypted data are saved to ld, which is located at memory that can be accessed by SCA i. In guard **(iii)**, TrustVisor must have generated the encryption key before the event occurs.

In the final step, the mode of an SCA is switched back to executing mode. Hence, the step is refined from the event StoE^2. Since the refined event is similar to StoE^3_{ivc}, we simply provide the definition as follows and omit the explanations.

$$\begin{aligned}&\mathbf{Grd}_{\mathsf{StoE}^3_{seal}}\ i\ s \stackrel{def}{=}\\ &\quad \mathbf{(i)}\ s.func = Some(Seal(SEAL_{StoE}, prm_{seal}(i)))\\ &\mathbf{Act}_{\mathsf{StoE}^3_{seal}}\ i\ s\ s' \stackrel{def}{=}\\ &\quad newc = (s.core(i) := s.core(i) \cup \{UCORE\}) \wedge\\ &\quad newk = (s.know(i) := s.know(i) \cup s.data_{regs}(UCORE)\\ &\quad \cup\ clt(s.data_{mem}, s.mem(i))) \wedge\\ &\quad s \sim_{func, core, know=NONE, newc, newk}\ s'.\end{aligned} \tag{28}$$

Results: Since we have proved properties of memory isolation and data confidentiality in previous refinement levels, to guarantee the properties as well, we only need to individually prove that the guards and actions correctly refined in each event, *i.e.*, the theorem Eq. (4),(5) are satisfied in *S3*. Therefore, it is unnecessary to prove that each event satisfies the properties. Moreover, for many events in different functions are refined from the same functions, *e.g.*, StoE^2, our method also reduces the complexity of modeling the shielding systems.

We notice a potential security threat in using the function *terminate*. When TrustVisor copies the output prepared by an SCA to memory allocated for the OS, private data of the SCA may be in the output, if the function is not carefully used by the designers of SCAs. Therefore in the next steps, data confidentiality is violated because the private data in the output is added to the knowledge of

the OS. Formally, in the step of copying the output of an SCA, TrustVisor copies the set of data stored in *ls* to *ld*.

$$\begin{aligned}&\mathbf{Grd}_{\mathsf{Copy}^3_{trm}}\ i\ ld\ d\ ls\ s \stackrel{def}{=}\\ &\quad \mathbf{(i)}\ s.func = Some(Trm(TRM_{copy}, prm_{trm}(i)))\wedge\\ &\quad \mathbf{(ii)}\ ld \subseteq s.mem(OSID) \wedge \mathbf{(iii)}\ ls \subseteq s.mem(i)\wedge\\ &\quad \mathbf{(iv)}\ d = clt(s.data_{mem}, ls)\\ &\mathbf{Act}_{\mathsf{Copy}^3_{trm}}\ i\ ld\ d\ ls\ s\ s' \stackrel{def}{=}\\ &\quad newd = update_{mem}(s.data_{mem}, ld, d)\wedge\\ &\quad s \sim_{func, data_{mem}=Some(Trm(TRM_{lod}, prm_{trm}(i))), newd}\ s'.\end{aligned} \tag{29}$$

The step is refined from SWrite^2_{mem}. Here private data of SCA *i* may be in *d*.

Then, in the step of switching the mode of the OS to executing mode, the set of data stored in memory allocated for the OS is added to the knowledge of the OS.

$$\begin{aligned}&\mathbf{Grd}_{\mathsf{StoE}^3_{trm}}\ i\ s \stackrel{def}{=}\\ &\quad \mathbf{(i)}\ s.func = Some(Trm(TRM_{StoE}, prm_{trm}(i)))\\ &\mathbf{Act}_{\mathsf{StoE}^3_{trm}}\ i\ s\ s' \stackrel{def}{=}\\ &\quad newc = (s.core(OSID) := s.core(OSID) \cup \{UCORE\})\wedge\\ &\quad newk = (s.know(OSID) := s.know(OSID)\cup\\ &\quad s.data_{regs}(UCORE) \cup clt(s.data_{mem}, s.mem(OSID)))\wedge\\ &\quad s \sim_{func, core, know=NONE, newc, newk}\ s'.\end{aligned} \tag{30}$$

We briefly illustrate the reason why data confidentiality is violated in these steps. Since it can be proved that StoE^3_{trm} occurs only if Copy^3_{trm} have occurred, it implies that *d* in Copy^3_{trm} is a subset of $clt(s.data_{mem}, s.mem(OSID))$, *i.e.*, the set of data stored in memory allocated for the OS. Therefore, private data in *d* may be added to the knowledge of the OS.

In design of TrustVisor, it is suggested that SCAs encrypt private data of themselves in their outputs by calling hv_{seal}. To validate the suggestion, we replace the step Copy^3_{trm} with $\mathsf{CopyStrn}^3_{trm}$ in the formal model.

$$\begin{aligned}&\mathbf{Grd}_{\mathsf{CopyStrn}^3_{trm}}\ i\ ld\ d\ ls\ k\ s \stackrel{def}{=}\\ &\quad \mathbf{(i)}\ s.func = Some(Trm(TRM_{copy}, prm_{trm}(i)))\wedge\\ &\quad \mathbf{(ii)}\ ld \subseteq s.mem(OSID) \wedge \mathbf{(iii)}\ ls \subseteq s.mem(i)\wedge\\ &\quad \mathbf{(iv)}\ d = clt(s.data_{mem}, ls) \wedge \mathbf{(v)}\ s.symkey = Some(k)\wedge\\ &\quad \mathbf{(vi)}\ (\forall d_1.d_1 \in d \wedge (parts(d_1) \cap pdata(i) \neq \phi) \rightarrow\\ &\quad \exists d_2.d_1 = Enc(k, Cons(d_2, Id(i))))\\ &\mathbf{Act}_{\mathsf{CopyStrn}^3_{trm}}\ i\ ld\ d\ ls\ k\ s\ s' \stackrel{def}{=}\\ &\quad newd = update_{mem}(s.data_{mem}, ld, d)\wedge\\ &\quad s \sim_{func, data_{mem}=Some(Trm(TRM_{lod}, prm_{trm}(i))), newd}\ s'.\end{aligned} \tag{31}$$

Compared with Copy^3_{trm}, guard **(v)** and **(vi)** are added. The guard **(v)** states that TrustVisor has generated the encryption key used in hv_{seal}. The guard **(vi)** models the suggestion that if d_1 contains SCA *i*'s private data, then d_1 should be the output of hv_{seal} called by SCA *i*. Finally, data confidentiality is proved to be preserved when the guards are added.

5.2 Case 2: OSP

OSP is a shielding system that aims to overcome the weakness of TrustZone-based approaches and hypervisor-based approaches in ensuring safe execution of security sensitive codes (SCCs), *i.e.*, SCAs, on mobile devices. TrustZone-based approaches bloat the TCB of the system as they must increase the code base size of the most privileged software. The most privileged software is used for supporting the execution of SCCs in the secure world. Hypervisor-based approaches incur performance overhead on mobile devices that are already suffering from resource restrictions. Therefore, OSP uses a hybrid approach that utilizes both TrustZone and a hypervisor to not only avoid executing SCCs in the secure world, but also mitigate performance overhead by activating the hypervisor. Specifically, OSP consists of the OSP hypervisor, which protects and manages the SCCs, and the OSP core, which controls and configures OSP overall. The OSP core resides in the secure world, while the OSP hypervisor is implemented in the normal world for an additional TEE for executing SCCs. Thus the TCB bloating of the secure world is suppressed. The OS in the normal world cannot access resources, *e.g.*, memory, used in the TEE. OSP deactivates its hypervisor when there is no SCC that is executing, and therefore, performance overhead in activating the hypervisor is mitigated.

We now explain OSP's functions related to our goals as follows.

1) *SCC_register*: An SCA is registered by the OS by calling *SCC_register*. When the OS calls *SCC_register*, OSP prepares the execution environment of a new SCA. OSP allocates memory for the SCA, initializes contexts, *i.e.*, data to be written to registers, for the SCA and resumes the OS. The allocated memory is reserved for all SCAs, and isolated from memory allocated for the OS.

2) *SCC_invoke*: After *SCC_register* for an SCA, the SCA can be invoked by the OS. When the OS calls *SCC_invoke*, OSP performs two phases of execution. In the first phase, a CPU core originally used by the OS is stopped and assigned to the selected SCA by OSP. OSP saves the context of the OS for the core, loads the context of the SCA, and executes the SCA. Besides, the OS produces the input for the SCA to the memory region specified by parameters of *SCC_invoke*. OSP copies the input to memory allocated for the SCA. If the SCC finishes its work, OSP performs the second phase. Specifically, the SCA turns into suspending mode, and the CPU core originally used by the SCA is reassigned to the OS. OSP saves the context of the SCA, loads the context of the

OS for the core and makes the OS use the core. Besides, the SCA produces the output for the OS to the memory region specified by OSP. Then, OSP copies the output to the memory allocated for the OS. Note that the OS may use several CPU cores simultaneously, and therefore when the OS calls *SCC_invoke*, the OS may not be turned into suspending mode.

3) *SCC_unregister*: OSP completely clears every relevant state of an SCA specified by parameters of *SCC_unregister*. When the OS calls *unregistration*, OSP deallocates all memory originally allocated for the SCA, clears data in the deallocated memory, and resumes the OS.

States and Events: The record for states for OSP is similar to S^3_{tv}, except that *symkey* is unused, since OSP does not provide hv_{seal} or hv_{unseal}. We also omit the modeling of events, for it is similar to the modeling of events in TrustVisor as well.

Results: We also discover a potential security threat in using the function *SCC_invoke*. The threat is quite similar to the one we noticed in TrustVisor. Formally, first, if the OS is in executing mode when OSP copies the output of the SCA, d is added to the knowledge of the OS.

$$\begin{aligned}&\mathbf{Grd}_{\mathsf{Copy2}^3_{ivk}}\ i\ idc\ ld\ d\ ls\ s \stackrel{def}{=}\\ &\quad \mathbf{(i)}\ s.func = Some(Ivk(IVK_{copy2}, prm_{ivk}(i, idc)))\wedge\\ &\quad \mathbf{(ii)}\ ld \subseteq s.mem(OSID) \wedge \mathbf{(iii)}\ ls \subseteq s.mem(i)\wedge\\ &\quad \mathbf{(iv)}\ d = clt(s.data_{mem}, ls)\\ &\mathbf{Act}_{\mathsf{Copy2}^3_{ivk}}\ i\ idc\ ld\ d\ ls\ s\ s' \stackrel{def}{=}\\ &\quad newd = update_{mem}(s.data_{mem}, ld, d)\wedge\end{aligned} \tag{32}$$

If $s.core(OSID) \neq \phi$
then $newk$=($s.know(OSID)$:=$s.know(OSID)\cup\ d$)
else $newk$=$s.know$) $\wedge\ s \sim func, data_{mem},\ know = Some(Ivk(IVK_{lod2}, prm_{ivk}(i, idc))),\ newd, newks'$.

Second, if the OS is in suspending mode when $\mathsf{Copy2}^3_{ivc}$ occurs, d is written to ld for $\mathsf{Copy2}^3_{ivc}$.

$$\begin{aligned}&\mathbf{Grd}_{\mathsf{RC2}^3_{ivk}}\ i\ idc\ s \stackrel{def}{=}\\ &\quad \mathbf{(i)}\ s.func = Some(Ivk(IVK_{RC}, prm_{ivk}(i, idc)))\\ &\mathbf{Act}_{\mathsf{RC2}^3_{ivk}}\ i\ idc\ s\ s' \stackrel{def}{=}\\ &\quad newc = (s.core(OSID) := s.core(OSID) \cup \{idc\})\wedge\\ &\quad (\mathsf{if}\ s.core(OSID) \neq \phi\ \mathsf{then}\\ &\quad newk = (s.know(OSID) := s.know(OSID)\cup\\ &\quad s.data_{regs}(idc))\\ &\quad \mathsf{else}\ \ newk = (s.know(OSID) := s.know(OSID)\cup\\ &\quad s.data_{regs}(idc) \cup clt(s.data_{mem}, s.mem(OSID))))\wedge\\ &\quad s \sim func, core, know = NONE, newc, newks'\end{aligned} \tag{33}$$

It can be proved that d in $\mathsf{Copy2}^3_{ivk}$ is a subset of $clt(s.data_{mem}, s.mem(OSID))$ in $\mathsf{RC2}^3_{ivk}$. Therefore, private data in d may be added to the knowledge of the OS.

In design of OSP, it is suggested that SCAs encrypt private data of themselves in their outputs. We assume that SCAs encrypt private data in the output by using their private symmetric keys. Formally, we replace the step $\mathsf{Copy2}^3_{ivk}$ with $\mathsf{Copy2Strn}^3_{ivk}$ in the formal model.

$$\begin{aligned}&\mathbf{Grd}_{\mathsf{Copy2Strn}^3_{ivk}}\ i\ idc\ ld\ d\ ls\ s \stackrel{def}{=}\\ &\quad \mathbf{(i)}\ s.func = Some(Ivk(IVK_{copy}, prm_{ivk}(i, idc)))\wedge\\ &\quad \mathbf{(ii)}\ ld \subseteq s.mem(OSID) \wedge \mathbf{(iii)}\ ls \subseteq s.mem(i)\wedge\\ &\quad \mathbf{(iv)}\ d = clt(s.data_{mem}, ls)\wedge\\ &\quad \mathbf{(v)}\ (\forall d_1.d_1 \in d \wedge (parts(d) \cup s.pdata(i) \neq \phi) \rightarrow\\ &\quad \exists d_2.\exists k.d_1 = Enc(k, d_2)\wedge\\ &\quad k = invKey(k) \wedge Key(k) \in s.pdata(i)).\\ &\mathbf{Act}_{\mathsf{Copy2Strn}^3_{ivk}}\ i\ idc\ ld\ d\ ls\ s\ s' \stackrel{def}{=}\\ &\quad newd = update_{mem}(s.data_{mem}, ld, d)\wedge\\ &\quad (\mathsf{if}\ s.core(OSID) \neq \phi\ \mathsf{then}\\ &\quad newk = (s.know(OSID) := s.know(OSID) \cup d)\\ &\quad \mathsf{else}\ \ newk = s.know)\wedge\\ &\quad s \sim func, data_{mem},\\ &\qquad know = Some(Ivk(IVK_{lod}, prm_{ivk}(i, idc))),\\ &\qquad newd, newks'.\end{aligned} \tag{34}$$

Here, $\mathsf{Copy2Strn}^3_{ivk}$ refines SWrite^2_{mem} in *S2*. Compared with $\mathsf{Copy2}^3_{ivk}$, guard **(v)** is added. Guard **(v)** states that if d_1 contains SCA i's private data, then d_1 should be a ciphertext encrypted by a symmetric key that is privately owned by SCA i. Finally, we prove the validity of $\mathsf{Copy2Strn}^3_{ivk}$.

6 Related Work

6.1 Shielding Systems

The current shielding systems can be divided into two categories: (1) Systems using modern instructions in Intel or AMD chips. (2) Systems using TrustZone Technology.

Intel and AMD chips: Flicker [22] leverages Trusted Platform Module (TPM) and Intel TXT [17] or AMD SVM [30] to execute security sensitive code in isolation with the OS. Since it uses new features of processors, specially designed hardware or modifications for the OS are not needed in protecting applications, and it only requires that as few as 250 lines of additional code are trusted. TrustVisor [21] provides code integrity as well as data integrity and confidentiality for selected portions of an application. The goal is to leverage the features of modern processors to overcome the tradeoff between achieving a high level of security and high performance. It is achieved by implementing a software-based "micro-TPM" which

attests the existence of isolated execution to an external entity.

InkTag [15] is proposed to directly address the Iago attacks [8] in systems that solely protect memory of applications from untrusted OS. It simplifies the design of hypervisor by forcing the untrusted operating system to participate in its own verification. Haven [5] enables users to run applications on cloud hosting services without having to trust the service provider. It protects confidentiality and integrity of the user's applications from the platform on which it runs (i.e., the cloud service provider's OS, VM and firmware). MiniBox [20] is the first two-way sandbox for x86 native code, which not only protects a benign OS from a misbehaving application, but also protects an application from a malicious OS. MiniBox can be applied in Platform-as-a-Service cloud computing to provide two-way protection between a customer application and the cloud platform OS.

TrustZone: Though the secure world of ARM TrustZone [1] is used for executing security critical applications, the increased number of the applications makes the size of the most privileged software in the secure world complex and therefore, vulnerable. Hence, shielding systems, such as OSP [11] and PrivateZone [18], are proposed for handling the problem. OSP [11] relies on a hybrid approach that utilizes both TrustZone and a hypervisor to implement an additional execution environment for securely executing applications.

This scheme, called on-demand hypervisor activation, has been efficiently and securely implemented by leveraging the memory protection capability of TrustZone. PrivateZone [18] is a framework to enable individual developers to utilize TrustZone resources. Using PrivateZone, developers can run Security Critical Logics (SCL) in a Private Execution Environment (PrEE). The advantage of PrivateZone is its leveraging of TrstZone resources without undermining the security of existing services in the TEE.

6.2 Refinement of Security Systems

The design of TAP [28] is motivated by the phenomenon that recent proposals for trusted hardware platforms, such as Intel SGX [23] and the MIT Sanctum processor, offer compelling security features but lack formal guarantees. It is proved that SGX and Sanctum are refinements of TAP under certain parameterizations of the adversary, demonstrating that these systems implement secure enclaves for the stated adversary models. Specifically, TAP satisfies three security properties that entail secure remote execution: integrity, confidentiality and secure measurement. TAP is currently limited to concurrent execution on a single-threaded single-core processor.

Klein *et al.* [19] present post-hoc verification of the seL4 microkernel from an abstract specification down to its C implementation. The functional correctness is verified that the implementation of seL4 always strictly follows the high-level abstract specification of kernel behavior. Here, the refinement is used to prove the conformance between formalizations at different levels.

Zhao *et al.* [33] propose a security model for information flow security in certification of separation kernels and a refinement framework on ARINC 653 compliant Separation Kernels (ARINC SKs). According to code-to-spec review, they find six security flaws in the ARINC 653 standard and three flaws in ARINC SK implementations.

Refinement and verifications on security protocols have been studied as well. Sprenger *et al.* [27] propose to verify security protocols by stepwise refinement. Their refinement strategy guides the transformation of abstract security goals into protocols that are secure when operating over an insecure channel controlled by a Dolev-Yao-style intruder. They have implemented their method in Isabelle/HOL and used it to develop different entity authentication and key transport protocols. Huang *et al.* [16] make fine-grained refinement on TPM-based security protocols on the application level. The purpose is to guide the design of TPM-based protocol applications, which are generally security-critical and error-prone in implementation. The framework introduces a modified Dolve-Yao adversary model, where the normal entities outside TPM may also perform malicious operations.

6.3 Verifications without Refinement

There are also researches on formally verifying security systems without refinement [31, 32]. Barthe *et al.* [3, 4] formalize in the Coq proof assistant an idealized model of a hypervisor, and formally establish that the hypervisor ensures strong isolation properties between the different operating systems, and guarantees that requests from guest operating systems are eventually attended. Sinha *et al.* [25, 26] formally verifies confidentiality of applications running on Intel SGX. The main concerns are vulnerabilities of divulging secrets in the application caused by incorrect use of SGX instructions or memory safety errors.

7 Conclusions

We develop a formal framework for analyzing security properties ensured by shielding systems. We analyze the property of memory isolation and data confidentiality, and propose four refinement steps for guiding verification of shielding systems. Potential security threats in using the systems are found.

One of our future work is to refine the shielding systems into fine-grained pseudo-code level with full verification, in which the soundness of specific goals in each shielding system is also proved. We also plan to extend our framework with more realistic hypotheses to support side-channel attacks [2, 9, 24] towards shielding systems.

Acknowledgments

The research is supported by National Natural Science Foundation of China under Grant No.61572453, No.61202404, No.61520106007, No.61170233, No.61232018, No.61572454, and Anhui Provincial Natural Science Foundation, No.1508085SQF215. We gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] T. Alves, "Trustzone: Integrated hardware and software security," *White Paper*, 2004. (`https://www.researchgate.net/publication/244521018_Trustzone_Integrated_Hardware_and_Software_Security`)

[2] S. Arnautov, B. Trach, F. Gregor, T. Knauth, A. Martin, C. Priebe, J. Lind, D. Muthukumaran, D. O. Keeffe, M. Stillwell, D. Goltzsche, D. M. Eyers, R. Kapitza, P. R. Pietzuch, and C. Fetzer, "SCONE: Secure linux containers with intel SGX," in *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI'16)*, pp. 689–703, Nov. 2016.

[3] G. Barthe, G. Betarte, J. D. Campo, and C. Luna, "Formally verifying isolation and availability in an idealized model of virtualization," in *17th International Symposium on Formal Methods (FM'11)*, pp. 231–245, June 2011.

[4] G. Barthe, G. Betarte, J. D. Campo, and C. Luna, "Cache-leakage resilient OS isolation in an idealized model of virtualization," in *25th IEEE Computer Security Foundations Symposium (CSF'12)*, pp. 186–197, June 2012.

[5] A. Baumann, M. Peinado, and G. C. Hunt,Shielding applications from an untrusted cloud with haven in *11th USENIX Symposium on Operating Systems Design and Implementation (OSDI'14)*, pp. 267–283, Oct. 2014.

[6] G. Bella, "Formal correctness of security protocols," *Information Security and Cryptography*, 2007. (`https://www.springer.com/gb/book/9783540681342`)

[7] E. Boiten and J. Abrial, "Modeling in event-b system and software engineering," *Journal of Functional Programming*, vol. 22, no. 2, pp. 217, 2012.

[8] S. Checkoway and H. Shacham, "Iago attacks: Why the system call API is a bad untrusted RPC interface," in *18th ACM International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS'13)*, pp. 253–264, Mar. 2013.

[9] S. Chen, X. Zhang, M. K. Reiter, and Y. Zhang, "Detecting privileged side-channel attacks in shielded execution with déjá vu," in *Proceedings of the ACM on Asia Conference on Computer and Communications Security (ASIACCS'17)*, pp. 7–18, Apr. 2017.

[10] Y. Cheng, X. Ding, and R. H. Deng, "Efficient virtualization-based application protection against untrusted operating system," in *Proceedings of the ACM on Asia Conference on Computer and Communications Security (ASIACCS'15)*, pp. 345–356, Apr. 2015.

[11] Y. Cho, J. Shin, D. Kwon, M. Ham, Y. Kim, and Y. Paek, "Hardware-assisted on-demand hypervisor activation for efficient security critical code execution on mobile devices," in *USENIX Annual Technical Conference (USENIX ATC'16)*, pp. 565–578, June 2016.

[12] D. Dolev and A. Yao, "On the security of public key protocols," *IEEE Transactions on Information Theory*, vol. 29, no. 2, pp. 198–208, 1983.

[13] G. Kahn G. Huet and C. Paulin-Mohring, *The Coq Proof Assistant: A Tutorial: Version 7.2*, RT-0256, 2002.

[14] L. Guan, P. Liu, X. Xing, X. Ge, S. Zhang, M. Yu, and T. Jaeger, "Trustshadow: Secure execution of unmodified applications with ARM trustzone," in *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services (MobiSys'17)*, pp. 488–501, June 2017.

[15] O. S. Hofmann, S. Kim, A. M. Dunn, M. Z. Lee, and E. Witchel, "Inktag: secure applications on an untrusted operating system," in *18th ACM International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS'13)*, pp. 265–278, Mar. 2013.

[16] W. Huang, Y. Xiong, X. Wang, F. Miao, C. Wu, X. Gong, and Q. Lu, "Fine-grained refinement on tpm-based protocol applications," *IEEE Trans. Information Forensics and Security*, vol. 8, no. 6, pp. 1013–1026, 2013.

[17] C. Intel, "Lagrande technology preliminary architecture specification," *Intel Corporation*, 2006. (`http://kib.kiev.ua/x86docs/SDMs/315168-002.pdf`)

[18] J. Jang, C. Choi, J. Lee, N. Kwak, S. Lee, Y. Choi, and B. Kang, "Privatezone: Providing a private execution environment using arm trustzone," *IEEE Transactions on Dependable & Secure Computing*, no. 99, pp. 1–1, 2016.

[19] G. Klein, J. Andronick, K. Elphinstone, T. Murray, T. Sewell, R. Kolanski, and G. Heiser, "Comprehensive formal verification of an os microkernel," *ACM Transactions on Computer Systems*, vol. 32, no. 1, pp. 2, 2014.

[20] Y. Li, J. M. McCune, J. Newsome, A. Perrig, B. Baker, and W. Drewry, "Minibox: A two-way sandbox for x86 native code," in *USENIX Annual Technical Conference (USENIX ATC'14)*, pp. 409–420, June 2014.

[21] J. M. McCune, Y. Li, N. Qu, Z. Zhou, A. Datta, V. D. Gligor, and A. Perrig, "Trustvisor: Efficient TCB reduction and attestation," in *31st IEEE Symposium on Security and Privacy (S&P'10)*, pp. 143–158, May 2010.

[22] J. M. McCune, B. Parno, A. Perrig, M. K. Reiter, and H. Isozaki, "Flicker: An execution infrastructure for tcb minimization," in *Proceedings of the EuroSys Conference (EuroSys'08)*, pp. 315–328, Apr. 2008.

[23] F. McKeen, I. Alexandrovich, I. Anati, D. Caspi, S. Johnson, R. Leslie-Hurd, and C. Rozas, "Intel® software guard extensions (intel® sgx) support for dynamic memory management inside an enclave," in *Proceedings of the Hardware and Architectural Support for Security and Privacy (HASP'16)*, pp. 10, June 2016.

[24] M. Shih, S. Lee, T. Kim, and M. Peinado, "T-SGX: eradicating controlled-channel attacks against enclave programs," in *24th Annual Network and Distributed System Security Symposium (NDSS'17)*, 2017. (`https://www.cc.gatech.edu/~slee3036/papers/shih:tsgx.pdf`)

[25] R. Sinha, M. Costa, A. Lal, N. P. Lopes, S. Rajamani, S. A. Seshia, and K. Vaswani, "A design and verification methodology for secure isolated regions," in *Proceedings of the 37th ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'16)*, pp. 665–681, June 2016.

[26] R. Sinha, S. K. Rajamani, S. A. Seshia, and K. Vaswani, "Moat: Verifying confidentiality of enclave programs," in *22nd ACM Conference on Computer and Communications Security (CCS'15)*, pp. 1169–1184, June 2015.

[27] C. Sprenger and D. A. Basin, "Developing security protocols by refinement," in *17th ACM Conference on Computer and Communications Security (CCS'10)*, pp. 361–374, Oct. 2010.

[28] P. Subramanyan, R. Sinha, I. A. Lebedev, S. Devadas, and S. A. Seshia, "A formal foundation for secure remote execution of enclaves," in *24th ACM Conference on Computer and Communications Security (CCS 2017)*, pp. 2435–2450, 2017.

[29] R. Uhlig, G. Neiger, D. Rodgers, A. L. Santoni, F. Martins, A. Anderson, S. M. Bennett, A. Kagi, F. H. Leung, and L. Smith, "Intel virtualization technology," *Computer*, vol. 38, no. 5, pp. 48–56, 2005.

[30] A. Virtualization, "Secure virtual machine architecture reference manual," *AMD Publication*, vol. 33047, 2005.

[31] C. Wang, Y. Xiong, W. Cheng, W. Huang, H. Xia, and J. Huang, "A general formal framework of analyzing selective disclosure attribute-based credential systems," *International Journal of Network Security*, vol. 19, no. 5, pp. 794–803, 2017.

[32] M. Wu, J. Chen, and R. Wang, "An enhanced anonymous password-based authenticated key agreement scheme with formal proof," *International Journal of Network Security*, vol. 19, no. 5, pp. 785–793, 2017.

[33] Y. Zhao, D. Sanan, F. Zhang, and Y. Liu, "Refinement-based specification and security analysis of separation kernels," *IEEE Transactions on Dependable & Secure Computing*, no. 99, pp. 1–1, 2017.

Biography

Jiabin Zhu is a Ph.D. candidate in school of Computer Science and Technology, University of Science and Technology of China. His current research interests formal methods and information security.

Wenchao Huang received the B.S. and Ph.D degrees in computer science from University of Science and Technology of China in 2005 and 2011, respectively. He is currently an associate professor in School of Computer Science and Technology, University of Science and Technology of China. His current research interests include mobile computing, information security, trusted computing and formal methods.

Fuyou Miao received his Ph.D of computer science from University of Science and Technology of China in 2003. He is an associate professor in the School of Computer Science and Technology, University of Science and Technology of China. His research interests include applied cryptography, trusted computing and mobile computing.

Cheng Su is a Ph.D. candidate in school of Computer Science and Technology, University of Science and Technology of China. His current research interests formal methods and information security.

Baohua Zhao received His M.S. Degree from Beijing University of Technology in 2016. He is a director in Computing Technology and Applications Research Institute, Global Energy Interconnection Research Institute. His main research interests include trusted computing, information security and computer technology.

Yan Xiong received the B.S., M.S., and Ph.D degrees from University of Science and Technology of China in 1983, 1986 and 1990 respectively. He is a professor in School of Computer Science and Technology, University of Science and Technology of China. His main research interests include distributed processing, mobile computing, computer network and information security.

StegoNote: Steganography in Guitar Music Using Note Modulation

Hui Tian[1], Zhaohua Zhu[1], Chin-Chen Chang[2], Yongfeng Huang[3], Tian Wang[1], Yonghong Chen[1], and Yiqiao Cai[1]

(Corresponding author: Hui Tian)

College of Computer Science and Technology, National Huaqiao University[1]
Xiamen 361021, China
Department of Information Engineering and Computer Science, Feng Chia University[2]
Taichung 40724, Taiwan
Department of Electronic Engineering, Tsinghua University[3]
Beijing, 100084, China
(Email: cshtian@gmail.com)

(Received May 20, 2018; Revised and Accepted Sept. 22, 2018; First Online July 16, 2019)

Abstract

Due to its diversity, sensual and physical redundancies, music is considered as a type of ideal carrier for steganography, and has attracted increasing attention from the research community of information hiding. In this paper, we present a novel note-modulating steganographic scheme for guitar music. Differing from the existing works, the proposed scheme conceals secret messages into guitar accompaniments based on the fact that there are many note combinations available for expressing a group of similar harmony effects. Specifically, the proposed scheme first determines the available tones for information hiding, and then embeds the secret messages by modulating the note combination of each candidate tone with matrix embedding strategies. The embedding process has no appreciable impact on the playing effect of the music, because only a small part of the musical tones in an accompaniment are substituted by the other note combinations that can achieve similar harmony effects. The proposed scheme is further evaluated with thirty guitar-music samples collected from the Internet. The experimental results demonstrate that the proposed scheme is feasible and efficient. Particularly, employing an appropriate matrix embedding strategy, the proposed scheme can achieve a good balance between steganographic transparency and capacity.

Keywords: Guitar Music; Information Hiding; Music Steganography; Note Modulation

1 Introduction

Steganography is the art and science of concealing secret messages into normal carriers without imperceptible changes [19]. In contrast with cryptographic techniques that aim to protect the content of secret messages [11, 23, 34, 36], it focuses on hiding the very existence of the messages. Therefore, to an extent, steganography can render better security for communications [27], and have thereby attracted increasing attention from various research communities. So far, lots of research works on steganography have been carried out, and the candidate carriers have been also on the increase [39]. Almost all digital media (*e.g.*, image [2, 3, 6, 9, 10, 20, 21, 24, 28, 29, 32, 33], video [16, 38], audio [4, 37], text [12, 14], and Internet protocol [13, 15]) can be considered as steganographic carriers. In this paper, we focus on the steganography on music, which, compared with the existing steganographic techniques on traditional carriers, is largely unexplored but promising.

As is well known, music is a ubiquitous art for people to express their mood, emotion and feeling [31], which has various types and styles. The diversity of music provides an excellent condition for steganography. Moreover, for music, there are both sensual and physical redundancies, making the embedding of secret messages feasible. Specifically, for the same music, different people have diverse feelings, so a slight change will not attract the notice of people; moreover, both melody and harmony can be modulated to hide secret messages while achieving specific music effects. Therefore, the music can be considered as a type of ideal carrier for steganography. It is worth pointing out that, music steganography is different from audio steganography, since the former aims to conceal secret message into the musical content while the latter embeds the secret message into audio signals.

Recently, music steganography has also attracted increasing attention. Generally, the existing works regard-

ing music steganography can be divided into three categories. The first one modulates the pitches of the notes to hide secret messages. For example, Bach, a well-known German musician, uses a note sequence of B^b - A - C - $B^{\#}$ in his music to represent his name [7]; Hutchinson [8] proposed a scheme based on note modulation, which assigns musical notes to the letters of the embedded message according to their appearance frequency. It is worth noting that the steganographic music, if containing some inappropriate note modulations, will sound weird [17]. Thus, it is advisable to ascertain the correlations of the notes prior to modulating them for information hiding. The second one modifies the music elements (*e.g.*, duration and loudness of the notes) to conceal secret message, behind which the main idea is to exploit the auditory redundancies of these elements to hide the existence of the information hiding. For example, Adli *et al.* [1] proposed a steganographic method for MIDI files by modifying the loudness parameters for "note on" commands, which can be considered as a loudness-modulating method; Moreover, the authors pointed out that the repeated and exclusive commands can be also used to embed secret messages; In addition, Yamamoto *et al.* [35] proposed an adaptive steganographic approach to embed the secret message by modulating the duration of notes; Szczypiorski [25] designed a new steganographic scheme for club music, which embeds secret data into music beats in a subtle way. As mentioned above, this type of methods, due to taking advantage of the auditory redundancies of music, can provide good embedding transparency. Surprisingly, however, all the existing schemes focus on the MIDI music files. The third one conceals the secret messages into sheet music. The sheet music is a handwritten or printed form of music notation that employs musical symbols to indicate the musical content, such as pitches, rhythms and chords, whose purpose is to illustrate the performance skills accurately. For a given sheet music (also called music score), people mainly concentrate on its content, but nearly pay no attention to its typesetting style and visual quality. Therefore, the sheet music is an ideal carrier for steganography. For example, Funk *et al.* [5] proposed an information hiding technique for scanned music scores, which is essentially an image-based steganographic method. Of course, we can easily infer that it is also possible to embed secret messages by modulating the typesetting style (*e.g.*, note spacing and note size).

In this paper, we present a novel note-modulating steganography scheme for guitar music. Differing from the existing works, the proposed scheme aims to embed secret messages into guitar accompaniments, *i.e.*, the musical parts providing harmonic support for the melody. Generally, the harmony of music comes from the simultaneous sounding of multiple notes, and involves chord constructions as well as chord progressions. That is, there are many candidate combinations for notes to express a group of similar harmony effects. Thus, we can achieve information hiding by note modulation. Moreover, we introduce the Hamming matrix encoding strategy to further reduce the embedding distortions. To our best knowledge, this is the first steganographic scheme using the Guitar accompaniments. We evaluate the performance of the proposed scheme with thirty guitar-music samples. The results demonstrate that the proposed scheme is feasible and efficient. Particularly, employing an appropriate matrix embedding strategy, the proposed scheme can achieve a good balance between steganographic transparency and capacity.

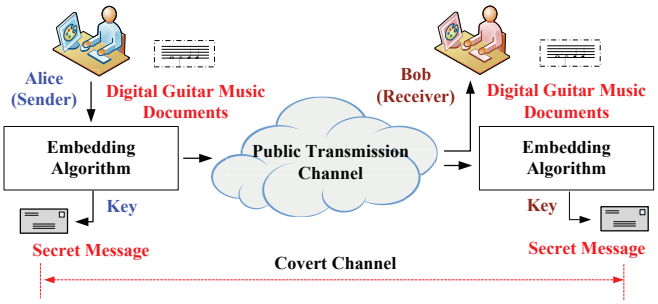


Figure 1: Steganography on music

The rest of this paper is organized as follows. The note-modulating steganographic method for Guitar music is proposed in Section 2, which is followed by evaluation criteria and the experimental results that are presented in Section 3. Finally, we conclude this paper in Section 4.

2 Proposed Scheme

Figure 1 depicts a general framework of steganography on music. Let us assume that Alice as the sender wants to send some secret messages to Bob as the receiver through a public but insecure channel. To achieve this goal, Alice hides secret messages into a piece of music (*e.g.*, guitar music in this paper) using an embedding algorithm, and send the steganographic music to Bob through the public channel; Upon receiving the steganographic music, Bob can obtain secret messages with the corresponding extracting algorithm. In this paper, we present a note-modulating steganographic scheme to achieve steganography on guitar music, which is described as follows.

Assume that Alice wants to send L_M bits of secret messages $M = \{m_i = 0 \ or \ 1 \mid i = 1, 2, \ldots, L_M\}$ to Bob by embedding them into a piece of guitar music Θ. Note that the secret messages are often encrypted prior to embedding, which, however, is independent of the proposed scheme. Thus, we omit the encryption process in this paper, and consider M as the secure form of the given messages for short. Let the accompaniment of the music be $A = \{T_1, T_2, \ldots, T_N\}$, where T_i is the i-th tone, and N is the number of tones in A; Let the chord progression of the music be $\Lambda = \{C_i \mid i = 1, 2, \ldots, L_C\}$, where L_C is the number of the chords in Θ.$C_i = \{c_{i,j} \mid j = 1, 2, \ldots, r_i, 3 \leq r_i \leq 6\}$, $c_{i,j}$ is the j-th

note (chord member) of the i-th chord, and r_i is the number of notes in C_i. In the j-th chord ($j = 1, 2, \ldots, L_C$), let the number of tones be n_j, then the accompaniment can be also denoted as $A = \{\mathcal{T}_j \mid j = 1, 2, \ldots, L_C\}$, where $\Sigma_{j-1}^{L_C} n_j = N$, $\mathcal{T}_1 = \{T_l \mid 1 \leq l \leq n_1\}$, and $\mathcal{T}_j = \{T_l \mid \Sigma_{k=1}^{j-1} n_k + 1 \leq l \leq \Sigma_{k=1}^{j} n_k\}$. Accordingly, the embedding process can be described as follows.

Step 1. Decision on available cover tones: With a key K shared by the communication parties, the sender first generates a random binary sequence $S_1 = \{s_{1,j} \mid j = 1, 2, \ldots, N\}$ intended for determining the embedding positions, namely, which tones are chosen to hide information. Note that the random binary sequence S_1 can be also generated according to the desired embedding rate. Assume that the embedding rate is ξ. For each tone, the sender generates a random number $p_j \in [0, 1]$. If $p_j \leq \xi$, then $s_{1,j} = 1$; otherwise, $s_{1,j} = 0$. For each tone T_j in the i-th chord, $\Sigma_{k=1}^{i-1} n_k + 1 \leq j \leq \Sigma_{k=1}^{i} n_k$, we determine the embedding factor λ_j as

$$\lambda_j = \alpha_j \wedge \beta_j \wedge \gamma_j \wedge s_{1,j}, \tag{1}$$

where "$\wedge$" means the AND operation; if T_j is the first tone of the i-th chord, $\alpha_j = 0$, otherwise, $\alpha_j = 1$; if the number of notes involved in T_j is equal to r_i, then $\beta_j = 0$, otherwise, $\beta_j = 1$; if T_j contains the notes not belonging to the i-th chord C_i, then $\gamma_j = 0$, otherwise, $\gamma_j = 1$. If $\lambda_j = 1$, T_j is available for hiding information; otherwise, it cannot be used. For ease of description, we denote the set of all available cover tones as $B = \{b_1, b_2, \ldots, b_{L_B}\}$.

Step 2. Matrix embedding: In our scheme, we employ the Hamming matrix encoding strategy (MES) [26, 30] to achieve information hiding with the minimum distortion. Assume that B is divided into U parts with a length of y shared by the communication parties, namely, $B = \{\mathcal{B}_1, \mathcal{B}_2, \ldots, \mathcal{B}_U\}$, where $\mathcal{B}_i = \{\mathfrak{b}_{i,1}, \mathfrak{b}_{i,2}, \ldots, \mathfrak{b}_{i,y}\}$, $i = 1, 2, \ldots, U$, $U = \lfloor L_B/y \rfloor$, $\mathfrak{b}_{i,j} = b_{((i-1)\times y+j)}$, $j = 1, 2, \ldots, y$. Note that, using MES, z bits of secret messages can be embedded into 2^z–1 bits of cover with no more than 1-bit change. That is, for each tone part $\mathcal{B}_i$, $z = \lfloor \log_2(y+1) \rfloor$ bits of secret messages can be embedded. If $y \geq y' = 2^z$–1, we only use the first y' tones in each tone part $\mathcal{B}_i$ as the cover. In this paper, we denote the adopted MES as MES (y', z) for short. Accordingly, M is divided into V parts, namely, $M = \mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_V$, where $\mathcal{M}_i = \mathfrak{m}_{i,1}, \mathfrak{m}_{i,2}, \ldots, \mathfrak{m}_{i,z}$, $i = 1, 2, \ldots, V$. $V = \lfloor L_M/z \rfloor$, $m_{i,j} = m_{((i-1)\times z+j)}$, $j = 1, 2, \ldots, z$. Note that V should be not more than U so that all the secret messages can be embedded. That is to say, the maximum embedding capacity is $z \times U$. For each tone part $\mathcal{B}_i$, the embedding process with MES can be described as follows:

Step 2.1: For $\mathcal{B}_i$, determine the state vector $W_i = \{w_{i,1}, w_{i,2}, \ldots, w_{i,y'}\}$, where

$$w_{i,j} = g(\mathfrak{b}_{i,j}) \ MOD \ 2, j = 1, 2, \ldots, y'. \tag{2}$$

In Equation (2), $g(\mathfrak{b}_{i,j})$ is the sequence number of $b_{i,j}$ in the set of all possible note combinations for the i-th chord. Let the number of notes in $b_{i,j}$ be t. Then, the number of all note combinations is $\mathcal{C}_{r_i}^t$. Table 1 shows all note combinations and their sequence numbers for various values of t and r_i. Note that the sequence numbers of tones in each set can be also randomly assigned to further enhance the security.

Step 2.2: Assign the dependencies with the binary coding of j to $w_{i,j}$; consider each binary coding to be a column vector $D_j = (d_{j,1}, d_{j,2}, \ldots, d_{j,z})^T$, where

$$j = \sum_{k=1}^{z} d_{j,k} \times 2^{(k-1)}. \tag{3}$$

The encoding matrix $\mathbf{D}$ consists of all these vectors, *i.e.*,

$$\mathbf{D} = (D_1, D_2, \ldots, D_{y'}) = \begin{bmatrix} d_{1,1} & d_{2,1} & \cdots & d_{y',1} \\ d_{1,2} & d_{2,2} & \cdots & d_{y',2} \\ \vdots & \vdots & \cdots & \vdots \\ d_{1,z} & d_{2,z} & \cdots & d_{y',z} \end{bmatrix} \tag{4}$$

Step 2.3: For each row in $\mathbf{D}$, calculate

$$x_{i,k} = \begin{cases} 0 & m_{i,k} = \oplus_{j=1}^{y'}(w_{i,j} \times d_{j,k}) \\ 0 & m_{i,k} \neq \oplus_{j=1}^{y'}(w_{i,j} \times d_{j,k}) \end{cases}, 1 \leq k \leq z, \tag{5}$$

where $\oplus_{j=1}^{y'}$ represents continuous XOR operations.

Step 2.4: Calculate the following expression:

$$X_i = \sum_{k=1}^{z} x_{i,k} \times 2^{k-1}. \tag{6}$$

If $X_i = 0$, there are no bits needed to be changed in W_i, which means the cover part $\mathcal{B}_i$ remains unchanged; otherwise, the X_i-th tone in $\mathcal{B}_i$ needs to be modulated. Specifically, the steganographic tone of the X_i-th tone (denoted by $\mathfrak{b}_{i,X_i}^*$) is an element adjacent to $\mathfrak{b}_{i,X_i}$ in the corresponding set of all possible note combinations. If there are two candidate elements for $\mathfrak{b}_{i,X_i}^*$, the sender can randomly choose one to substitute $\mathfrak{b}_{i,X_i}$. Repeat the above operation for each tone in $\mathcal{B}_i$ until all the secret message are embedded.

Note that, to achieve successful covert communication, both the communication parties should agree on the key, the embedding rate, the adopted MES and the length of secret messages to be embedded. In this paper, we assume that the sender can distribute the parameters to the receiver in a secure manner.

Table 1: Note combinations and their sequence numbers for various values of t and r_i

r_i	t	Tones (No.: Note Combination)
3	1	$1:T_1=\{c_{i,1}\};2:T_2=\{c_{i,2}\};3:T_3=\{c_{i,3}\}$.
	2	$1:T_1=\{c_{i,1},c_{i,2}\};2:T_2=\{c_{i,2},c_{i,3}\};3:T_3=\{c_{i,1},c_{i,3}\}$.
4	1	$1:T_1=\{c_{i,1}\};2:T_2=\{c_{i,2}\};3:T_3=\{c_{i,3}\};4:T_4=\{c_{i,4}\}$.
	2	$1:T_1=\{c_{i,1},c_{i,2}\};2:T_2=\{c_{i,2},c_{i,3}\};3:T_3=\{c_{i,3},c_{i,4}\};4:T_4=\{c_{i,1},c_{i,3}\};$ $5:T_5=\{c_{i,1},c_{i,4}\};6:T_6=\{c_{i,2},c_{i,4}\}$.
	3	$1:T_1=\{c_{i,1},c_{i,2},c_{i,3}\};2:T_2=\{c_{i,2},c_{i,3},c_{i,4}\};3:T_3=\{c_{i,1},c_{i,2},c_{i,4}\};$ $4:T_4=\{c_{i,1},c_{i,3},c_{i,4}\};$
5	1	$1:T_1=\{c_{i,1}\};2:T_2=\{c_{i,2}\};3:T_3=\{c_{i,3}\};4:T_4=\{c_{i,4}\};5:T_5=\{c_{i,5}\}$.
	2	$1:T_1=\{c_{i,1},c_{i,2}\};2:T_2=\{c_{i,2},c_{i,3}\};3:T_3=\{c_{i,3},c_{i,4}\};4:T_4=\{c_{i,4},c_{i,5}\};$ $5:T_5=\{c_{i,1},c_{i,3}\};6:T_6=\{c_{i,1},c_{i,4}\};7:T_7=\{c_{i,1},c_{i,5}\};8:T_5=\{c_{i,2},c_{i,4}\};$ $9:T_9=\{c_{i,2},c_{i,5}\};10:T_{10}=\{c_{i,3},c_{i,5}\}$.
	3	$1:T_1=\{c_{i,1},c_{i,2},c_{i,3}\};2:T_2=\{c_{i,2},c_{i,3},c_{i,4}\};3:T_3=\{c_{i,3},c_{i,4},c_{i,5}\};$ $4:T_4=\{c_{i,1},c_{i,2},c_{i,4}\};5:T_5=\{c_{i,1},c_{i,2},c_{i,5}\};6:T_6=\{c_{i,1},c_{i,3},c_{i,4}\};$ $7:T_1=\{c_{i,1},c_{i,3},c_{i,5}\};8:T_8=\{c_{i,1},c_{i,4},c_{i,5}\};9:T_9=\{c_{i,2},c_{i,3},c_{i,5}\};$ $10:T_{10}=\{c_{i,2},c_{i,4},c_{i,5}\}$.
	4	$1:T_1=\{c_{i,1},c_{i,2},c_{i,3},c_{i,4}\};2:T_2=\{c_{i,2},c_{i,3},c_{i,4},c_{i,5}\};3:T_3=\{c_{i,1},c_{i,2},c_{i,3},c_{i,5}\};$ $4:T_4=\{c_{i,1},c_{i,2},c_{i,4},c_{i,5}\};5:T_5=\{c_{i,1},c_{i,3},c_{i,4},c_{i,5}\}$.
6	1	$1:T_1=\{c_{i,1}\};2:T_2=\{c_{i,2}\};3:T_3=\{c_{i,3}\};4:T_4=\{c_{i,4}\};5:T_5=\{c_{i,5}\};$ $6:T_6=\{c_{i,6}\}$.
	2	$1:T_1=\{c_{i,1},c_{i,2}\};2:T_2=\{c_{i,2},c_{i,3}\};3:T_3=\{c_{i,3},c_{i,4}\};4:T_4=\{c_{i,4},c_{i,5}\};$ $5:T_5=\{c_{i,5},c_{i,6}\};6:T_6=\{c_{i,1},c_{i,3}\};7:T_7=\{c_{i,1},c_{i,4}\};8:T_5=\{c_{i,1},c_{i,5}\};$ $9:T_9=\{c_{i,1},c_{i,6}\};10:T_{10}=\{c_{i,2},c_{i,4}\};11:T_{11}=\{c_{i,2},c_{i,5}\};12:T_{12}=\{c_{i,2},c_{i,6}\}.$ $13:T_13=\{c_{i,3},c_{i,5}\};14:T_{14}=\{c_{i,3},c_{i,6}\};15:T_{15}=\{c_{i,4},c_{i,6}\}$.
	3	$1:T_1=\{c_{i,1},c_{i,2},c_{i,3}\};2:T_2=\{c_{i,2},c_{i,3},c_{i,4}\};3:T_3=\{c_{i,3},c_{i,4},c_{i,5}\};$ $4:T_4=\{c_{i,4},c_{i,5},c_{i,6}\};5:T_5=\{c_{i,1},c_{i,2},c_{i,4}\};6:T_6=\{c_{i,1},c_{i,2},c_{i,5}\};$ $7:T_1=\{c_{i,1},c_{i,2},c_{i,6}\};8:T_8=\{c_{i,1},c_{i,3},c_{i,4}\};9:T_9=\{c_{i,1},c_{i,3},c_{i,5}\};$ $10:T_{10}=\{c_{i,1},c_{i,3},c_{i,6}\};11:T_{11}=\{c_{i,1},c_{i,4},c_{i,5}\};12:T_{12}=\{c_{i,1},c_{i,4},c_{i,6}\}.$ $13:T_{13}=\{c_{i,1},c_{i,5},c_{i,6}\};14:T_{14}=\{c_{i,2},c_{i,3},c_{i,5}\};15:T_{15}=\{c_{i,2},c_{i,3},c_{i,6}\}.$ $16:T_{16}=\{c_{i,2},c_{i,4},c_{i,5}\};17:T_{17}=\{c_{i,2},c_{i,4},c_{i,6}\};18:T_{18}=\{c_{i,2},c_{i,5},c_{i,6}\}.$ $19:T_{19}=\{c_{i,3},c_{i,4},c_{i,6}\};20:T_{20}=\{c_{i,3},c_{i,5},c_{i,6}\}$.
	4	$1:T_1=\{c_{i,1},c_{i,2},c_{i,3},c_{i,4}\};2:T_2=\{c_{i,2},c_{i,3},c_{i,4},c_{i,5}\};3:T_3=\{c_{i,3},c_{i,4},c_{i,5},c_{i,6}\};$ $4:T_4=\{c_{i,1},c_{i,2},c_{i,3},c_{i,5}\};5:T_5=\{c_{i,1},c_{i,2},c_{i,3},c_{i,6}\};6:T_6=\{c_{i,1},c_{i,2},c_{i,4},c_{i,5}\};$ $7:T_7=\{c_{i,1},c_{i,2},c_{i,4},c_{i,6}\};8:T_8=\{c_{i,1},c_{i,2},c_{i,5},c_{i,6}\};9:T_9=\{c_{i,1},c_{i,3},c_{i,4},c_{i,5}\};$ $10:T_{10}=\{c_{i,1},c_{i,3},c_{i,4},c_{i,6}\};11:T_{11}=\{c_{i,1},c_{i,3},c_{i,5},c_{i,6}\};$ $12:T_{12}=\{c_{i,1},c_{i,4},c_{i,5},c_{i,6}\};13:T_{13}=\{c_{i,2},c_{i,3},c_{i,4},c_{i,6}\};$ $14:T_{14}=\{c_{i,2},c_{i,3},c_{i,5},c_{i,6}\};15:T_{15}=\{c_{i,2},c_{i,4},c_{i,5},c_{i,6}\}$.
	5	$1:T_1=\{c_{i,1},c_{i,2},c_{i,3},c_{i,4},c_{i,5}\};2:T_2=\{c_{i,2},c_{i,3},c_{i,4},c_{i,5},c_{i,6}\};$ $3:T_3=\{c_{i,1},c_{i,2},c_{i,3},c_{i,4},c_{i,6}\};4:T_4=\{c_{i,1},c_{i,2},c_{i,3},c_{i,5},c_{i,6}\};$ $5:T_5=\{c_{i,1},c_{i,3},c_{i,4},c_{i,5},c_{i,6}\}$.

To illustrate the above embedding process, we give a concrete example, as shown in Figure 2. The set of tone parts is $A=\{T_1,\ T_2,\ \cdots,\ T_{15}\}=\{\mathrm{d}^1,\ \{a^1,\ c^2,\ f^2\},\ a,\ \{a^1,\ c^2\},\ \{a^1,\ c^2\},\ g,\ \{d^1,\ g^1,\ b^1\},\ g,\ \{d^1,\ g^1\},\ \{d^1,\ g^1\},\ c^1,\ \{e^1,\ g^1,\ b^1\},\ g,\ \{g^1,\ b^1\},\ \{g^1,\ b^1\}\}$, the chord progression is $\wedge=\{C_1,\ C_2,\ C_3\}$, where $C_1=\{\mathrm{d}^1,\mathrm{a}^1,\mathrm{c}^2,\mathrm{f}^2,\mathrm{a}^2\}$, $C_2=\{\mathrm{g},\mathrm{d}^1,g^1,\mathrm{b}^1,\mathrm{g}^2\}$ and $C_3=\{\mathrm{c}^1,\mathrm{e}^1,\mathrm{g}^1,\mathrm{b}^1,\mathrm{e}^2\}$(marked by blue notes). Hence, we can obtain the embedding factors for the tones in A is {0, 1, 0, 1, 1, 0, 1, 1, 1, 1, 0, 1, 0, 1, 1}, because T_1, T_6 and T_{11} are respectively the first tone of corresponding chords, $T3$ and T_{13} contain the notes not belonging to the corresponding chords.

Accordingly, the set of all available cover tones $B=\{T_2,T_4,T_5,T_7,T_8,T_9,T_{10},T_{12},T_{14},T_{15}\}$. Assume that MES (3, 2) is adopted, and B is divided into three parts, *i.e.*,$B=\{\mathcal{B}_1,\mathcal{B}_2,\mathcal{B}_3\}$. For the first part $\mathcal{B}_1=\{T_2,T_4,T_5\}=\{\{\mathrm{a}^1,\mathrm{c}^2,\mathrm{f}^2\},\{\mathrm{a}^1,\mathrm{c}^2\},\{\mathrm{a}^1,\mathrm{c}^2\}\}$ (marked by gray area), the set of the tone states $W_1=\{0,0,0\}$. According to Step 2.3, we can get the encoding matrix as

$$\mathbf{D}=\begin{bmatrix}1&0&1\\0&1&1\end{bmatrix}.$$

Assume the current part of the secret message is $\mathcal{M}_1=\{1,0\}$. According to Equation (5), we get

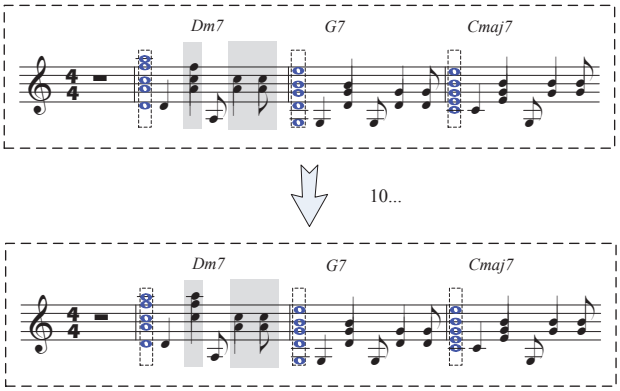


Figure 2: An Example for the Embedding Process of Note-modulating Steganographic Scheme

$x_{1,1} = 1$, $x_{1,2} = 0$. Due to $X_1 = 1 \times 1 + 0 \times 2 = 1$, the first element in W_1 needs to be changed. That is, the first tone in $\mathcal{B}_1$, *i.e.*, T_2, needs to be modulated to an adjacent note combination (*e.g.*, $\{c^2, f^2, a^2\}$) to represent state "1". Therefore, the steganographic version of $\mathcal{B}_1$ is $\{d^1, \{c^2, f^2, a^2\}, a, \{a^1, c^2\}, \{a^1, c^2\}\}$.

The receiver can restitute the secret message M by the following steps:

Step 1. Extraction: With the shared key K and the embedding rate ξ, the receiver first generates the random binary sequence $S_1 = \{s_{1,j} \mid j = 1, 2, \ldots, N\}$, and calculate the embedding factor λ_j as Equation (1). Accordingly, the set of the tones containing secret information can be obtained, denoted as $B^* = \{b_1^*, b_2^*, \ldots, b_{L_B}^*\}$.

Step 2. Decoding: Divide B^* into U parts with y (denote as $B^* = \{\mathcal{B}_1^*, \mathcal{B}_2^*, \ldots, \mathcal{B}_U^*\}$); for the first y' tones in each tone part $\mathcal{B}_i^*$, the receiver calculates the corresponding tone state $W_i^* = \{w_{i,1}^*, w_{i,2}^*, \ldots, w_{i,y'}^*\}$ according to Equation (2), and then get each bit of embedded message by calculating the following expression:

$$\mathfrak{m}_{i,k} = \oplus_{j=1}^{y'}(w_{i,j}^* \times z_{j,k}),\ 1 \le k \le z. \tag{7}$$

Combining all the extracted bits, the receiver can obtain the whole secret message M.

Let the possibility of $\alpha_j = 0$ be P_α, the possibility of $\beta_j = 0$ be P_β, the possibility of $\gamma_j = 0$ be P_γ. Then, the possibility of $\lambda_j = 1$ can be determined as

$$P(\lambda_j = 1) = (1 - P_\alpha - P_\beta - P_\gamma) \times \xi, \tag{8}$$

where $P_\alpha = L_C \,/\, N$; $P_\beta = N_\beta \,/\, N$, N_β is the number of tones whose notes are equal to the number of the notes in the corresponding chords; $P_\gamma = N_\gamma \,/\, N$, N_γ is the number of tones which contain the notes not belonging to the corresponding chords. Note that the three conditions for unavailable tones are judged one by one, so the three sets of unavailable tones have no common elements. Therefore, the capacity of the proposed scheme (denoted by ω) can be determined as

$$\begin{aligned} \omega &= \lfloor \frac{N \times (1 - P_\alpha - P_\beta - P_\gamma) \times \xi}{y} \rfloor \times z \\ &= \lfloor \frac{(N - L_C - N_\beta - N_\gamma) \times \xi}{y} \rfloor \times z. \end{aligned} \tag{9}$$

3 Performance Evaluation

To evaluate the performance of the proposed scheme, we collect thirty pieces of guitar accompanies from the Internet, including three types of music styles, namely, BossaNova, Folk and Popular. For each sample, we perform four kinds of steganographic experiments at the embedding rate of 100%, namely, the steganography with no MES, the ones with MES (3, 2), the one with MES (7, 3) and the one with MES (15, 4). The secret message produced randomly can be successfully embedded and extracted in any case. Table 2 shows the embedding capacities of all the music samples in various steganographic cases. Figure 3 further shows the average embedding capacities per measure for all the music samples in the four steganographic modes, respectively. From them, we can learn the following facts. First, the embedding capacities of all the music samples are identical with the ones calculated as Equation (9), indicating the proposed scheme is feasible and correct. Second, the thirty music samples render different embedding capacity even in the same steganographic mode. The reason for the difference on their capacities is that there are different numbers of tones in each measure. In other words, the average capacity per measure is proportionate to the number of tones in each measure. Therefore, we can choose the samples containing as many tones as possible in each measure for hiding information. For example, the twelfth, nineteenth, twenty-fifth and twenty-ninth music samples are much better than the others in term of embedding capacity.

In addition, we introduce dissonance value [18, 22] and harmony entropy [22], which are popularly applied in the objective evaluation of music, to evaluate the steganographic transparency in terms of the sensory consonance.

Plomp and Levelt [18, 22] pointed out that the dissonance of two single tones can be parameterized by a model as

$$d(x) = e^{-b_1 x} - e^{-b_2 x}, \tag{10}$$

where x denotes the absolute value of the difference in frequency between two single sinusoids, $b_1 = 3.5$ and $b_2 = 5.75$.

According to this definition, the inherent dissonance of the complex tone $F = \{f_i \mid i = 1, 2, \ldots, n\}$ can be calculated as the sum of the dissonances of all pairs of partials, namely,

$$D_F = \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} d(f_i, f_j). \tag{11}$$

Table 2: The Statistical result of the ABX tests for different music styles

No.	Music Samples	N	L_C	N_β	N_γ	None	MES(3,2)	Capacity(bit) MES (7,3)	MES(15,4)
1	Autumn leaves[1]	326	66	0	73	187	124	78	48
2	Blue bossa[1]	337	79	0	80	178	118	75	44
3	Cherie sweet honey[1]	416	72	0	82	262	174	111	68
4	Fly me to the moon[1]	303	56	0	65	182	120	78	48
5	Happy birthday[1]	343	58	0	42	243	162	102	64
6	Moon represent my heart[1]	487	82	0	84	321	214	135	84
7	Night swing[1]	278	65	17	110	86	56	36	20
8	Skimming over the surface[1]	433	73	0	108	252	168	108	64
9	The warmest love song[1]	625	105	0	13	507	338	216	132
10	Write a song[1]	553	97	0	106	350	232	150	92
11	Auld lang syne[2]	265	67	0	0	198	132	84	52
12	Childhood memory[2]	545	69	0	0	476	316	204	124
13	Forest birch[2]	607	76	0	277	254	168	108	64
14	Grandma's penghu bay[2]	373	63	0	0	310	286	132	80
15	Katyusha[2]	377	95	0	0	282	188	120	72
16	Lilac[2]	643	81	0	280	282	188	120	72
17	Orchid[2]	229	58	0	0	171	114	72	44
18	Red river valley[2]	261	66	0	65	130	86	54	32
19	Seasons song[2]	289	37	0	0	252	168	108	64
20	Snail and oriole bird[2]	287	54	0	0	233	154	99	60
21	Baby[3]	440	80	0	59	301	200	129	80
22	Can't help falling in love[3]	501	129	0	103	269	178	114	68
23	Crescent moon[3]	558	84	0	172	302	200	129	80
24	I miss you[3]	407	119	1	112	175	116	75	44
25	June rain[3]	516	68	0	0	448	298	192	116
26	Endless story love[3]	472	100	0	2	370	246	156	96
27	Rainbow[3]	566	77	0	243	246	164	105	64
28	Starry mood[3]	524	78	0	150	296	196	126	76
29	T1213121[3]	305	39	0	8	258	172	108	68
30	Wonderful power song[3]	512	104	0	0	408	272	174	108

Note: [1] belongs to the style of BossaNova, [2] belongs to the style of Folk, [3] belongs to the style of Popular.

Further, the dissonance of F at an interval α can be calculated as follows.

$$D_F(\alpha) = D_F + D_{\alpha F} + \sum_{i=1}^{n}\sum_{j=1}^{n} d(f_i, \alpha f_j), \tag{12}$$

where $\alpha F = \{\alpha f_i \mid i = 1, 2, \ldots, n\}$ represents the note of F at an interval α.

Accordingly, the dissonance of a chord of three notes at the intervals 1, a and b can be calculated by adding the dissonances between all partials, namely,

$$D_F(a, b) = D_F(a) + D_F(b) + D_{aF}(b \ / \ a). \tag{13}$$

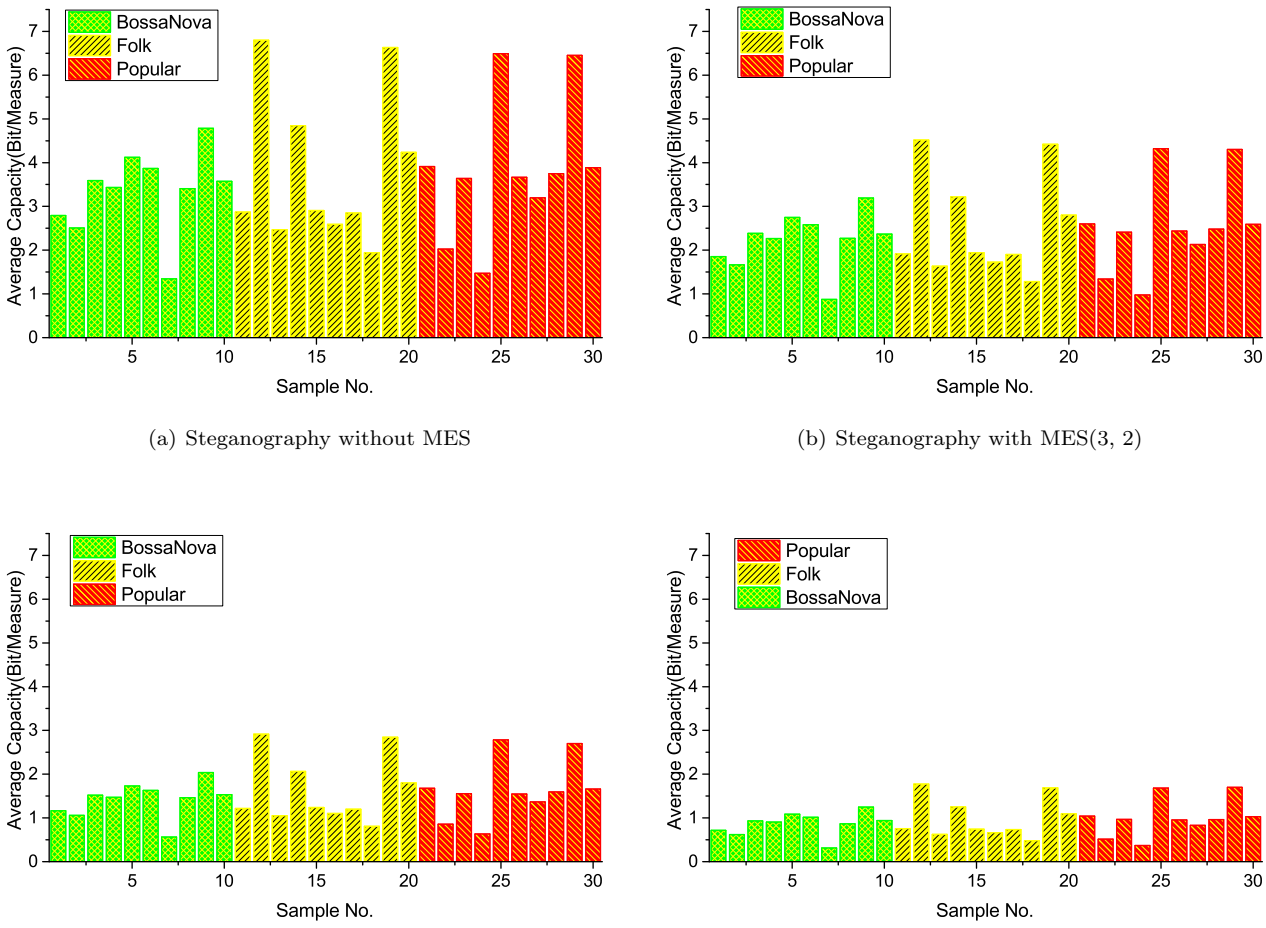
Harmonic entropy is a measure of the uncertainty in pitch perception [23]. When two musical notes are played simultaneously at an interval α, they must have a simple-integer ratio (denoted as f_α) about the frequency, which can be modeled with a Farey series $\mathcal{F}_m$ of an order m. For any interval α, the probability that α is perceived as a mistuning of the j-th member of the Farey series is

$$p_j(\alpha) = \frac{1}{\sigma\sqrt{2\pi}} \int_{t \in r_j} e^{-(t-i)^2/2\sigma^2} dt, \tag{14}$$

where $\sigma = 0.007$, r_j represents the region over which f_j dominates, going from the median below to the median above. Assume that the j-th member of the Farey series is $f_j = c_j \ / \ d_j$, and $r_j \in [(c_{j-1} + c_j) \ / \ (d_{j-1} + d_j), (c_j + c_{j+1}) \ / \ (d_j + d_{j+1})]$. Then the harmonic entropy of α can be defined as

$$HE(\alpha) = -\sum_j p_j(\alpha) \log(p_j(\alpha)). \tag{15}$$

To evaluate the steganographic transparency of the proposed scheme, we randomly choose six music samples and their steganographic versions in different modes to make statistics on the dissonance values and harmony entropy. Figures 4 and 5 show the experimental results of the dissonance values and harmony entropy, respectively. From the charts, we can learn the following fact. First, the



(a) Steganography without MES

(b) Steganography with MES(3, 2)

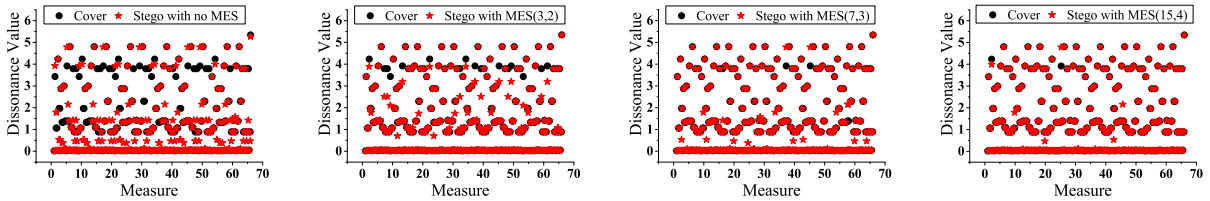
(c) Steganography with MES (7, 3)

(d) Steganography with MES (15, 4)

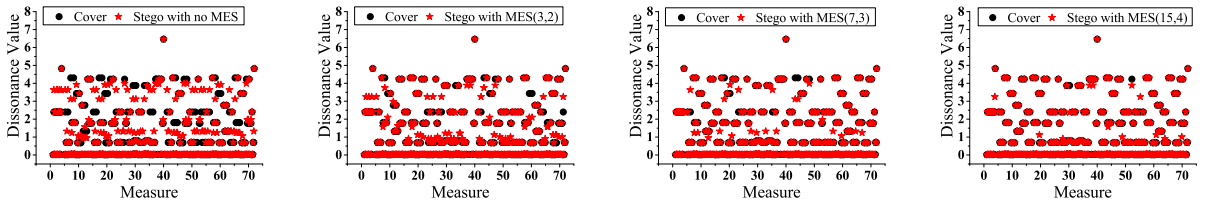
Figure 3: Average steganographic capacity per measure for all the music samples

dissonance values and harmony entropy of some measures are diverse, although all of them are in the normal range. Particularly in the first style, the distributions have a relatively large range, which is caused by various chord types. For example, the major chord, which is a type of consonant chords in music theory, contains a major third and a perfect fifth above its root note. The major seventh chord, which is a dissonant chord, contains a major third, a perfect fifth, and a major seventh. Therefore, the ranges of both dissonance values and harmony entropy for tones in different chords are also different. Second, the dissonance value and harmony entropy of each measure in the steganographic accompaniments for a given music sample, are identical or highly similar to those in the original accompaniments, indicating that the steganographic samples can be played at a good level of sensory consonance. That is, the proposed scheme can achieve good steganographic transparency in term of the sensory consonance. Particularly, as the length of cover part is increased, the embedding distortions are accordingly reduced, namely, better steganographic transparency can be achieved. In this sense, the proposed scheme can achieve a good balance between steganographic transparency and capacity by introducing an appropriate MES.

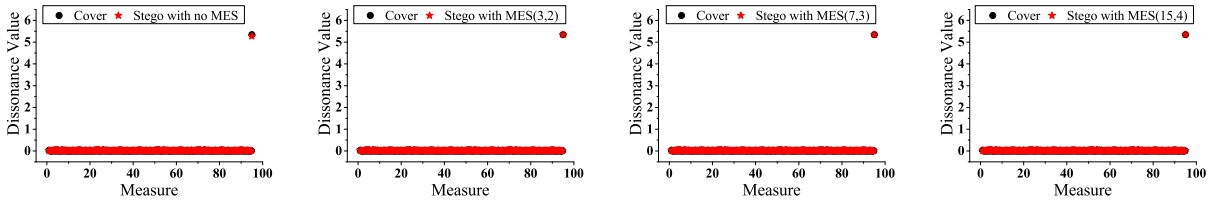
In addition, we also conduct ABX tests to further evaluate the steganographic transparency of the proposed scheme. We create a sample set as X by randomly selecting two original accompaniments and eight steganographic ones (two samples without MES, two samples with MES (3, 2), two samples with MES (7, 3), and two samples with MES (15, 4)) for each style, and invite thirty persons (including ten professionals in the information hiding field, ten guitar lovers and ten common participators) to identify the categories of all the music samples in X independently. Specifically, if a sample is identified as an original one, it is labeled as A; otherwise, it is labeled as B. Table 3 and Table 4 show the statistical results of the ABX tests for different music styles and for different steganographic modes. It is not hard to find out that for any given test set, the participants even the guitar lovers who are familiar with these music styles, cannot accurately distinguish between the original and stegano-



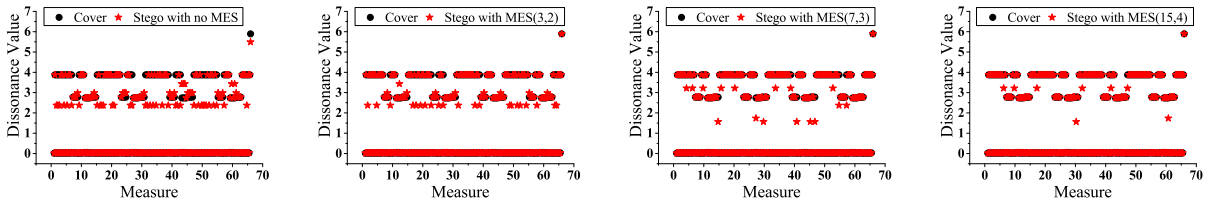
(a) Autumn Leaves



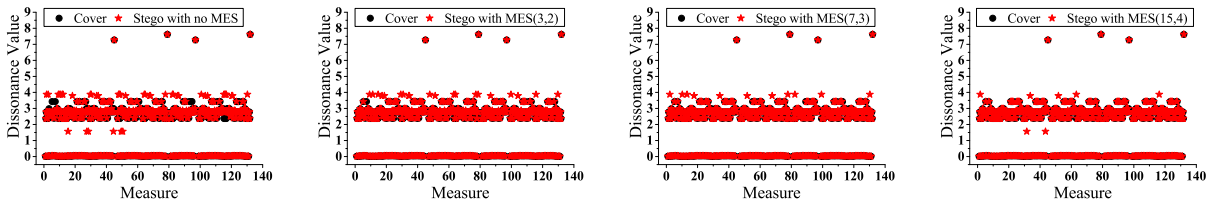
(b) Cherie Sweet Honey



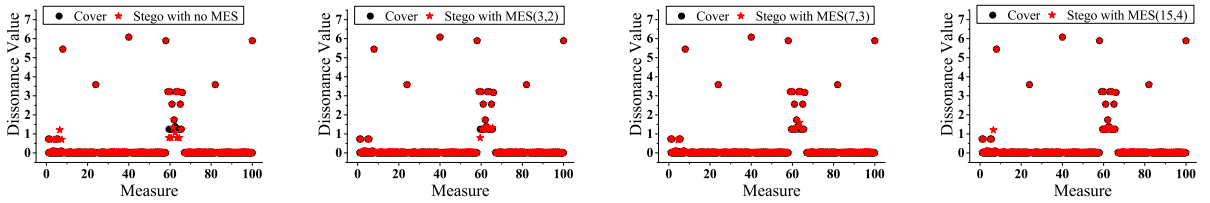
(c) Katyusha



(d) Red River Valley

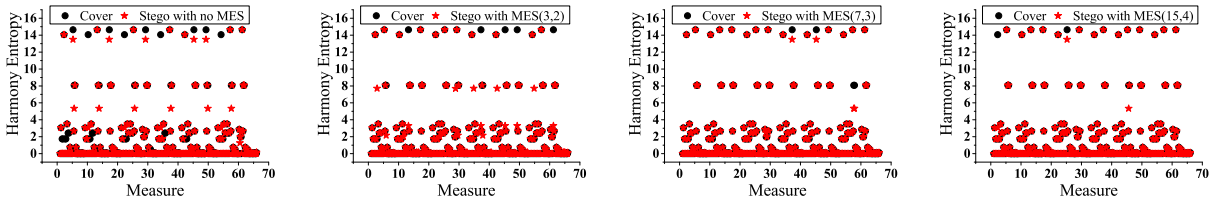


(e) Can't Help Falling In Love

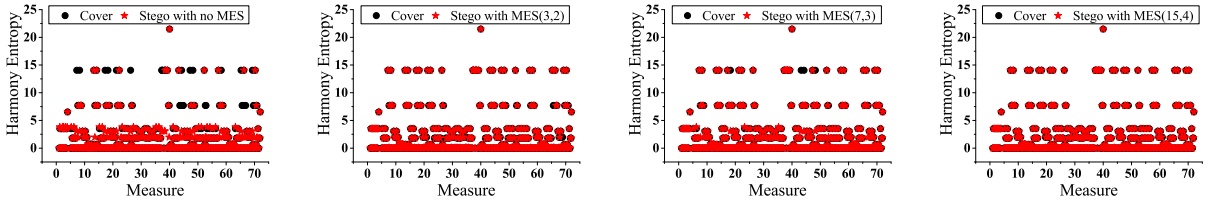


(f) Endless story love

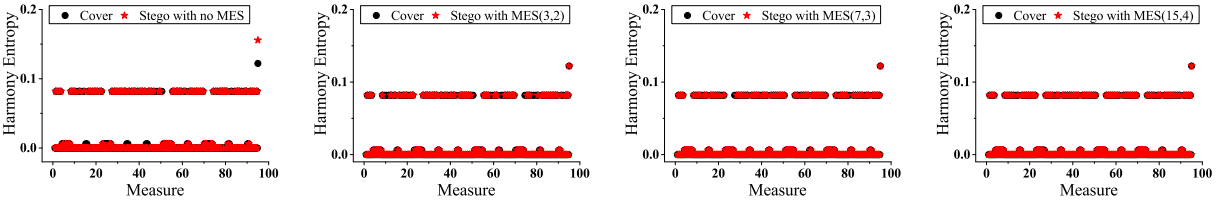
Figure 4: The statistical results for dissonance values



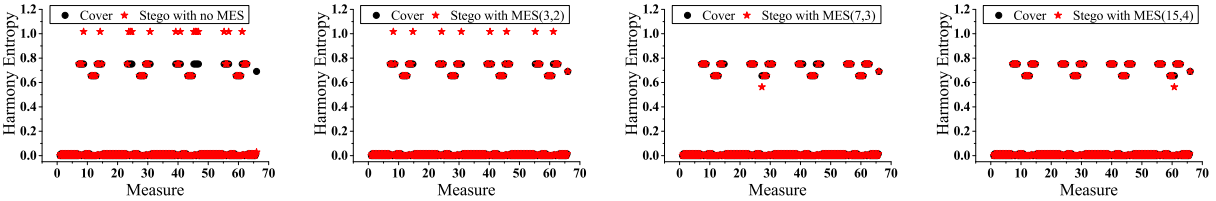
(a) Autumn Leaves



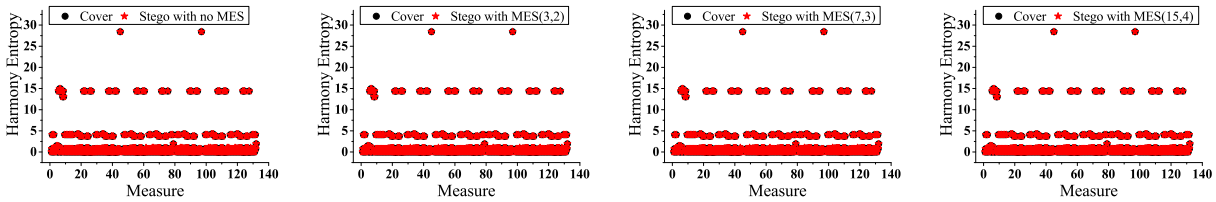
(b) Cherie Sweet Honey



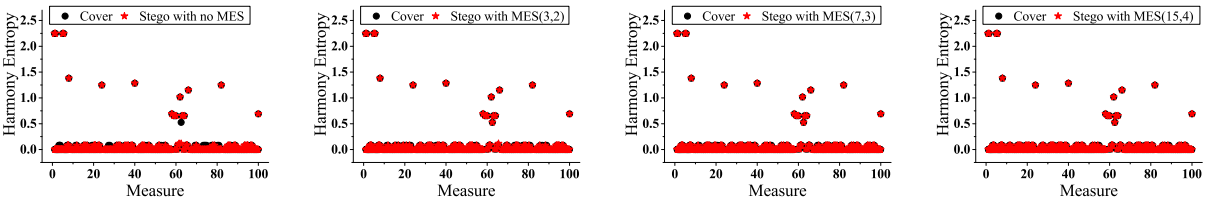
(c) Katyusha



(d) Red River Valley



(e) Can't Help Falling In Love



(f) Endless story love

Figure 5: The statistical results for harmony entropy

Table 3: The statistical result of the ABX tests for different music styles

	Bossa Nova	Folk	Popular
Guitar Lovers	46%	53%	47%
Professionals	42%	48%	61%
Participators	49%	51%	53%

Table 4: The statistical result of the ABX tests for different steganographic modes

	With no MES	With MES (3, 2)	With MES (7, 3)	With MES (15, 4)
Guitar Lovers	60%	53%	53%	52%
Professionals	57%	50%	51%	53%
Participators	48%	49%	46%	53%

graphic samples, which demonstrates again that the proposed scheme can achieve excellent steganographic transparency. Particularly, for the guitar lovers, as the length of cover part is increased, their accuracy for distinguishing the original and steganographic samples is further decreased, which demonstrates again that MES can further improve the steganographic transparency. Moreover, the results also suggest again that the proposed scheme can achieve a good balance between steganographic transparency and capacity by choosing a proper MES.

4 Conclusions

Steganography, which can conceal secret messages into seemingly normal carriers without any perceptible change, provides an efficient means for secure communication. So far, extensive researches on steganography have been carried out, and steganographic covers have been also extended from initial images to almost all multimedia. Due to its diversity, sensual and physical redundancies, music is considered as a type of ideal carrier for steganography, and has attracted increasing attention from the research community of information hiding. In this paper, we present a novel note-modulating steganographic scheme for guitar music. Differing from the existing works, the proposed scheme aims to embed secret messages into guitar accompaniments based upon the fact that there are many note combinations for expressing a group of similar harmony effects. In other words, the proposed scheme conceals the secret messages by suitably modulating the note combinations for the corresponding candidate tones. Additionally, we introduce the Hamming matrix encoding strategy to further reduce the embedding distortions. To our best knowledge, this is the first steganographic scheme using the Guitar accompaniments. We evaluate the proposed scheme with a large number of guitar-music samples. The experimental results demonstrate that the proposed scheme is indeed feasible and efficient. In particular, by introducing an appropriate matrix embedding strategy, the proposed scheme can achieve a good balance between steganographic transparency and capacity.

Acknowledgments

This work was supported in part by National Natural Science Foundation of China under Grant Nos. 61972168, U1536115 and U1405254, Natural Science Foundation of Fujian Province of China under Grant No. 2018J01093, Program for New Century Excellent Talents in Fujian Province University under Grant No. MJK2016-23, Program for Outstanding Youth Scientific and Technological Talents in Fujian Province University under Grant No. MJK2015-54, Promotion Program for Young and Middle-aged Teacher in Science & Technology Research of Huaqiao University under Grant No. ZQN-PY115 and Program for Science & Technology Innovation Teams and Leading Talents of Huaqiao University under Grant No.2014KJTD13.

References

[1] A. Adli and Z. Nakao, "Three steganography algorithms for midi files," in *Proceedings of the International Conference on Machine Learning and Cybernetics*, pp. 2401–2404, Aug. 2005.

[2] Y. Cao, Z. Zhou, X. Sun, and C. Gao, "Coverless information hiding based on the molecular structure images of material," *Computers, Materials & Continua*, vol. 54, no. 2, pp. 197–207, 2018.

[3] A. Cheddad, J. Condell, K. Curran, and P. M. Kevitt, "Digital image steganography: Survey and analysis of current methods," *Signal processing*, vol. 90, no. 3, pp. 727–752, 2010.

[4] F. Djebbar, B. Ayad, K. A. Meraim, and H. Hamam, "Comparative study of digital audio steganography techniques," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2012, no. 1, p. 25, 2012.

[5] W. Funk and M. Schmucker, "High capacity information hiding in music scores," in *Proceedings of the*

First International Conference on Web Delivering of Music, pp. 12–19, Nov. 2001.

[6] L. C. Huang, T. H. Feng, and M. S. Hwang, "A new lossless embedding techniques based on HDWT," *IETE Technical Review*, vol. 34, no. 1, pp. 40–47, 2017.

[7] P. Hunt. "J. S. bach and steganography," *Electrum Magazine*, 2013. (`http://www.electrummagazine.com/2013/12/j-s-bach-and-steganography/`)

[8] L. Hutchinson, "Live musical steganography," *Scholar Commons*, 2014. (`https://scholarcommons.sc.edu/senior_theses/20/`)

[9] B. Jana, "Dual image based reversible data hiding scheme using weighted matrix," *International Journal of Electronics and Information Engineering*, vol. 5, no. 1, pp. 6–19, 2016.

[10] J. Li, Q. Lin, C. Yu, X. Ren, and P. Li, "A QDCT- and svd-based color image watermarking scheme using an optimized encrypted binary computer-generated hologram," *Soft Computing*, vol. 22, no. 1, pp. 47–65, 2018.

[11] J. Li, X. Huang, J. Li, X. Chen, and Y. Xiang, "Securely outsourcing attribute-based encryption with checkability," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 8, pp. 2201–2210, 2014.

[12] Y. Liu, X. Sun, C. Gan, and H. Wang, "An efficient linguistic steganography for chinese text," in *Proceedings of the IEEE International Conference on Multimedia and Expo*, pp. 2094–2097, July 2007.

[13] J. Lubacz, W. Mazurczyk, and K. Szczypiorski, "Principles and overview of network steganography," *IEEE Communications Magazine*, vol. 52, no. 5, pp. 225–229, 2014.

[14] Y. B. Luo, Y. F. Huang, F.F. Li, and C. C. Chang, "Text steganography based on ci-poetry generation using markov chain model," *KSII Transactions on Internet and Information Systems*, vol. 10, no. 9, pp. 4568–4584, 2016.

[15] W. Mazurczyk, "Voip steganography and its detection—a survey," *ACM Computing Surveys (CSUR'13)*, vol. 46, no. 2, p. 20, 2013.

[16] R. J. Mstafa, K. M. Elleithy, and E. Abdelfattah, "A robust and secure video steganography method in dwt-dct domains based on multiple object tracking and ECC," *IEEE Access*, vol. 5, pp. 5354–5365, 2017.

[17] F. A. P. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Information hiding-a survey," *Proceedings of the IEEE*, vol. 87, no. 7, pp. 1062–1078, 1999.

[18] R. Plomp and W. J. M. Levelt, "Tonal consonance and critical bandwidth," *The Journal of the Acoustical Society of America*, vol. 38, no. 4, pp. 548–560, 1965.

[19] N. Provos and P. Honeyman, "Hide and seek: An introduction to steganography," *IEEE security & privacy*, vol. 99, no. 3, pp. 32–44, 2003.

[20] Z. Qian and X. Zhang, "Reversible data hiding in encrypted images with distributed source encoding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 4, pp. 636–646, 2016.

[21] S. Rajendran and M. Doraipandian, "Chaotic map based random image steganography using lsb technique.," *International Journal Network Security*, vol. 19, no. 4, pp. 593–598, 2017.

[22] W. A. Sethares, "Consonance and dissonance of harmonic sounds," *Tuning, Timbre, Spectrum, Scale*, pp. 77–95, 2005.

[23] J. Shen, Z. Gui, S. Ji, J. Shen, H. Tan, and Y. Tang, "Cloud-aided lightweight certificateless authentication protocol with anonymity for wireless body area networks," *Journal of Network and Computer Applications*, vol. 106, pp. 117–123, 2018.

[24] M. Shobana, "Efficient x-box mapping in stego-image using four-bit concatenation," *International Journal of Electronics and Information Engineering*, vol. 1, no. 1, pp. 29–33, 2014.

[25] K. Szczypiorski, "Stegibiza: New method for information hiding in club music," in *Proceedings of the 2016 2nd International Conference on Frontiers of Signal Processing (ICFSP'16)*, pp. 20–24, Oct. 2016.

[26] H. Tian, H. Jiang, K. Zhou, and D. Feng, "Transparency-orientated encoding strategies for voice-over-IP steganography," *The Computer Journal*, vol. 55, no. 6, pp. 702–716, 2012.

[27] H. Tian, J. Qin, Y. Huang, Y. Chen, T. Wang, J. Liu, and Y. Cai, "Optimal matrix embedding for voice-over-IP steganography," *Signal Processing*, vol. 117, pp. 33–43, 2015.

[28] Y. L. Wang, J. J. Shen, and M. S. Hwang, "An improved dual image-based reversible hiding technique using lsb matching.," *International Journal Network Security*, vol. 19, no. 5, pp. 858–862, 2017.

[29] Y. G. Wang, G. Zhu, and Y. Q. Shi, "Transportation spherical watermarking," *IEEE Transactions on Image Processing*, vol. 27, no. 4, pp. 2063–2077, 2018.

[30] A. Westfeld, "F5—A steganographic algorithm," in *Proceedings of the fourth International Workshop on Information Hiding*, pp. 289–302, 2001.

[31] C. L. Wu, C. H. Liu, and C. K. Ting, "A novel genetic algorithm considering measures and phrases for generating melody," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC'14)*, pp. 2101–2107, July 2014.

[32] N. I. Wu and M. S. Hwang, "Development of a data hiding scheme based on combination theory for lowering the visual noise in binary images," *Displays*, vol. 49, pp. 116–123, 2017.

[33] N. I. Wu and M. S. Hwang, "A novel LSB data hiding scheme with the lowest distortion," *The Imaging Science Journal*, vol. 65, no. 6, pp. 371–378, 2017.

[34] Z. Wu, B. Liang, L. You, Z. Jian, and J. Li, "High-dimension space projection-based biometric encryption for fingerprint with fuzzy minutia," *Soft Computing*, vol. 20, no. 12, pp. 4907–4918, 2016.

[35] Kotaro Yamamoto and Munetoshi Iwakiri, "A standard midi file steganography based on fluctuation of duration," in *Proceedings of the 2009 International Conference on Availability, Reliability and Security*, pp. 774–77, Mar. 2009.

[36] L. Yang, Z. Han, Z. Huang, and J. Ma, "A remotely keyed file encryption scheme under mobile cloud computing," *Journal of Network and Computer Applications*, vol. 106, pp. 90–99, 2018.

[37] X. Zhang, Y. A. Tan, C. Liang, Y. Li, and J. Li, "A covert channel over volte via adjusting silence periods," *IEEE Access*, vol. 6, pp. 9292–9302, 2018.

[38] Y. Zhang, M. Zhang, X. Yang, D. Guo, and L. Liu, "Novel video steganography algorithm based on secret sharing and error-correcting code for H. 264/avc," *Tsinghua Science and Technology*, vol. 22, no. 2, pp. 198–209, 2017.

[39] E. Zielińska, W. Mazurczyk, and Krzysztof Szczypiorski, "Trends in steganography," *Communications of the ACM*, vol. 57, no. 3, pp. 86–95, 2014.

Biography

Hui Tian received the PhD degree in 2010 in computer science from Huazhong University of Science and Technology, Wuhan, China. He is now a professor and associate dean of the college of computer science and technology, National Huaqiao University, Xiamen, China. His present research interests include network & information security, steganography and steganalysis, digital forensics, and cloud computing security. He has published more than 80 papers in refereed proceedings of conferences, journals and books, and got five patents. He is a senior member of IEEE, a senior member of China Computer Federation (CCF), a member of the Technical Committee on Internet of CCF and a member of the Technical Committee on Information Storage of CCF.

Zhaohua Zhu received the B.Sc degree in computer science and technology in 2015 from National Huaqiao University, Xiamen, China. He is now pursuing the M. Sc. degree in computer science from National Huaqiao University, Xiamen, China. His interests are in the areas of information hiding, with current focus on steganography based on music.

Chin-Chen Chang received both the B.Sc. degree in Applied Mathematics in 1977 and the M.Sc. degree in Computer and Decision Sciences in 1979 from National Tsinghua University, Hsinchu, Taiwan, and the Ph.D. degree in computer engineering in 1982 from National Chiao Tung University, Hsinchu, Taiwan. His current title is Chair Professor in department of information engineering and computer science, Feng Chia University, from Feb. 2005. Prior to joining Feng Chia University, he was an associate professor in Chiao Tung University, professor in National Chung Hsing University, chair professor in National Chung Cheng University. His present research interests include computer cryptography and information security, cloud computing, data engineering and database systems. He has over 850 publications in major journals and international Conferences in these areas. Since his early years of career development, he consecutively won Outstanding Youth Award of Taiwan, Outstanding Talent in Information Sciences of Taiwan, AceR Dragon Award of the Ten Most Outstanding Talents, Outstanding Scholar Award of Taiwan, Outstanding Engineering Professor Award of Taiwan, Chung-Shan Academic Publication Awards, Distinguished Research Awards of National Science Council of Taiwan, Top Fifteen Scholars in Systems and Software Engineering of the Journal of Systems and Software, etc. He is currently a Fellow of IEEE and a Fellow of IEE, UK.

Tian Wang received his PhD degree in computer science in City University of Hong Kong in 2011. Currently, he is a professor in the college of computer science and technology, National Huaqiao University, Xiamen, China. His research interests include wireless sensor networks, fog computing and mobile computing. He has published more than 100 papers in refereed proceedings of conferences and journals. He is a member of IEEE.

Yonghong Chen received the PhD degree in 2005 in automation control engineering from Chongqing University, Chongqing, China. He is now a professor in the college of computer science and technology, National Huaqiao University, Xiamen, China. His present research interests include network and multimedia information security, information hiding and watermarking. He has published over 60 papers in refereed proceedings of conferences and journals.

Yiqiao Cai received the Ph.D. degree in computer science in 2012 from Sun Yat-sen University, Guangzhou, China. He is now an associate professor in the college of computer science and technology, National Huaqiao University, Xiamen, China. His present research interests include differential evolution, multi-objective optimiza-tion, and other evolutionary computation techni-ques. He has published over 40 papers in refereed proceedings of conferences and journals.

Automatic Verification of Security of Identity Federation Security Protocol Based on SAML2.0 with ProVerif in the Symbolic Model

Jintian Lu[1,2], Xudong He[1], Yitong Yang[1], Dejun Wang[1], and Bo Meng[1]
(Corresponding author: Bo Meng)
School of Computer Science, South-Central University for Nationalities[1]
Wuhan 430074, China
School of Data and Computer Science, Sun Yat-Sen University[2]
Guangzhou 510006, China
(Email: mengscuec@gmail.com)
(Received June 20, 2018; Revised and Accepted Nov. 22, 2018; First Online June 17, 2019)

Abstract

In recent years, several Identity Federation security protocols have been introduced to enhance the security of Identity authentication. Owning to the complexity, assessing security of Identity Federation security protocols has becoming a hot issue. Hence, in this study, we firstly review the development of formal methods on Identity Federation Security Protocol Based on SAML. And then, an Identity Federation Security Protocol Based on SAML is formalized with Applied PI calculus. After that, the formal model is translated into the inputs of ProVerif. Finally, we run ProVerif to analyze the security properties of Identity Federation Security Protocol Based on SAML. The result shows it has not secrecy, but it has some authentications. At the same time, we present a solution to address the security problems.

Keywords: Applied PI Calculus; Authentication; Formal Method; Security Protocol

1 Introduction

Identity Federation has been playing an increasingly important role in information security [2, 9, 26] and can allow the end users to use the same set of credentials to obtain access to multiple resources in different organization. Identity Federation security protocols typically include Microsoft U-Prove, OASIS SAML, and Liberty. But the OASIS Security Assertion Markup Language (SAML) is the emerging standard in this context and it is the most important technology to establish and manage Identify Federation. According to the related researches, the security of Identity Federation based on SAML2.0 has not been analyzed based on rigorous proofs and has been challenged by several analysis.

In order to obtain the strong confidence on security properties of security protocols [2, 8, 10, 15, 18, 25], the symbolic model and the computational model are introduced. Firstly, each model formally defines security properties of security protocol, and then propose methods for strictly proving and analyzing that whether given security protocols meet these requirements in adversarial environments or not. The computational model is too extreme complicated and difficult to get the support of automatic tools. In contrast, the symbolic model is considerably simpler than the computational model, hence proofs are also simpler, and can sometimes benefit from automatic tools support. For example: SMV, NRL, Casper, Isabelle, Athena, Revere, SPIN, Brutus, Coq [4, 7], ProVerif [6], Scyther [22, 24]. ProVerif is an automatic security protocol verifier and accepts the Applied PI calculus [27] as its input. It can process a lot of the different cryptographic primitives and an unbounded number of sessions of the security protocol in an unbounded message space. ProVerif has been tested on security protocols of the literature with very great results.

Therefore, in this paper, we use ProVerif to formally verify security properties of Identity Federation Security Protocol Based on SAML2.0 in the symbolic model.

2 Contribution

Several Identity Federation security protocols have been introduced in the recent years. Owning to the complexity, how to assess its security has become a challenging issue. Formal method is crucial to assess its security. So in this paper, we firstly review the development of the formal methods on Identity Federation Security Protocol Based on SAML 2.0 and apply the automatic tool developed by Blanchet to analyze its security properties. Hence,

firstly, Identity Federation Security Protocol Based on SAML is modeled with the Applied PI calculus. And then the model is translated into the inputs of ProVerif. Finally the translated model is performed by automatic tool ProVerif. The result shows that it has not secrecy of some keys, but it has some authentications based on the model implemented by us. At the same time, we present a solution to address the security problems.

We use the Applied PI calculus to model Identity Federation Security Protocol Based on SAML according to the fact that the Applied PI calculus allows the modeling of relations between data in a simple and precise manner using equational theories over term algebra. The general analysis model is presented in Figure 1.

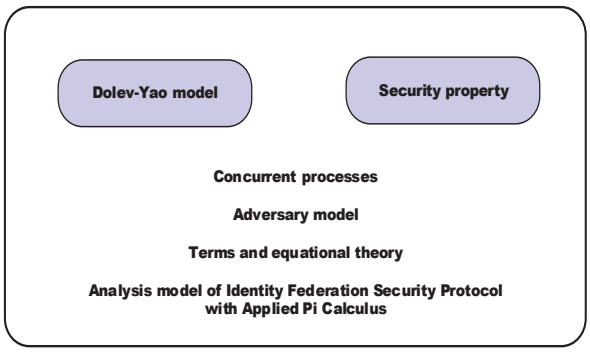


Figure 1: Analysis model of identity federation security protocol based on SAML with the applied PI calculus

There, the security properties model is equivalence between processes, while the attacker is modeled as an arbitrary process running in parallel with the protocol process representing the adversary model, which is the parallel composition of the protocol participants processes. The considered attacker is stronger than the basic Dolev-Yao attacker since it can exploit particular relations between the messages by using particular equational theories stating the message relations. Figure 2 presents the automatic verification of Identity Federation Security Protocol Based on SAML 2.0.

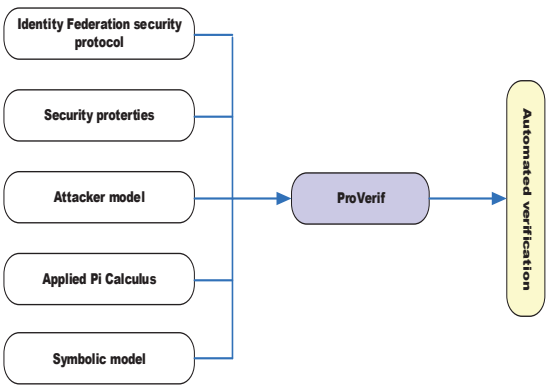


Figure 2: Automatic verification of identity federation security protocol based on SAML

3 Related Work

Here, we present the state-of-art of the analysis of the Identity Federation Security Protocol Based on SAML. Armando *et al.* [1] mechanically analyzed SAML SP-Initiated SSO profile with the model checker SATMC based on HLPSL++ as the specification language and found a severe security vulnerability that allows a dishonest service provider to impersonate a user at another service provider. Cabarcos *et al.* [20] proposed a generic extension for the SAML standard for facilitating the creation of federation relationships in a dynamic way between prior unknown parties. But its security is not proved by formal methods. ter Beek *et al.* [5] used model checker PaMoChSA to analyze the security aspects of the Identity Federation protocol proposed by Telecom Italia and found a man-in-the-middle attack [3, 12, 14, 16, 19].

Ferdous and Poet [17]presented a simple approach based on SAML Profile and allow users to create federations using SAML between two prior unknown organizations in a dynamic fashion. Ghazizadeh *et al.* [11]presented an overview on Identity Federation in the cloud computation environment and pay a special attention on the identity theft issue. Cabarcos *et al.* [21]introduced the IdMRep which is a decentralized reputation-based mechanism which allows trust relationships to be established on-demand driven by user's needs. While they do not analyze its security.

Wang *et al.* [13] presented a browser-based mutual authentication for Federated Identity management to protect the token mutually by binding the client certificate and using TLS protocol. Apart from that they also analyze the security in the Random model and prove that it supports authentication. Saklikar and Saha [23] proposed the VoIP Identity Federation Framework which can make a user to establish Identity Federation and the assertion of any relevant Identity information from one VoIP context to another based on the federate-out and federate-in primitives. While they do not prove its security with formal methods.

4 Applied PI Calculus and ProVerif

The Applied PI calculus is a formal language for describing concurrent processes and their interactions based on Dolev-Yao model. Applied PI calculus is an extension of the PI calculus that inherits the constructs for communication and concurrency from the pure PI calculus. It preserves the constructs for generating statically scoped new names and permits a general systematic development of syntax, operational semantics equivalence and proof techniques. At the same time, there are several powerful automatic tool supported the Applied PI calculus, for example, ProVerif. The Applied PI calculus with ProVerif has been used to study a variety of complicated security protocols.

In the Applied PI calculus, terms consists of names variables and signature $\sum$. $\sum$ is the set of function symbols, each with an arity. Terms and function symbols are sorted, and of course function symbol application must respect sorts and arties. Typically, we let a,b and c range over channel names. Let x,y and zrange over variables, and u over variables and names.over variables and names. We abbreviate an arbitrary sequence of terms $M_1, \cdots, M_i$ to $\widetilde{M}$. In applied PI calculus, it has plain processes and extended processes. Plain processes are built up in a similar way to processes in the PI calculus, except that messages can contain terms and that names need not be just channel names. The process 0 is an empty process. The process Q|p is the parallel composition of P and Q. The replication !p produces an infinite number of copies of P which run in parallel. The process vn.p firstly creates a new, private name then executes as P.The abbreviation $v\widetilde{n}$ is a sequence of name restrictions $vn_1, \cdots, vn_i$. The process in (u, x). P receives a message from channel u,and runs the process P by replacing formal parameter x by the actual message. We use in(u, $\widetilde{M}$). P is the abbreviation for the output of terms $N_1, \cdots, N_i$. The conditional construct if M=N then P else Q runs that if M and N are equal, execute P, otherwise execute Q.

Extended processes add active substitutions and restriction on variables. We write { M / x } for active substitution which replaces the variable x with the term M. The substitution typically appears when the term M has been sent to the environment, but the environment may not have the atomic names that appear in M; The variable x is just a way to refer to M in this situation.

In general an event is used to mark important steps of the security protocol under study but do not otherwise affect its behavior. It can be used to record the context of the sending or receiving message in security protocol. In the applied PI calculus, event event(M) just outputs message M through a special channel. So event event(M) does not reveal M to the adversary. Hence, the execution of the process P after inserting events is the execution of P without events, plus the recording of event(M). The process the event(M). P executes the event(M), then executes P.

ProVerif is an automatic cryptographic protocol verifier based on a representation of the protocol by Horn clauses and the Applied PI calculus. It can handle many different cryptographic primitives, including shared- and public-key cryptography, hash function, and Deffie-Hellman key agreements, specified both as rewrite rules and as equations. It can also deal with an unbounded number of sessions of the protocol and an unbounded message space. When ProVerif cannot prove a property, it can reconstruct an attack, that is, an execution trace of the protocol that falsifies the desired property. ProVerif can prove the following properties: secrecy, authentication and more generally correspondence properties, strong secrecy, equivalences between processes that differ only by terms. ProVerif has been tested on protocols of the literature with very encouraging results. When ProVerif cannot prove a security property, it can reconstruct an attack, ProVerif can prove secrecy, authentication and more generally correspondence properties, strong secrecy, equivalences between processes that differ only by terms.

5 Identity Federation Security Protocol Based on SAML2.0

Identity federation security protocol is mainly made up of three principles: User Agent (UA), Service Provider (SP) and Identity Provider (IdP). Generally there is a Single Sign-On (SSO) service component in the identity provider. SSO allows the end users to provide their credentials once and obtain access to multiple resources. In other words, the identity provider can provide the SSO service. Service provider has the components of access check and assertion consumer service. Hence it has the ability to check and verify the identity of user and assertion consumer. User agent can be browser which is the agent of the users. There are two models in identity federation security protocol based on SAML. One is IdP -initiated model. The other is SP-initiated model. Figure 3 describes identity federation security protocol based on SP-initiated SAML 2.0 using HTTP.

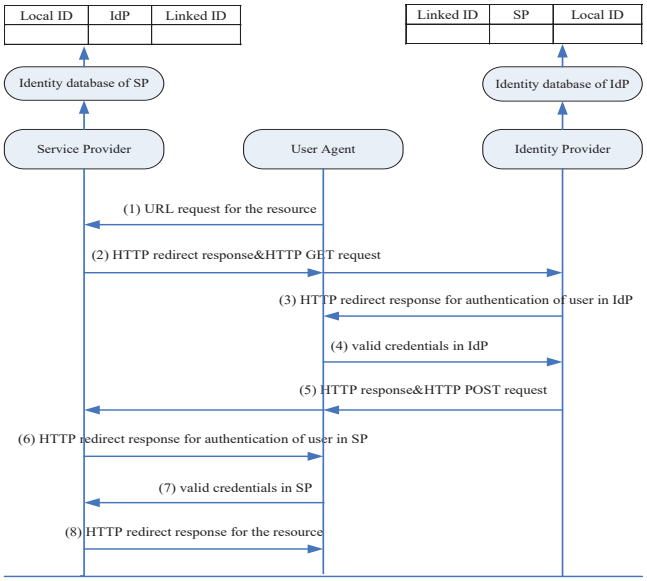


Figure 3: Identity federal security protocol based on SP-initiated SAML 2.0 using HTTP

Apart from that we assume that the service provider has the digital signature public key PU_{sp} and and private key PR_{sp} and the identity provider has two pairs of digital signature public key and private key (PU^1_{idp}, PR^1_{idp}), (PU^2_{idp}, PR^2_{idp}). Service provider and identity provider has an identity database in which it has the information of local ID, SP and linked ID, respectively. The linked ID is the identifier that is used to establish the federation between the local ID in SP identity database and

local ID in IdP database. Apart from that, the identity federation security protocol based on SP-initiated SAML provides the authentication from service provider to user agent and from identity provider to user agent.

The identity federation security protocol based on SP-initiated SAML 2.0 includes eight messages exchanged among service provider, user agent and identity provider.

$$\left\{\begin{array}{l}\text{URLrequest}\\ \text{fortheresource}\end{array}\right\}:=\left\{\begin{array}{l}\text{URLrequest}\\ \text{foraresource}\end{array}\right\} \quad (1)$$

The user agent generates Message (1) which is used to request a target resource that is a secured resource at the service provider and sends it to the service provider.

$$\left\{\begin{array}{l}\text{HTTPredirectresponse}\\ \text{\&HTTPGETrequest}\end{array}\right\}:=\left\{\begin{array}{l}\text{URI}||\text{SAML}-\\ \text{Request}\\ ||\text{RelayState}\end{array}\right\}$$

$$\text{SAMLRequest}:=\left\{\begin{array}{l}\text{ID[Required]}||\\ \text{Version[Required]}\\ ||\text{IssueInstant}\\ \text{[Required]}\\ ||<\text{saml}:\text{Issuer}>\\ \text{[Optional]}\\ ||<\text{cds}:\text{Signature}>\\ \text{[Optional]}\\ ||<\text{NameIDPolicy}>\\ \text{[Optional]}\end{array}\right\} \quad (2)$$

If the parameters of SAMLRequest and RelayState are present in Message (1), the user agent has already been verified by the identity provider and can access the resource in service provider. Here we assume that the parameters of SAMLRequest and RelayState are not included in Message (1). Hence service provider constructs Message (2) and sends it to identity provider by means of service provider. Message (2) mainly consists of URI, SAMLRequest and RelayState parameters. The parameter URI is the address of SSO service component and is generated by service provider. The parameter SAMLRequest is URL-encoded <AuthnRequest> element in SAML and is also generated by service provider. <AuthnRequest> element is used to authenticate the user agent and is mainly composed of ID, Version, IssueInstant, <saml:Issuer> and <cds: Signature> elements. <cds: Signature> element is used to store the digital signature of the <AuthnRequest> element. The digital signature of the <AuthnRequest> element is generated with the private key PR_{sp} of service provider. The parameter RelayState is used to describe the state information maintained at the service provider, for example, URL in Message (1). Apart from that, SP sets the AllowCreate attribute on the NameIDPolicy element to 'true' value to allow the IdP to generate a new identifier for the user that is not already exist.

$$\left\{\begin{array}{l}\text{HTTPredirect}\\ \text{responsefor}\\ \text{authenticationinIdP}\end{array}\right\}:=\left\{\begin{array}{l}\text{HTTPredirect}\\ \text{responsefor}\\ \text{useragent}\end{array}\right\} \quad (3)$$

The SSO component in the identity provider uses the public key PU_{sp} of service provider to verify the digital signature stored in the <cds: Signature> element which is included in the <AuthnRequest> element. If the verification is successful, the identity provider executes a security check. If the user agent does not have a valid logon security context, the identity provider requires the user to provide the valid logon credentials made up of usernameIdP and passwordIdP to be verified by the the identity provider. Thus the identity provider generates Message (3) and sends it to user agent.

$$\left\{\begin{array}{l}\text{Validcredentials}\\ \text{inIdP}\end{array}\right\}:=\left\{\begin{array}{l}\text{usernameIdP}\\ ||\text{passwordIdP}\end{array}\right\} \quad (4)$$

The user agent receives Message (3) and generates Message (4) which is made up of usernameIdP and passwordIdP and sends it to the identity provider through HTTP protocol.

$$\left\{\begin{array}{l}\text{HTTPresponse\&}\\ \text{HTTPPOSTrequest}\end{array}\right\}:=\left\{\begin{array}{l}\text{SAMLResponse}\\ ||\text{RelayState}\end{array}\right\}$$

$$\text{SAMLResponse}:=\left\{\begin{array}{l}\text{ID[Required]}||\\ \text{InResponseTo}\\ \text{[Required]}||\text{Version}\\ \text{[Required]}\\ ||\text{IssueInstant}\\ \text{[Required]}||\\ \text{Destination}\\ \text{[Optional]}\\ ||<\text{saml}:\text{Issuer}>\\ \text{[Optional]}||\\ <\text{cds}:\text{Signature}>\\ \text{[Optional]}||\\ <\text{Status}>\text{[Required]}\\ ||<\text{saml}:\text{Assertion}>||\\ <\text{Extensions}>\\ \text{[Optional]}\end{array}\right\} \quad (5)$$

$$<\text{saml}:\text{Assertion}>:=\left\{\begin{array}{l}\text{Version[Required]}||\\ \text{ID[Required]}\\ ||\text{IssueInstant}\\ \text{[Required]}\\ ||<\text{Issuer}>\\ \text{[Required]}\\ ||<\text{ds}:\text{Signature}>\\ \text{[Optional]}\\ ||<\text{AuthnStatement}>\\ ||<\text{Subject}>\\ \text{[Optional]}\end{array}\right\}$$

$$<\text{AuthnStatement}>:=\left\{\begin{array}{l}\text{AuthnInstant}\\ \text{[Required]}||\\ <\text{AuthnContext}>\\ \text{[Required]}\end{array}\right\}$$

When the identity provider receives Message (4), it firstly checks the validity of the credential of the user, which are usernameIdP and passwordIdP. If the verification is successful, then the SSO Service in the identity provider checks whether usernameIdP in its identity

database is or not and whether the AllowCreate attribute is true or not. If all is true, it creates a persistent name identifier SPandIdP, which is stored in the element persistentid in element <Extensions>, to be used for the session at the service provider. The persistent name identifier SP and IdP is used to link the account username IdP in the identity provider and username SP in the service provider. Apart from that, the identity provider produces Message (5) which is made up of SAMLResponse and RelayState parameters. The parameter RelayState is gotten through the service provider. The parameter SAMLResponse is mainly composed of ID, InResponseTo,Version, IssueInstant, Destination, <saml: Issuer>, <cds: Signature>, <Status> and <saml: Assertion>. <saml: Assertion> is the most important element in SAML Response. The local logon security context generated by the identity provider is stored in the SAML Assertion <saml: Assertion> element. The content in InResponseTo element is identical to the content in the ID element in <AuthnRequest> element. The digital signature of SAMLResponse generated with the private key PR^1_{Idp} of IdP is stored in <cds:Signature> element. <saml: Assertion> element is mainly composed of Version, ID, IssueInstant, <Issuer>, <ds: Signature>, <subject> and <AuthnStatement>. Among these elements the <ds: Signature> element is important because the digital signature of <saml:Assertion> generated with the private key PR^2_{Idp} is stored in <ds: Signature> elements. The authentication context information is stored in <AuthnStatement> element which is composed of AuthnInstant and <AuthnContext> elements. The usernameIdP is stored in the element <subject>. After Message (5) is produced, then it is sent to the service provider. Assertion Consumer Service component in the service provider will process Message (5).

$$\left\{\begin{array}{l}\text{HTTPredirect}\\ \text{responsefor}\\ \text{authentication}\\ \text{inSP}\end{array}\right\} := \left\{\begin{array}{l}\text{HTTPredirect}\\ \text{responseforuser}\\ \text{agent}\end{array}\right\} \quad (6)$$

When Message (5) arrives at the service provider, Assertion Consumer Service will processes it. Firstly, it uses the public key PU^1_{Idp} of the identity provider to verify the digital signature of <Response> stored in the element <cds: Signature>, and then uses the public key PU^2_{Idp} of the identity provider to verify the digital signature of <saml: Assertion> in <ds: Signature> element. Secondly, the service provider generates the local logon security context using the information stored in <saml: Assertion> element. Thirdly, the supplied name identifier SPandIdP is then used to check whether a previous federation has been established in the service provider identity database. If no federation exists for the persistent identifier in the assertion, then the service provider needs to determine the local identity to which it should be assigned. Finally, service provider sends Message (6) HTTP redirect response to user agent to challenge the usernameSP at the service provider.

$$\left\{\begin{array}{l}\text{validcredentials}\\ \text{inSP}\end{array}\right\} := \left\{\begin{array}{l}\text{usernameSP}\\ ||\text{passwordSP}\end{array}\right\} \quad (7)$$

When Message (6) arrives at user agent, the user provides valid credentials and identifies his account at the service provider as usernameSP. The persistent name identifier SPandIdP is then stored and registered with the usernameSP account along with the name of the identity provider that created the name identifier.

$$\left\{\begin{array}{l}\text{HTTPredirect}\\ \text{responsefor}\\ \text{theresource}\end{array}\right\} := \left\{\begin{array}{l}\text{HTTPredirect}\\ \text{response}\end{array}\right\} \quad (8)$$

After the service provider receives Message (7) which is made of usernameSP and passwordSP and makes a verification of the identity of user agent, If the verification is successful, a local logon security context is generated for user usernameSP. Apart from that, the federation is established between the usernameSP and usernameIdP through the persistent identifier SPandIdp in the service provider identity database. Finally the service provider generates Message (8) for the user agent for the desired resource. If the access check passes, the desired resource is returned to the browser.

6 Formalize Identity Federation Security Protocol Based on SAML 2.0 Using the Applied PI Calculus

6.1 Function and Equational Theory

The functions and equational theory are introduced in this section. We use the Applied PI calculus to formalize Identity Federation security protocol based on SAML 2.0. We model cryptography in a Dolev-Yao model as being perfect. Figure 4 describes the functions and the equational theory in the Identity Federation security protocol based on SAML.

$$\left\|\begin{array}{l}\text{fun sign(x,PR).}\\ \text{fun PU(c).}\\ \text{fun PR(c).}\\ \text{fun decsign(x,PU)}\\ \text{fun versign(y,PU)}\\ \text{equation versign(sign(x,PR),PU)=true.}\\ \text{equation decsign(sign(x,PR),PU)=x}\end{array}\right\|$$

Figure 4: The functions and the equational theory

.

Digital signature is modeled as being signature with message recovery, i.e. the signature itself contains the

signed message which can be extracted using the function. Digital signature algorithm includes the generation signature algorithm sign(x,PR) sign the message x with private key PR and the verification algorithm versign(y,PU) verify the digital signature y with public key PU. And the decsign(x,PU) recover the message from the digital signature x with the public key PU. The function PU(c) accepts private value c as input and produces public key as output. The function PR(c) accepts private value c as input and produces private key as output.

6.2 Process

The complete formal model of Identity Federation security protocol based on SAML 2.0 in the Applied PI calculus is given in Figures 5, 6, 7 and 8, which report the basic process include main process, user agent process, service provider process and identity provider process forming the model of Federation security protocol based on SAML. The main process IFSAML in Figure 5 sets up the process User Agent, Service Provider and Identity Provider.

```
IFSAML ≜
(!User Agent | !Service Provider | !Identity Provider)
```

Figure 5: Main process

The process User Agent is modeled using the Applied PI calculus in Figure 6.

```
User Agent ≜                                        (*User Agent(UA) process*)
 new url; new finish;
 out(pub,url);                                      (*UA sends the message1 to SP*)
 in(pub,httpgetrequest);                            (*UA rceives the message2 form SP*)
 let (uria,samlrequesta,relaystatea)=httpgetrequest in
 if relaystatea=url then out(pub,httpgetrequest);   (*UA sends the message2 to IdP*)

 in(pub,m3);                                        (*UA receives the message3 from IdP *)
 let reauthuseridp=m3 in
 new authuseridp;
 if authuseridp=reauthuseridp then
 let secretX= passwordidp in
 let valididp=(usernameidp,passwordidp) in
 out(pub,valididp);                                 (*UA sends the message4 to IdP*)

 in(pub,m5);                                        (*UA receives the message5 from IdP*)
 let httppostrequest=m5 in
 let (samlresponsea,responserelaystatea)=httppostrequest in
 if responserelaystatea=url then out(pub,httppostrequest); (*UA sends the message5 to SP*)

 in(pub,m7);                                        (*UA receives the message6 from SP*)
 let reauthusersp=m7 in
 if authusersp=reauthusersp then
 let secretY= passwordsp in
 let validsp=(usernamesp,passwordsp) in
 out(pub,validsp);                                  (*UA sends the message7 to SP*)

 in(pub,m8);if m8=resourse then out(pub,finish).(*UA receives the message8 form SP*)
```

Figure 6: Server agent process

Firstly, the User Agent produces the target resource address url by the statement new url and sands sends it to the service provider through the public channel pub. At the same time it also generates the information finish by y the statement new finish which shows that the protocol ends. After that, the User Agent receives the message httpgetrequest using the public channel pub by the statement in (pub, httpgetrequest).And then it extract the elements uria,samlrequest, relaystate from the message httpgetrequest the item uria is the address of SSO service component and is generated by service provider. The item samlrequest is URL-encoded <AuthnRequest> element in SAML and is also generated by service provider. The item relaystate the state information maintained at the service provider. User Agent compare the value relaystate with url. If they are equal then User Agent forwards the message httpgetrequest to the process Identity Provider through the public channel pub.

The User Agent receives Message m3 from the process Identity Provider through the public channel pub.Then it extracts the message reauthuseridp which shows that the user should provide the valid logon credentials. After that, the User Agent provides the username usernameidp and password passwordidp through valididp=(usernameidp,passwordidp).And also it sends valididp to Identity provider process by the public channel pub.

And then it receives Message m5 from the public channel c which is sent from the Identity provider process. The User Agent gets the message samlresponsea and responseerelaystatea from httpposttrequest. samlresponsea is mainly composed of ID, InResponseTo,Version, IssueInstant, Destination, <saml: Issuer>, <cds: Signature>, responserelaystatea is the target resource address. If the responserelaystatea is equal to url, and then the message httpposttrequest is sent to the Service Provider through the public channel pub.

After that, the User Agent receives message m7 from the Service Provider from the public channel c. Then it gets the message reathuserp which shows the user should provide the username and password. And then it generates his username usernamesp and password passwordsp and construct the message secrectY. The user Agent sends the message validsp through the public channel pub to the Servicer Provider.

Finally, it receives Message m9 through the public channel pub. If Message m9 is equal to resource, and then it sends the message finish from the public channel pub. The protocol ends.

The Service Provider process in Figure 7 receives Message (1) urlx from the public channel pub. In order to construct Message (2), firstly, it generates ID id, Version version, IssueInstant issueinstant, <saml: Iusser> iussuer and nameidppolicy nameidpolicy using the statements: new id; New version; New issuestant; New issuer; New nameidpolicy. And then it uses the digital signature function sign() to generate the digital signature signature of id, version, issuestant, iussuer,

nameippolicy with the Service Provider's private key PR(keysp). Then the SAMLRequest samlrequest is produced though let samlrequest=(id, version, issuestant, issuer, nameipdolicy) in . The SAMLRequest samlrequest mainly consists of id, version, issuestant, issuer, nameipdolicy. Finally uri,samlrequest,relaystate are used to construct Message (2) httpredirectresponse which is sent to the User Agent through the public channel pub.

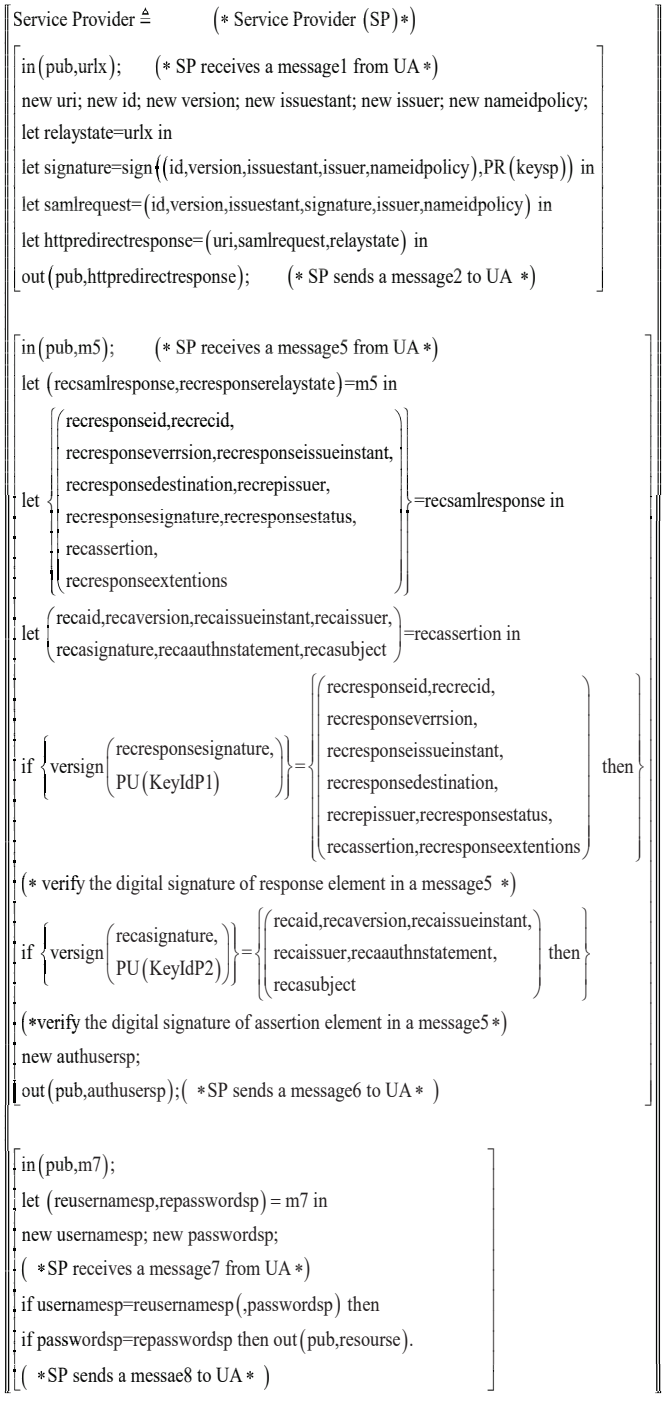


Figure 7: Server provider process

After that, it receives Message (5) from the User Agent process and gets the SAMLResponse recsamlresponse and RelayState recresponserelaystate form Message (5). Based on the SAMLResponse recsamlresponse, it generates ID recresponseid, InResponseTo recrecid, Version recresponseverrion, IssueInstantre recresponseissueinstant, Destination recresponsedestination, <saml: Issuer> recrepissuer, <cds: Signature> recresponsesignature, <Status> recresponsestatus and <saml: Assertion> recassertion. From the <saml: Assertion> element recassertion, Version recaversion, ID recaid, IssueInstant recaissueinstant, <Issuer> recaissuer, <ds: Signature> recasignature, <subject> recasubject and <AuthnStatement> recaauthnstatementare gotten. After that, the digital signature of <cds: Signature> recresponsesignature is verified by the function verign (recresponsesiganture, PU(KeyIdp1)) with the public key PUIdP1 of the Identity Provider. At the same time the digital signature of <ds: Signature> recsiganture is verified by the function versign (recsignature, PU(KeyIdp2)) with the public key PUIdP2 of the Identity Provider. If the two digital signature are all successful, the HTTP redirect response authusersp is generated and is sent to the User Agent through the public channel pub.

When Service Provider process receives Message m7 from the public channel pub, the usernameSP usernamesp and passwordSP passwordsp and makes a verification of the identity of user agent. If the verification is successful, then Service Provider generates Message (8) resource for the User Agent for the desired resource through the public channel pub.

The Identity Provider process in Figure 8 generates the elements responseid, responseverrion, responseissueinstant, responsedetination, aid, aversion, aissuestant, aissuer,aauthnstatement,asubject. And then it receives message m2 through the public channel pub. The Identity Provider process gets the elements URI recuri, SAMLRequest recsamlrequest and RelayState recrelaystate from Message m2 through the public channel pub. After that it extracts the elements ID recid, Version recversion, IssueInstant recissuestant, <saml: issue> recissuer and <cds: Signature> recsignature and NameID policy recnameidpolicy from the element SAMLRequest recsamlrewuest. Then, the Identity Provider process verifies the digital signaturerecsignature using the function versign (recsignature, PU(Keysp)) with the public key PU(Keysp) of Service Provider. If If the verification is successful, it generates message3 authuseridp which shows that the user should provide the valid logon credentials made up of usernameIdP and passwordIdP to be verified by the IdP. Thus the Identity Provider process sends Message (3) authuseridp to user agent process through the public channel pub.

After that, the Identity Provider process receives Message (4) m4 from the public channel pub. And then it extracts the usernameIdP usernameidp and passwordIdP passwordidp of the User Agent. It checks the validity of the credential of the user, which are usernameIdP and passwordIdP. If the verification is ok, it creates a per-

Table 1: The authentications

Non-Injective agreement	Authentications
ev:endauthUSERIDP(x)− >ev:geginauthUSERIDP(x)	Identity Provider authenticates User Agent
ev:endauthUSERSP(x)− > ev:geginauthUSERSP(x)	Server Provider authenticates User Agent
ev:endauthSAMLREQ(x) − > ev:beginauthSAMLREQ(x)	Identity Provider authenticates Server
ev:endauthSAMLRSP(x) − > ev:beginauthSAMLRSP(x)	Service Provider authenticate Identity Provider

sistent name identifier SPandIdP, which is stored in the element persistentid in element <Extensions>, to be used for the session at the service provider.

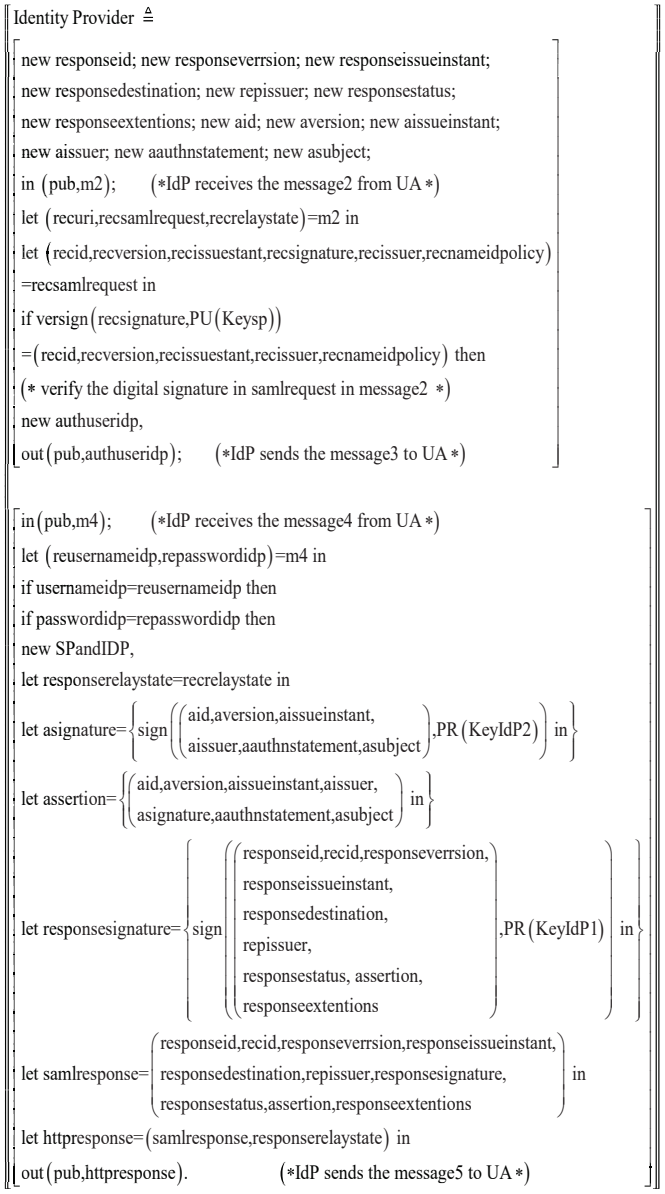


Figure 8: Identity provider process

Apart from that, the <ds: Signature> element asignature is produced by the digital signature function sign() with the inputs of <saml: Assertion> (aid, aversion, aissueinstant, aissuer, aathnstatement, asubject) and the private key PR(KeyIdP2) of Identity Provider. The <saml: Assertion> element assertion is mainly composed of Version aservsion, ID aid,IssueiInstant aissueinstant, <Issuer> aissuer, <dc: Signature> asignature, <subject> asubject and <AuthnStatement> aauthnstatement. At the same time the element <cds: Signature> responsesignature is generated by the digital signature function sign() with the inputs of (responseid,recid, responseserverrion, responseissueinstant, responsedestination,repissuer,reponsestatus, assertion,responseextentions) and the private key PR(KeyIdP1) of Identity Provider. Finally Message (5) httpresponse is generated which is made up of SAMLResponse samlresponse and RelayState responserelaystate parameters. The parameter SAMLResponse samlresponse is mainly composed of ID responseid, InResponseTo,Version responseverrion, IssueInstant responsedestination, <saml: Issuer> repissuer, <cds: Signature> responsesignture, <Status> responsestatus and <saml: Assertion> assertion and Message (5) httpresponse is sent to the user agent process through the public channel pub.

7 Automatic Verification of Secrecy and Authentications with ProVerif

Here we use the statements query attacker:secretX in ProVerif to verify the secrecy of which is the password of passwordidp the User Agent to assess the Identity Provider and query attacker:secretX is used to verify the secrecy of passwordidp to assess the Service Provider.

ProVerif uses the non-injective agreement to model the authentication. So we use query ev: event one–¿ ev:event two to model the authentication. It is true when if the event one has been executed, then the event event two must have been executed (before the event one). Here we use the non-injective agreement to model the authentications showed in Table 1 .

ProVerif can take two formats as input. The first one is in the form of Horn. The second one is in the form of a process in an extension of the Applied PI calculus. In both cases, the output of the system is essentially the same. In this study we use the Applied PI calculus as the input of ProVerif. In order to prove the authentication in Identity Federation security protocol based on SAML. The model using the Applied PI calculus is needed to be translated into the syntax of ProVerif and generated the

ProVerif inputs in extension of the PI calculus. Figures 9, 10, 11, 12, 13 and 14 are the inputs for Identity Federation security protocol based on SAML 2.0. We use the ProVerif to run the input for Identity Federation security protocol based on SAML 2.0 showed in Figure 9, 10, 11, 12, 13 and 14.

```
free pub.
free authuseridp,authusersp.
free usernamesp,passwordsp,usernameidp,passwordidp,resourse.
fun sign/2.
fun PU/1.
fun PR/1.
fun versign/2.
fun decsign/2.

equation versign(sign(x1,PR(y1)),PU(y1))=true.
```

Figure 9: The functions and equation in ProVerif

```
query attacker:secretX; (* the secrecy of passwordidp *)
query attacker:secretY. (* the secrecy of passwordsp *)

query ev:endauthUSERIDP(x) → ev: beginauthUSERIDP(x).
 (* Identity Provider authecticates User Agent *)
query ev:endauthUSERSP(x) → ev: beginauthUSERSP(x).
(* Service Provider authecticates User Agent *)
query ev:endauthSAMLREQ(x) → ev: beginauthSAMLREQ(x).
(* Identity Provider authecticates Service Provider *)
query ev:endauthSAMLRESP(x) → ev: beginauthSAMLRESP(x).
(* Service Provider authecticates Identity Provider *)
```

Figure 10: Query secrecy and authentications in ProVerif

```
let processuseragent =                    (*User Agent(UA) process *)
 new url; new finish;
 out(pub,url);  (*UA sends the message1 to SP *)
 in(pub,httpgetrequest);    (*UA rceives the message2 form SP *)
 let (uria,samlrequesta,relaystatea)=httpgetrequest in
 if relaystatea=url then out(pub,httpgetrequest);
 (*UA sends the message2 to IdP *)

 in(pub,m3);       (*UA receives the message3 from IdP *)
 let reauthuseridp=m3 in
 new authuseridp;
 if authuseridp=reauthuseridp then
 let secretX= passwordidp in
 let valididp=(usernameidp,passwordidp) in
 event beginauthUSERIDP(valididp);
 out(pub,valididp);
 (*UA sends the message4 to IdP *)

 in(pub,m5);       (*UA receives the message5 from IdP *)
 let httppostrequest=m5 in
 let (samlresponsea,responserelaystatea)=httppostrequest in
 if responserelaystatea=url then out(pub,httppostrequest);
  (*UA sends the message5 to SP *)

 in(pub,m7);       (*UA receives the message6 from SP *)
 let reauthusersp=m7 in
 if authusersp=reauthusersp then
 let secretY= passwordsp in
 let validsp=(usernamesp,passwordsp) in
 event beginauthUSERSP(validsp);
 out(pub,validsp);
  (*UA sends the message7 to SP*)

 in(pub,m8);if m8=resourse then out(pub,finish).
 (*UA receives the message8 form SP*)
```

Figure 11: The user agent process in ProVerif in ProVerif

```
let processserviceprovider =          (* Service Provider (SP)*)
 in(pub,urlx);      (* SP receives a message1 from UA *)
 new uri; new id; new version; new issuestant; new issuer;
 new nameidpolicy;
 let relaystate=urlx in
 let signature={sign((id,version,issuestant,issuer,nameidpolicy),PR(keysp))} in
 let samlrequest=(id,version,issuestant,signature,issuer,nameidpolicy) in
 let httpredirectresponse=(uri,samlrequest,relaystate) in
 event beginauthSAMLREQ(signature);
 out(pub,httpredirectresponse);
 (* SP sends a message2 to UA *)

 in(pub,m5);       (* SP receives a message5 from UA *)
 let (recsamlresponse,recresponserelaystate)=m5 in
 let {(recresponseid,recrecid,recresponseverrsion,
       recresponseissueinstant,recresponsedestination,
       recrepissuer,recresponsesignature,recresponsestatus,
       recassertion,recresponseextentions)}=recsamlresponse in
 let (recaid,recaversion,recaissueinstant,recaissuer,
      recasignature,recaauthnstatement,recasubject)=recassertion in
 if {versign(recresponsesignature,PU(KeyIdP1))}={(recresponseid,recrecid,
       recresponseverrsion,
       recresponseissueinstant,
       recresponsedestination,
       recrepissuer,recresponsestatus,
       recassertion,
       recresponseextentions)} then
 (* verify the digital signature of response element in a message5 *)
 event endauthSAMLRESP(recresponsesignature);
 if versign(recasignature,PU(KeyIdP2))={(recaid,recaversion,
       recaissueinstant,recaissuer,
       recaauthnstatement,
       recasubject)} then
 (*verify the digital signature of assertion element in a message5 *)
 new authusersp; out(pub,authusersp);( *SP sends a message6 to UA * )

 in(pub,m7);
 let (reusernamesp,repasswordsp) = m7 in
 new usernamesp; new passwordsp; ( *SP receives a message7 from UA *)
 if usernamesp=reusernamesp then
 if passwordsp=repasswordsp then
 event endauthUSERSP(m7);
 out(pub,resourse).
 ( *SP sends a messae8 to UA * )
```

Figure 12: The service provider process in ProVerif

```
process
new Keysp;
new KeyIdP1;
new KeyIdP2;
out(pub, PU(Keysp));
out(pub, PU(KeyIdP1));
out(pub, PU(KeyIdP2));
!processuseragent | !processserviceprovider | !processidentityprovider
```

Figure 13: The identity provider process in ProVerif

```
let processidentityprovider =
new responseid; new responseverrsion; new responseissueinstant;
new responsedestination; new repissuer; new responsestatus;
new responseextentions; new aid; new aversion; new aissueinstant;
new aissuer; new aauthnstatement; new asubject;
in (pub,m2);      (*IdP receives the message2 from UA*)
let (recuri,recsamlrequest,recrelaystate)=m2 in
let {(recid,recversion,recissuestant,
      recsignature,recissuer,recnameidpolicy)}=recsamlrequest in
if versign(recsignature,PU(Keysp))={(recid,recversion,
                                     recissuestant,recissuer,
                                     recnameidpolicy)} then
(* verify the digital signature in samlrequest in message2 *)
event endauthSAMLREQ(recsignature);
new authuseridp,
out(pub,authuseridp);      (*IdP sends the message3 to UA*)

in(pub,m4);      (*IdP receives the message4 from UA*)
let (reusernameidp,repasswordidp)=m4 in
if usernameidp=reusernameidp then
if passwordidp=repasswordidp then
new SPandIDP,
let responserelaystate=recrelaystate in
let asignature={sign((aid,aversion,aissueinstant,
                      aissuer,aauthnstatement,
                      asubject),PR(KeyIdP2))} in
let assertion={(aid,aversion,aissueinstant,
                aissuer,asignature,aauthnstatement,
                asubject)} in
let responsesignature={sign((responseid,recid,responseverrsion,
                             responseissueinstant,
                             responsedestination,repissuer,
                             responsestatus, assertion,
                             responseextentions),PR(KeyIdP1))} in
let samlresponse={(responseid,recid,responseverrsion,
                   responseissueinstant,responsedestination,
                   repissuer,responsesignature,
                   responsestatus,assertion,responseextentions)} in
let httpresponse=(samlresponse,responserelaystate) in
event endauthUSERIDP(m4);
event beginauthSAMLRESP(responsesignature);
out(pub,httpresponse).      (*IdP sends the message5 to UA*)
```

Figure 14: The main process in ProVerif

Figure 15 shows the result of the secrecy of query attacker:secrectX and query attacker:secrectY. From the result we find that the secrectX and secrectY have not secrecy. The result is consistent with the fact. That is because the secrectX and secrectY are sent in the way of plaintext.

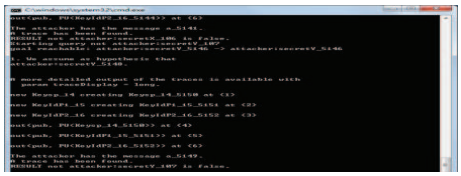


Figure 15: The results of secrecy

Hence the attacker can monitor the public channel to get the secrectX and secrectY. Hence the secrectX and secrectY have not secrecy. In order to implement the secrecy of the secrectX and secrectY some security mechanism must be used, for example, encryption.

Figure 16 shows the result that Identity Provider does not authenticate User Agent because the User Agent sends the password passwordidp in the way of plaintext to the Identity Provider. Hence the attacker can get the password passwordidp and launch an impersonation attack. We can use the encryption cipher or digital signature to address the problem.

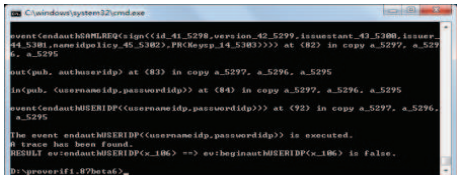


Figure 16: The result that identity provider does not authenticate user agent

Figure 17 shows the result that Service Provider does not authenticate User Agent because the User Agent sends the password passwordsp in the way of plaintext to the Service Provider. Hence the attacker can get the password passwordsp and launch an impersonation attack. We can use the encryption cipher or digital signature to address the problem.

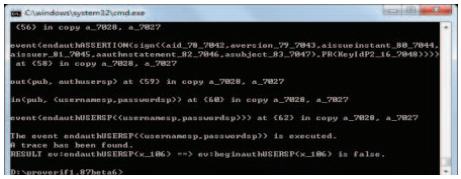


Figure 17: The result that service provider does not authenticate user agent

Figure 18 shows the result that Identity Provider can authenticate Service Provider because the Service Provider sends the its digital signature sign((id,version,issuestant,issuer,nameidpolicy),PR(keysp)) to the Identity Provider. Hence the Identity Provider can authenticate Service Provider.

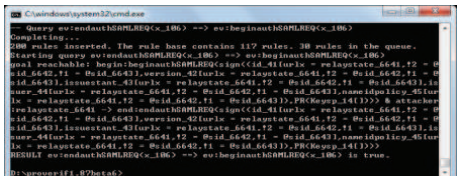


Figure 18: The result that identity provider authenticates service provider

Figure 19 shows the result that Service Provider can authenticate Identity Provider because the Identity Provider sends the digital signature sign((aid,aversion,aissueinstant,aissuer,aauthnstatement, asubject),PR(KeyIdP2)) to the Service Provider. Hence Service Provider can authenticate Identity Provider

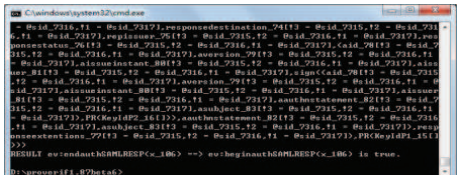


Figure 19: The result that service provider authenticates identity provider

8 Conclusion and Discussion

Owning to the complexity of Identity Federation Security Protocol, security analysis is important and has become a challenging issue. Therefore, in this paper, we firstly review the development of the formal methods on Identity Federation Security Protocol Based on SAML2.0, and then apply the ProVerif tool to analyze its security properties. The result shows that it has not secrecy for some keys and but it has some authentications. At the same time we present solutions to address the vulnerabilities.

Our method is basically similar to the method in the reference [1]. But there are two differences between the reference [1] and our work. The first difference is that in the reference [1], SAML SP-Initiated SSO profile is analyzed, but in our work, the identity federation security protocol based on SP-initiated SAML 2.0 includes eight messages exchanged among service provider, user agent and identity provider is analyzed. The second difference is that in reference [1] the model checker SATMC based on HLPSL++ as the specification language are used, but in our work, the an automatic cryptographic protocol verifier ProVerif and the Applied PI calculus are used.

In the near future, we will automatically analyze it in the computational model. At the same time, we will use the automatic method to generate the secure implementation in programming languages, for examples, JAVA language, C++ language.

Acknowledgments

This work was supported in part by the Fundamental Research Funds for the Central Universities, South Central University for Nationalities No. CZZ19003 and QSZ17007, and in part by the natural science foundation of Hubei Province under the grants No.2018ADC150."

References

[1] A. Alessandro, R. Carbone, L. Compagna, J. Cuéllar, G. Pellegrin-o, and A. Sorniotti, "An authentication flaw in browser-based single sign-on protocols: Impact and remediations," *Computers & Security*, vol. 33, pp. 41–58, 2013.

[2] D. S. AbdElminaam, "Improving the security of cloud computing by building new hybrid cryptography algorithms," *International Journal of Electronics and Information Engineering*, vol. 8, no. 1, pp. 40–48, 2018.

[3] R. Amin, N. Kumar, G.P. Biswas, R. Iqbal, , and V. Chang, "A light weight authentication protocol for iot-enabled devices in distributed cloud computing environment," *Future Generation Computer System*, vol. 78, no. 3, pp. 1005–1019, 2018.

[4] A. Bauer, J. Gross, P.L. Lumsdaine, M.Shulman, M. Sozeau, and B.Spitters, "The hott library: a formalization of homotopy type theory in coq," in *Proceedings of the 6th ACM SIGPLAN Conference on Certified Programs and Proofs (CPP2017)*, pp. 164–172, Paris, France, Jan. 2017.

[5] M. Beek and P.Moiso C.Petrocchi, "Towards security analyses of an identity federation protocol for web services in convergent networks," in *Proceedings of the 3rd Advanced International Conference on Telecommunications*, pp. 1–8, Morne, Mauritius, May. 2007.

[6] B. Blanchet, B. Smyth, V. Cheval, and M. Sylvestre. "Proverif 2.00: Automatic cryptographic protocol verifier, user manual and tutoria,". Tech. Rep. http://prosecco.gforge.inria.fr/personal/bblanche/proverif/manual.pdf, Apr. 2018.

[7] Q. Carbonneaux, J. Hoffmann, T. Reps, and Z. Shao, "Automated recourse analysis with coq proof objects," in *Proceedings of 29th International Conference on Computer-Aided Verification (CAV2017)*, pp. 64–85, Heidelberg, Germany, July. 2017.

[8] C. Guo, C. Chang, and S. Chang, "A secure and efficient mutual authentication and key agreement protocol with smart cards for wireless communications," *International Journal of Network Security*, vol. 20, no. 2, pp. 323–331, 2018.

[9] M. Y. Chen, C. W. Liu, and M. S. Hwang, "Securedropbox: A file encryption system suitable for cloud storage services," in *Proceedings of the 2013 ACM Cloud and Autonomic Computing Conference*, pp. 1–334, Miami,FL,USA, Aug. 2013.

[10] K. Chetioui, G. Orhanou, and S. Hajji, "New protocol e-dnssec to enhance dnssec security," *International Journal of Network Security*, vol. 20, no. 1, pp. 19–24, 2018.

[11] M. Zamani E.Ghazizadeh, J. A. Manan and A, "Pashang. a survey on security issues of federated identity in the cloud computing," in *Proceedings of IEEE 4th International Conference on Cloud Computing Technology and Science (CloudCom)*, pp. 1–4, Taipei, Taiwan, Dec. 2012.

[12] M. A. Elakrat and J. C. Jung, "Development of field programmable gate array–based encryption module to mitigate man-in-the-middle attack for nuclear power plant data communication network," *Nuclear Engineering and Technology*, vol. 50, no. 5, pp. 780–787, 2018.

[13] Y. F. Zhu K. Wang and M. Lin, "Provably secure browser-based mutual authentication protocol for federated identity management," *Application Research of Computers*, vol. 30, no. 6, pp. 1843–1846, 2013.

[14] X. H. Li, S. X. Li, J. Hao, Z. Y. Feng, and B. An, "Optimal personalized defense strategy against man-in-the-middle attack," in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17)*, pp. 593–599, San Francisco,USA, Feb. 2007.

[15] J. Ling, Y. Wang, and W. Chen, "An improved privacy protection security protocol based on nfc," *International Journal of Network Security*, vol. 19, no. 1, pp. 39–46, 2017.

[16] V. Mittal, S. Gupta, and T. Choudhury, "Comparative analysis of authentication and access control protocols against malicious attacks in wireless sensor networks," in *Proceedings of the First International Conference on SCI*, pp. 555–262, San Francisco,USA, Jan. 2018.

[17] M. S. Ferdous. and R. Poet, "Dynamic identity federation using security assertion markup language (saml)," in *Proceedings of the 3rd IFIP WG 11.6 Working Conference*, pp. 131–146, London, UK, Apr. 2013.

[18] F. Nabi Muhammad and Mustafa Nabi, "A process of security assurance properties unification for application logic," *Seventh International Conference on Complex, Intelligent, and Software Intensive Systems*, vol. 48, no. 6, pp. 40–48, 2017.

[19] G. Oliva, S. Cioabă, and C. N. Hadjicostis, "Distributed calculation of edge-disjoint spanning trees for robustifying distributed algorithms against man-in-the-middle attacks," *IEEE Transaction on Control of Network System,*, no. DOI: 10.1109/TCNS.2017.2746344, pp. 1–1, 2017.

[20] A. Marín-López P. A .Cabarcos, F. A. Mendoza and D. Díaz-Sánchez, "Enabling saml for dynamic identity federation management," in *Proceedings of the Second IFIP WG 6.8 Joint Conference on Wireless and Mobile Networking(WMNC'09)*, pp. 173–184, Gdańsk, Poland, Sep. 2009.

[21] F. G. Mármol P. A. Cabarcos, F. Almenárez and A. Marín, "To federate or not to federate: A reputation-based mechanism to dynamize cooperation in identity management," *Wireless Personal Communications*, vol. 75, no. 3, pp. 1769–1786, 2014.

[22] D. Puthal, M. S. Obaidat, P. Nanda, M. Prasad, S. P. Mohanty, and A. Y. Zomaya, "Secure and substainable load balancing of edge data centers in fog computing," *IEEE Communication Magazine*, vol. 56, no. 5, pp. 60–65, 2018.

[23] S. Saklikar and S. Saha, "Identity federation for voip systems," *Journal of Computer Security*, vol. 18, no. 4, pp. 499–540, 2010.

[24] K. Suthar and J. Patel, "Encryscation: An secure approach for data security using encryption and obfuscation techniques for iaas and daas services in cloud environment," in *Proceedings of International Conference on Communication and Networks, Advances in Intelligent System and Computing 508*, pp. 323–331, India, July. 2017.

[25] O. Wahballa1, Ab. Wahaballa, F. Li, I. Idris, and C. Xu, "Medical image encryption scheme based on arnold transformation and id-ak protocol," *International Journal of Network Security*, vol. 19, no. 5, pp. 776–784, 2017.

[26] C. Y. Yang, Y. Lin, and M. S. Hwang, "Downlink relay selection algorithm for amplifyand-forward cooperative communication systems," in *Seventh International Conference on Complex, Intelligent, and Software Intensive Systems*, pp. 331–334, Dalian, China, July 2013.

[27] L. Yao, J. Liu, D. Wang, J. Li, and B. Meng, "Formal analysis of sdn authentication protocol with mechanized protocol verifier in the symbolic model," *International Journal of Network Security*, vol. 20, no. 6, pp. 1125–1136, 2018.

Biography

Jintian Lu received his M.S degree at school of computer, South-Center University for Nationalities, China. Now he is pursuing the Ph.D. degree with School of Data and Computer Science, Sun Yat-sen University, Guangzhou, Guangdong, China. His current research interests include the security of security protocol and its implementations and cloud security.

Xudong He was born in 1991 and is now a postgraduate at school of Computer Science, South-Central University for Nationalities. His research interests include: security protocol implementations and reverse engineering.

Yitong Yang was born in 1991 and is now a postgraduate at the school of computer, South-Center University for Nationalities, China. Her current research interests include security protocols and formal methods.

Dejun Wang was born in 1974 and received his Ph.D. in information security at Wuhan University in China. Currently, he is an associate professor in the school of computer, South-Center University for Nationalities, China.

He has authored/coauthored over 20 papers in international/national journals and conferences. His current research interests include security protocols and formal methods.

Bo Meng was born in 1974 in China. He received his M.S. degree in computer science and technology in 2000 and his Ph.D. degree in traffic information engineering and control from Wuhan University of Technology at Wuhan, China in 2003. From 2004 to 2006, he worked at Wuhan University as a postdoctoral researcher in information security. Currently, he is a full Professor at the school of computer, South-Center University for Nationalities, China. He has authored/coauthored over 50 papers in International/National journals and conferences. In addition, he has also published a book "secure remote voting protocol" in the science press in China. His current research interests include Cyberspace security.

Social Media, Cyberhoaxes and National Security: Threats and Protection in Indonesian Cyberspace

Budi Gunawan and Barito Mulyo Ratmono
(Corresponding author: Budi Gunawan)

Higher School of State Intelligence
Sumur Batu, Babakan Madang, Bogor, West Java 16810, Indonesia
(Email: bgunawan9916@gmail.com)
(Received Mar. 3, 2018; Revised and Accepted Sept. 7, 2018; First Online July 11, 2019)

Abstract

This study examines the proliferation of hoaxes and hate speech through websites and social media in Indonesia. Such provocative content utilizes sectarian issues to attack its creators' political opponents. This study finds that hate has been politicized and hoaxes have been commodified, both for economic and political interests, in cyberspace. There has been a transformation from freedom of speech to freedom to hate, particularly on social networks. This proliferation of hoaxes, as a means of furthering specific political interests, may potentially threaten national security and stability. To overcome the threat posed by cyberhoaxes, the state, industry, and society must take an active role in protecting cyberspace.

Keywords: Cyberhoax; Cyber Security; Freedom to Hate; Politics of Threat

1 Introduction

Since mid-2015, hoaxes and fake news have become increasingly common in Indonesia, particularly on the internet and social media. This was not the first time that hoaxes spread in Indonesia. For example, during the 2014 presidential election the tabloid Obor Rakyat ('Torch of the People') deliberately disseminated provocative fake news and emphasized sectarian issues to attack political opponents. Similar cases have occurred in India, the United States, Germany, China, France, and Malaysia, where accurate news has been mixed with gossip and hate speech before being rapidly spread through social media.

The proliferation of hoaxes has been made possible through the widespread adoption of Facebook, Twitter, WhatsApp, Line, Google+, and other new media platforms, which have made the rapid dissemination of information possible through their high degrees of interactivity and interconnectivity. Hoaxes have spread uncontrolled through cyberspace, and some have had serious social implications. In response to hoaxes, people have been killed and national stability and security has been threatened. Most hoaxes have involved fake news about sensitive tribal, religious, and racial issues as well as hate speech directed towards those in power. The razing of Chinese temples in Tanjung Balai, North Sumatra, in July 2016, is just one example of social unrest and conflict caused by hoaxes disseminated through social media. Likewise, national security was threatened by hoaxes related to Chinese migrant labor that began to be spread in mid-2015.

The cyberhoax phenomenon has become crucial in an Indonesian context, and as such requires serious attention, particularly given that half of Indonesians are active internet users. According to a survey by the Association of Indonesian Internet Service Providers (Asosiasi Penyelenggara Jasa Internet Indonesia, APJII), in 2016 more than half of Indonesia's population enjoyed internet access. Of Indonesia's population of 256.2 million, 132.7 million actively use the internet. This represents a 51.8 percent increase from 2014. Similarly, a survey by the Singapore-based social marketing firm showed that internet penetration in Indonesia had reached 51 percent in January 2017.

The rapid increase in internet usage in Indonesia has been supported by new media technologies such as smartphones and tablets. According to the Directorate General of Public Communications and Information at the Ministry of Communications and Information, in 2013 some 240 million gadgets were in use in Indonesia (Kompas, 13/04/2015). Meanwhile, according to We are Social, as of January 2015 some 308.2 million cellular phones are used in Indonesia. A 2016 survey by APJII showed that most mobile gadgets in Indonesia (including smartphones and tablets) are used to access the internet, either to seek information or to participate actively in social networks. From this data, it is clear that half of Indonesia's population uses the internet and relies on new media technology

in everyday personal and social activities. They are targeted by the cyberhoaxes and hate speech produced and circulated online.

This study will examine the cyberhoaxes in Indonesia and their implications for national security and stability. It focuses on the production and dissemination of hoaxes, particularly those that discredit the government, by five Indonesian websites—saracennews.com, postmetro.com, nusanews. com, portalpiyungan.co, and NBCIndonesia. com—between 2015 and 2017. This study is intended to examine the practice, identify actors involved, and the interests that inform their activities. Furthermore, this study will also examine the potential security threat posed by such hoaxes and the possibility of defending cyberspace as part of national security.

2 Cyberhoaxes and Politics of Threat

In everyday discourse, hoaxes are often understood as untrue or fake news. Boese, in his book The Museum of Hoaxes (2002), defines hoaxes as deception involving public response [15]. Boese writes that hoaxes are lies that successfully draw the attention and imagination of the public. In their study of the hoaxes perpetrated by Alex Sokal, Marie Sekor and Linda Walsh (2004) conclude that hoaxes are rhetoric devices used deliberately to attack those opposed to the hoaxer. In the case of Sokal, they identify two types of consumers/readers, i.e. those capable of quickly recognizing the intent of the hoaxer and become co-conspirators by rapidly and massively distributing and circulating the hoax, and those deceived by the hoax and ashamed of this fact.

Today, hoaxes generally operate using internet-based new media. User generated content platforms such as weblogs and social network accounts enable hoaxers to hide their authorship, and thus rapidly and anonymously disseminate their deceit—which Lovell defines as "content whose main purpose is to attract attention and encourage visitors to click on a link to a particular webpage" [11]—is also an important factor in the dissemination of deceptive content, as it enables people to spread hoaxes with a single click. Hoaxes are similar to chain letters in their distribution, as they are normally presented together with buttons intended to facilitated their reposting.

Referring to the findings of Sekor & Walsh [15] that hoaxes are devices used to attack one's opposition, power and control are concentrated on a single button used to attack others. This is congruent with the concept of dromology introduced by Virilio in Speed and Politics [16]. According to Virilio, war and conflict has been dematerialized, as people in conflict no longer require physical territory for conquest. The mobilization of soldiers and weapons is no longer necessary, as victory relies only on the vectors of virtual technology and speed with which a "virtual" button is pushed. Weapons no longer need to be borne by soldiers, as attacks can be made with the push of a button. Similarly, cyberhoaxes are a form of virtual warfare, with its attacks being rooted in visual imagery and clickbait. As such, it is not excessive to identify cyberhoaxes as part of the politics of threat practiced in cyberspace.

In Cyber Security and Threat Politics: US Efforts to Secure the Information Age, Myriam Dunn Cavelty [4] emphasizes the different threats that have emerged together with information and communication technologies [4]. More specifically, Cavelty positions cyberthreats as products of irresponsible use of global information infrastructures. According to Cavelty, threats in cyberspace must be understood as part of the political process, as their dynamics, characteristics, and transformations are framed and informed by political agendas. As an example, she identifies how cyberthreats have become important parts of national security agendas in the 21st century, both in the United States and in the United Kingdom. Where these countries' national security policies once focused solely on material threats that clearly and physically endangered people, the increased integration of information and communications technology in everyday life has transformed countries' framing of national threats and national security.

3 Cyberhoaxes in Indonesia

In Indonesia, the emergence of new media has invigorated civil society and empowerment movements, particularly following the fall of the New Order regime. Cyberspace has seemed to promise citizens the freedom of expression and active participation in political processes. At the same time, general elections, a common manifestation of the democratization process, have been faced with intense public distrust. Few citizens trust political parties or the commitment and performance of politicians. There has been considerable public disappointment in and resistance to political processes. In cyberspace, people have greater opportunity to voice criticism and resist those in power, something not possible under authoritarian regimes. However, resulting excesses have become the basis for fake news and hate speech in Indonesia.

3.1 From "Freedom of Speech" to "Freedom to Hate"

According to previous research into cyberhoaxes, the websites in this researcher essentially follow the same template. These five websites position hoaxes within political contexts, particularly presidential and regional elections. In general, the creators of these hoaxes are considerably disappointed in election results, and they are dissatisfied with the performance of the political party and government in power. Initially, these hoaxers positioned themselves as critics of the government. They feel themselves to be 'victims' of government policies that they consider incapable of accommodating public interests. As citizens,

they seek to represent people in similar positions.

Hoaxers' lack of trust in those in power is the basis for their criticism in cyberspace. They feel dissatisfied with the performance of the government and feel that their own interests are marginalized. This has, to some extent, become a positive influence on the democratization process. As mentioned by Dahlgren, voice is an important aspect of political participation [5]. Citing Couldry (2010), Dahlgren explains that having a voice is a fundamental part of being human, and as such silencing someone's voice is an affront to humanity. However, in the neoliberal structure some voices are unfortunately marginalized through particular economic and political designs. All peoples' voices should be accommodated within public space. The internet has been hoped to become such a new public space, in which once marginalized voices can be accommodated as a manifestation of active political participation.

This initial logic lies behind the rise of websites with provocative content. The administrators of these websites felt disappointed because they perceived that their own interests were being marginalized. They felt that those in power, whom they hoped would defend their interests, were not performing as they hoped. They also felt that those in mainstream media were unable and unwilling to promote the interests of the common people, with corporate-owned media being not neutral in their coverage because their owners are political elites affiliated with the government. Consequently, they held that mainstream media served only to support those in power. Observers and informants in the media likewise, they argued, supported the status quo.

The administrators of hoax websites thus used cyberspace as an alternative space for resisting and criticizing those in power, hoping to transform the dynamics and policies of the government. This is important given that freedom of speech and public participation have frequently been promoted during the democratization process. Citizens, it is argued, should have the agency to voice their opinions and inform governance. However, their criticism and resistance has been transformed into anarchy, hatred, and agitation, while their disapproval has transformed into provocation and incitement. They have positioned the government and those affiliated with it as common enemies to be conquered. The freedom of expression facilitated by cyberspace has been transformed into the freedom to hate. In elections, where different political parties compete for volunteers and buzzers to promote their candidates and challenge their opponents. As noted by Lim [10], in the 2017 Jakarta election volunteers and buzzers generally defend their activities as part of freedom of speech even as they were silencing their opponents. They demanded freedom of speech for their own interests, but silenced those whose interests were opposed to their group. As a result of such practices, constructive criticism was reduced to deception and hate speech with a minimal basis in objective fact as shown in Figure 1.

Figure 1: Opinions about the president is a communist

3.2 The Production and Dissemination of Hoaxes

Interestingly, the cyberhoaxes perpetrated by the five websites investigated in this researcher utilize a similar modus operandi. Desiring to criticize the government in power, the website administrators use cyberspace to voice their aspirations, holding that social media and microblogs do not offer them sufficient space to promote their interests. Visibility is a central aspect of public participation [5], as people seek to gain the attention and recognition of others for their interests. This, according to Dahlgren, creates a "regime of democratic visibility".

The desire for visibility and attention, not only from those being criticized but also from others, underlay administrators' decision to create websites where they began producing hoaxes. These websites lack clear information on their founders, and their "About Us" pages appear perfunctory or even deceptive. Information on these websites' organizational structures and addresses are often not included.

At their core, these hoax websites rely on the journalistic products of the mainstream media. In selecting specific issues, they observe mainstream media coverage. Issues with the potential for controversy and support the administrators' own interests (or can be used to attack their opponents) are identified and selected. The issues they select are modified by administrators using one or more of the techniques discussed below. First, facts may be exaggerated with fiction, particularly that which can be mobilized to promote tribal, religious, and racial tensions and hatred. Second, the substance of the story may remain unchanged, but be given a provocative and bombastic headline. Third, the main points of the coverage may be maintained, but presented in clear, direct, and

provocative language. Fourth, the titles of photographs or illustrations may be changed to make them more provocative. Fifth, photographs or illustrations of incidents unrelated to that being covered may be used to suggest a connection and thereby provoke readers.

Aside from modifying coverage from the mainstream media, the administrators of these websites may also cover statements and opinions from politicians and commentators who share their vision. To do so, the administrators cultivate relations and friendships with such politicians and commentators, who are frequently opposed to existing government policy. Furthermore, these politicians and commentators are used as references or given space to voice their (anti-government) opinions on the websites.

The fake news and hate speech produced by these websites are not journalistic products that follow the accuracy and accountability standards of the profession. However, the website administrators do not care that their content violates journalistic principles. The owner and operator of postmetro.co, for example, holds that many in the mainstream media are dishonest and deliberately violate journalistic ethics. He views his website as only reproducing practices that are common, even in reputable mainstream media. They also do not fear their websites being blocked by the government, as their medium allows them to create new websites readily. For example, postmetro.com has changed domains several times after being blocked; it was first posmetro.info, before becoming postmetro.com and finally postmetro.co. They do not fear losing readers, because they believe that their readers are loyal consumers that will actively seek out their new domains. Their relations with their audiences are more emotional than rational; if these relations were rational, readers would reconsider trusting such sources or seeking content updates from websites with little accountability.

Once fake news items are produced, it is most crucial to distribute them. The most rapid means of distributing such fake news is making it go viral. To reach as many internet users as possible, the stories must become as virulent as possible. These websites' visibility and ability to draw readers' interest is key to their popularity. Website administrators are perfectly aware that, to become popular, the stories on their websites must become viral, and for this they rely on social media. As noted by Allcott & Gentzkow [1] in their study of fake news in the United States, "...social media are well-suited for fake news dissemination, and social media use has risen sharply." In an Indonesian context, Lim [10] in her investigation of the 2017 Jakarta gubernatorial election, also identified the role of social media in disseminating fake news, sectarian provocations, and racist content. The selection of social media for the virulation of hoaxes in Indonesia is not without grounds. According to data from We are Social, as of January 2017, 92 million Indonesians use social media on their mobile devices. Facebook, one of the most popular social media platforms, records 106 million Indonesian users. Based on this data, it can be said that social media networks have become an integral part of Indonesians' mobility. Regarding this, Lim (2017) writes:

> "Across the world, and most certainly in Indonesia, the expansion of social media usage has sparked new hopes of and hype about political participation and civic engagement."

Based on this statement, it is apparent that social media—particularly in Indonesia—has been seen as offering the potential to empower people and increase civil participation in political processes. These new media platforms are also thought to ease residents' shaping and sharing of their political opinions and aspirations. Nonetheless, Lim also identifies pessimism for the negative aspects of social media, such as loss of privacy, decreased quality of information, proliferation of lies, and emergence of online radical groups. This last tendency has become problematic in Indonesia, particularly given the widespread dissemination of hoaxes in social media. The popularity of social media in Indonesia has eased hoaxers in rapidly spreading provocative content (Figure 2).

Figure 2: Another lie about Freeport

3.3 The Commodification of Hoaxes and Politicization of Hate

Why have fake news and hate speech become widespread on the internet, particularly on social networks? Allcott & Gentzkow [1] write that, in the United States, the main reason for spreading hoaxes are financial. The virulation of fake news through social media promises them tempting financial incentives. Every news story that goes viral will bring significant financial income for the website that originated it. Clickbait logic does not only promote the virulation of hoaxes, but also directs traffic towards the

websites that originate hoaxes, bringing significant funding for their administrators. A similar phenomenon is apparent in Indonesia.

Although the administrators of websites that spread hoaxes and hate speech originally intended to promote active participation in political processes and their ideals, the financial income they have received has transformed their orientation. Their political participation was thus easily diverted towards the seeking of profit, a transformation made possible by the media industry that has used cyberspace for its political activism. Media platforms such as Facebook and Google AdSense have had an important role in promoting the commodification of hoaxes and hate speech. The advertising revenue that they receive for every click offers website administrators an extraordinary financial incentive. The higher the traffic (i.e. readers) on websites that originate hoaxes and hate speech, the higher the revenue received from advertisements.

According to the administrators of hoax websites, they can earn an annual income of 600 to 700 million rupiah from advertisements. For example, by producing some eighty fake news stories per annum, the administrator of postmetro.com can earn some 25 to 30 million per month. Owing to the highly profitable nature of deceit and agitation, postmetro.com recruited several personnel to manage the website. A new administrative structure was established, with staff working specifically on seeking out stories from the mainstream media and rewriting them with their own titles and styles. Others, meanwhile, focus specifically on the virulation of their agitation and deceit.

Similar practices are used by nusanews.com and NBCIndonesia.com. Meanwhile, nusanews.com and postmetro.co spent some time as partners. The success of these websites' advertisements can be measured through websites such as Site Worth Traffic. A single hoax story that is seen 1,000 times will earn US $1 for the hoaxer; clicks on advertisements earn them US $0.04 each. As such, the amount of money earned by websites can be measured by the number of visitors. Using Site Worth Traffic, it can be seen that NBCIndonesia.com—before being blocked by the government—received an average of 481 visits/day, 83.73% of which came from Facebook. As such, the website operator could earn US $194 per day or US $69,840 (approximately 1 billion rupiah) per annum, a fantastic amount. To maximize its income from advertisements, another hoax website, portalpiyungan.co, hired an advertising consultant to ease its dissemination of fake news. The results were impressive. Advertising income from portalpiyungan.com increased from 1.5 million per month to 150 million per month [12].

Of the five hoax providers discussed here, only Saracen was organized and had massive operations. Police investigations into the Saracen syndicate found that 33 people were involved, divided into two groups. The first group, the core team, consisted of 22 people. This team was responsible for the production of hoaxes, including fake news, hate speech, and provocative memes. The second team, which consisted of 11 people, worked to disseminate these hoaxes by making them go viral on social media. This indicates that, within this group at least, hoaxes were viewed as a business needing a professional and structured system. According to reports, the Saracen syndicate offered hoax production and dissemination services, charging between 75 and 100 million rupiah. Following the laws of supply and demand, such hoaxing practices have emerged not only because of the availability of advertising platforms such as Google AdSense, but also because of demand from specific sectors. Hoaxers have people—most of whom remain invisible—who pay for their services and their website management.

In Economies of Signs and Spaces, Lash & Urry [8] writes that, in the 21st century, the capitalist economy is no longer motored by tangible goods, but rather by indeterminate symbols that are borderless and fluid. According to Lash & Urry, the production of symbols has proliferated cognitive symbols (such as information and digital codes), abstract symbols, as well as aesthetic and expressive symbols that are used in representation (i.e. branding and image-building). The production of hoaxes is one such example. As businesses, cyberhoax websites produce intangible products, specifically cognitive symbols of hate and incitement presented as inaccurate information and hate speech. Fake news is made more "pretty" and interesting by packaging real news stories as provocative stories, with misleading titles and fiction replacing fact. To draw attention, they mobilize sensitive issues to draw readers' anger and hatred. Tribal, religious, and racial tensions are exploited to draw public attention. Symbolic and sectarian language (such as kafir "unbeliever", cina "Chinese", penista agama "blasphemer", and komunis "communist') is used to exploit primordial tensions. This language is circulated in the unlimited space known as cyberspace for mass consumption. These hoaxes are disseminated among the masses and bring with them significant profits.

The proliferation of hoaxes and hate speech in cyberspace also has ideological reasons, as it is intended to exert power and to counter that exerted by those in power. As such, these actors seemingly attempt to utilize the optimistic views of politics and civil participation. Owing to their disappointment in various government policies, which they consider to marginalize them or not represent their interests, these actors have used social media as an alternative space for voicing their views and searching for information.

The sharing and reposting of provocative news, filled with hate and incitement, can be understood as a means of gaining others' attention and recognition, part of the desire for social visibility [5]. In the context of political participation, as supported by new media platforms, the accumulation of knowledge also offers users' power. The sharing, reposting, and retweeting of news on social media indicates that this practice is perpetrated by those already drawn-in by the stories. They are people with

knowledge of these stories and the issues they contain. The sharing of news stories on social media is intended to show their friends that they are knowledgeable. In other words, the people who regularly share posts on social media are those with the desire for power, as they seek to control and shape their friends' knowledge. People such as these frequently become opinion leaders among their friends. By sharing news stories, they feel in power, as they accumulate knowledge and gain greater visibility in social media. This feeling of being in power brings them pleasure. Such actors work on their own, becoming effective means of disseminating the hoaxes and provocations produced by hoaxers. Actors involved in cyberhoax practices and their underlying interests can be seen in Table 1.

3.4 Production of Uncertainty: Political Symbolism and Misinformation

As economic and political commodities, hoaxes represent an exchange of deceptive and inflammatory symbols. According to Edelman [6], the use of such symbols in a political context is an element of political symbolism [7]. The proliferation of symbols, including their use and misuse, is intended to manipulate political discourse and public opinion, According to Edelman, symbols have taken an increasingly important role in politics. Political influence and power is no longer based on material and objective facts, but the mobilization of symbols. For example, political symbolism is rampant in political campaigns. Eriksson & Giacomello [7] argue that political practices in the digital media ecosystem emphasize the exploiting of various symbols for mass mobilization and manipulation.

In the practice of hoaxing, linguistic symbols (both verbal and visual) are used to construct certain views of the issues discussed. Edelman [6] identifies two different types of symbols used in political practice symbolism: referential symbols and condensation symbols [2]. Referential systems are those related to objective elements of certain situations and objects. These symbols are frequently used to legitimize specific political views and guide the masses towards a specific and shared understanding of a situation or object, such as statistics or budgets. Meanwhile, condensational symbols are those that create certain emotions and subjective reactions to a situation or object. Such symbols are capable of shaping people's imagination of a desired world, one quite different from the real world. It is such condensational symbols that are mobilized by cyberhoaxes in Indonesia. Nonetheless, according to Edelman, both types of symbols can be used to manipulate public discourse and public opinion about certain issues. This is one-sided, intended to justify specific ideas and logics.

In cyberspace, political symbolism promotes specific simplified narratives and framings of certain situations and objects. The new media, which enables the consumption of information (and distraction), contributes importantly to this symbolization process. This can be seen, for example, in the use of clickbait, in which symbols (images) in cyberspace serve are provided as "keys" to exploring issues and problems. Through clickbait, overly simplified logics are brought into the digital ecosystem. Lim [9] writes that it is no surprise that "trailer vision" dominates social media, with "light packages" or simplified narratives presented to whet audiences "headline appetite". The (over) simplification of narratives is common in new media, and consequently very few media users seek detailed information or seriously investigate the events and processes reported. Spaces for discussion and reflection disappear as access is accelerated. Events and processes are framed as nothing but headlines, visual images that draw the eye, and short commentary. All of this is oriented towards rapid consumption, and as a result various problems are reduced and simplified through one-sided and stereotypical coverage.

Political symbolism and simplification of issues are national threats that must be minded, as they are deceptive and present nothing but misinformation mistaken beliefs [6], which can promote improper activities and result in physical conflict. The politics of threat use political symbols to produce uncertainty within society, particularly in the tense periods in the lead-up to elections. In such times of uncertainty, chaos is very possible, as people lack clear and objective information.

3.5 Tribal Nationalism based on Political Identity

When undergoing activities on social media, users frequently ignore platforms' ability to filter and sort users' digital activities. Social media platforms such as Facebook, for example, use specific mechanisms to identify users' interests and content, and presenting content to users with specific interests. These algorithms construct what is known as a bubble, in which people are isolated from different people and their diverse opinions and views [10]. As a result, social media users are only exposed to content that reaffirms their own political views and people that share said views. Differences of opinion, as well as argument, are seemingly eliminated by the "bubble" created. Because people's political preferences differ, these filters and algorithms produce bubbles of shared political views that can be termed algorithmic enclaves [10]. Lim defines these algorithmic enclaves as groups "that are formed whenever a group of individuals, facilitated by their constant interactions with algorithms, attempt to create a (perceived) shared identity online for defending their beliefs and protecting their resources from both real and perceived threats".

Algorithmic enclaves are dynamic imagined communities, membership in which may change over time. Their algorithms focus on sorting, classifying, and creating a hierarchy of political preferences, information, and people.

Borrowing Lim's concept of algorithmic enclaves, hoax consumers establish their own enclaves. Such enclaves are formed as part of an identity formation process, in which they use their resources to defend their beliefs and

Table 1: Actors, roles, and interests in cyberhoaxes

Actor	Role	Interest
Hoaxers	Commodification of Hoaxes: - Producing and disseminating hate and incitement - Seeking clients for their services	Economic: Profiting from advertisements (up to 600 million–700 million per year).
Clients	- Pay for production and virulation of hoaxes - Conspirators in hoaxing	Political: - Spread hate and incitement against their opponents
Observers/ Politicians	Politicization of Hate: - Use hoaxes as political commodities - Co-conspirators in hoaxing - Make hoaxes go viral	Political: - Spread hate and incitement against their opponents - Political personal branding - Become visible and gain political influence
Consumers/	- Consume hoaxes - Make hoaxes go viral - Co-conspirators in hoaxing	Political: - Spread hate and incitement against their opponents
	- Consume hoaxes - Make hoaxes go viral	Pleasure: Become visible and gain public attention - Feel in power

to protect themselves from threats. Such groups establish their own "tribes", which live in and influence cyberspace. Borrowing a concept first formulated by the German political scientist Hannah Arendt, Lim [10] argues that such algorithmic enclaves promote the development of tribal nationalism. In her book, the Origins of Totalitarianism, Arendt (1973) identifies tribal nationalism as one that differs from mainstream models. Where the mainstream model of nationalism is constructed on actual political experiences, tribal nationalism is based on a sense of feeling and inner soul [3]. In other words, within tribal nationalism there exists a disconnect with real-world political processes. This sense of nationalism is primarily based on a sense of fate shared among a "tribe" (group). The spreading of hoaxes on Indonesian social media also has the potential to create tribal nationalism through which political identities are mobilized. This can be seen in the religious sentiments that underlie many of the hoaxes shared on the five websites examined here, as well as the stories shared on social media. Although it is true that religious sentiments are not the only ones exploited to provoke readers—questions of ethnicity and liberalism are also used by these websites—these non-religious issues are ultimately subordinated to religions ones.

The administrators of hoax websites position Islam as a "victim", despite the religion being the most commonly practiced in Indonesia. As mentioned by Arendt, those who create a sense of tribal nationalism tend to feel threatened by "outsiders". They feel surrounded by a "world of enemies", and thus seek to create a shared sense of solidarity and struggle [3]. They may quickly form mobs and create social fragmentation. For example, the 212 Demonstrations mobilized religious issues to attack incumbent governor Basuki Tjahaya Purnama during the 2017 Jakarta gubernatorial election. In particular, the administrator of postmetro.co expressed satisfaction with the fake news he created and its ability to unite Muslims against the governor. Members of such groups tend to legitimize the exclusion of persons outside their group. For example, again using the 2017 Jakarta gubernatorial elections, a number of imams and mosque administrators refused to pray for deceased community members with different political beliefs (i.e. who supported Basuki Tjahaya Purnama). In cyber hoaxes, tribes are created through online enclaves that are formed through the production and dissemination of hoax. The emergence of political tribes poses a serious threat to national unity and stability.

4 Cybersecurity: Combatting and Preventing Hoaxes

The production and dissemination of cyberhoaxes and hate speech are part of the politics of threat and designed by certain actors to promote certain interests. Hoaxes, as with cyber threats in general, are not material, nor do they cause direct physical harm to humans. Nonetheless, they have serious social effects. In other words, the cyberhoaxes that have become increasingly widespread in Indonesia have the potential to threaten national stability. Pursuant to Law No. 17 of 2011 on State Intelligence, intelligence is an important means of maintaining national security, and the Indonesian State Intelligence Agency is tasked with the coordination of the national intelligence system. In combating the hoaxes that have become rampant in Indonesia, the Indonesian State Intelligence Agency is also tasked and authorized to conduct

studies regarding various threats and the potential dangers they pose. Once these threats and their potential dangers have been identified, the Indonesian State Intelligence Agency has the duty and authority to combat them and anticipate the emergence of further threats—i.e. the future spread of hoaxes. The role and contribution of the Indonesian State Intelligence Agency is central, recognizing that several social conflicts have become physical because of the proliferation of hoaxes.

To combat cyberhoaxes, three different parties must work together: the state, market, and civil society. They must collaborate to address various strategic issues that threaten cybersecurity (in particular), as well as national security and stability in general. In the context of national authority, specific legal products must be prepared to provide stricter judicative sanctions. Referring to the concept of reflexive politics presented by Beck in his book Risk Society (1992), in risk management the government does not need rule-directed politics (i.e. politics based on existing regulations). Because risks are always transforming, society requires what is known as rule-altering politics [14]. To provide cybersecurity in Indonesia, particularly against cyberhoaxes, it is possible to apply this latter concept, for example by regulating domain ownership and setting fines for platform providers. Furthermore, it is important to increase the capacity of the State's cyber troops. The main actors behind cyberhoaxes have shown considerable reflectivity in examining Law No. 19 of 2016 about Electronic Information and Transactions, as by doing so they have been able to avoid legal snares. They do not simply accept their websites' blocking by the government, even though they only need to move to another domain. They have studied the blocking mechanisms to best avoid them. As such, the government must act to change those regulations it has enacted. The Indonesian government must transform the regulations applicable to the media platforms used to make hoaxes go viral. Thus far, Indonesia has not provided for any fines for them, relying solely on blocking mechanisms—even though forcing platforms such as Facebook and Google to pay large fines if they fail to remove fake news, hate speech, and hoaxes may serve to limit their spread on social media.

Media industries, meanwhile, must work together with corporations and media platforms. Aside from urging advertising services such as Google Ad Senseto stop providing incentives to domains that contain and propagate hoaxes, they can also urge that the code necessary to detect hoaxes and other deceptive content be enacted. The government can also prepare an agreement regarding the algorithms used to prevent the rise of online enclaves and thus the spread of a tribal nationalism based on political identity.

To face and abate the threat of hoaxes, it is insufficient for the state to collaborate solely with the media industry. Civil society itself, which is targeted by hoaxers, must be involved. The involvement of various communities in combating hoaxes is paramount. This may be done, for example, by promoting digital literacy, so that members of society act can more intelligently and critically in cyberspace.

The government must also work with and accommodate civil troops in its combating hoaxes. Groups such as the Anti-Defamation League of Indonesia (Mafindo), which was established in 2012, are actively working to combat hoaxes and promote the honest dissemination of knowledge throughout Indonesia, both online and offline. Mafindo recognizes that the scale of its activities pales in comparison to that of hoax propagation and proliferation [13], and as such it is urgent to create synergy between state-operated cyber troops and civil troops. As hoaxes become increasingly common online, attempts to counter them must also be intensified. As such, collaboration between the above three elements—the state, industry, and civil society—is paramount.

5 Conclusion

The creation and dissemination of cyberhoaxes in Indonesia is a deliberate practice intended to promote certain motives and interests. It is perpetrated by actors who seek to spread deceit and hate in the digital ecosystem. The proliferation of hoaxes in cyberspace indicates a shift from freedom of speech (facilitated by new media platforms) into freedom to hate, which is used to attack those opposed to them. The websites in this study use similar production patterns. To draw public attention, they mobilize rumors and tribal, religious, and racial sentiments. To popularize their websites, hoaxers use social media and networks to spread fake news and hate speech. The proliferation of hoaxes and hate speech in cyberspace threaten national security and stability. The State, and specifically the Indonesian State Intelligence Agency, should pay serious attention to the risks posed by such cyberhoaxes. Efforts to secure cyberspace require the active participation of not only the state, but also industry and civil society.

Acknowledgments

This study was supported by the Higher School of State Intelligence through Research Initiative Program. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author (s) and do not necessarily reflect the views of the Higher School of State Intelligence.

References

[1] H. Allcott, M. Gentzkow, "Social media and fake news in the 2016 election," *Journal of Economic Perspectives*, vol. 31, no. 2, pp. 211–236, 2017.

[2] L. Arnhart, "Murray edelman, political symbolism, and the incoherence of political science," *Political Science Reviewer*, vol. 15, no. 1, pp. 185–213, 1985.

[3] R. J. Bernstein, *Hannah Arendt and the Jewish Question*, Cambridge: Polity Pres, 1996.
[4] M. D. Cavelty, *Cyber Security and Threat Politics: US Efforts to Secure the Information Age*. London, New York: Routledge, 2008.
[5] P. Dahlgren, *The Political Web: Media, Participation and Alternative Democracy*, New York: Palgrave Macmillan, 2013.
[6] M. Edelman, *The Politics of Misinformation*, Cambridge: Cambridge University Press, 2013.
[7] J. Eriksson, G. Giacomello, *International Relations and Security in the Digital Age*, London, New York: Routledge, 2007.
[8] S. Lash, J. Urry, *Economies of Signs and Spaces*, London, Thousand Oaks, New York: Sage, 1993.
[9] M. Lim, "Many clicks but little sticks: Social media activism in Indonesia," *Journal of Contemporary Asia*, vol. 43, no. 4, pp. 636–657, 2013.
[10] M. Lim, "Freedom to hate: Social media, algorithmic enclaves, and the rise of tribal nationalism in Indonesia," *Critical Asian Studies*, vol. 49, no. 3, pp. 411–427, 2017.
[11] D. Lovell, *Native Advertising: The Essential Guide*, London, New York, New Delhi: Kogan Page, 2017.
[12] T. Majalah, *Wabah Hoax*, 8 January 2017.
[13] S. E. Nugroho, "Upaya masyarakat anti-fitnah Indonesia mengembalikan jatidiri bangsa dengan gerakan anti hoax," in *Proceedings of the National Conference of Young Indonesian Psychology Researchers*, vol. 2, no. 1, pp. 1–4, 2017.
[14] M. V. Rasmussen, "Reflexive security: NATO and international risk society," *Millennium: Journal of International Studies*, vol. 30, no. 2, pp. 285–309, 2001.
[15] M. Sekor, L. Walsh, "A rhetorical perspective on the Sokal hoax: Genre, style, and context," *Written Communication*, vol. 21, no. 1, pp. 69–91, 2004.
[16] P. Virilio, *Speed and Politics*, Los Angeles: Semiotext(e), 2007.

Biography

Budi Gunawan, Ph.D. and Senior Lecturer at the Higher School of State Intelligence, Indonesia. Currently he is the Director of National Intelligence Board of the Republic of Indonesia. His main research interests include Computer Law and Information Security Management.

Barito Mulyo Ratmono, Ph.D. and Associate Director at the Higher School of State Intelligence, Indonesia. His main research interests include Information Security System and Digital Right Management.

Certificateless Ring Signcryption Scheme from Pairings

Hui Guo and Lunzhi Deng
(Corresponding author: Lunzhi Deng)
School of Mathematical Sciences, Guizhou Normal University
Guiyang 550001, China
(Email: denglunzhi@163.com)
(Received June 23, 2018; Revised and Accepted Nov. 22, 2018; First Online June 22, 2019)

Abstract

Signcryption is a useful primitive which simultaneously provides the functions of encryption and signature. Certificateless cryptography not only eliminates the key escrow property, but also removes certificates. In a ring signcryption scheme, an entity can anonymously signcrypt a message on behalf of ring members including himself. In this paper, a new certificateless ring signcryption (CLRSC) scheme is proposed, and it is proved to be secure in the random oracle model. In the scheme, it requires only one bilinear pairing operation in signcryption, and three bilinear pairing operations in unsigncryption. To the best of our knowledge, our scheme is more efficient than previous ones in computation.

Keywords: Certificateless Cryptography; Pairing; Random Oracle Model; Ring Signcryption

1 Introduction

Public key cryptography [16] is an important technique to realize network and information security. Traditional public key infrastructure (PKI) [1,3,8,20] needs a trusted certification authority (CA) to issue a certificate binding the identity and the public key of the user. Hence, the management problem of public key certificates arises. To solve the problem, Shamir [27] defined a idea of identity-based cryptography in 1984. In the identity-based cryptography [14,18], a trusted third party called the private key generator (PKG) generates all user's private keys, which bring a new problem of the key escrow.

In 2003, Al-Riyami *et al.* [2] introduced the concept of certificateless public key cryptography (CL-PKC). In CL-PKC, a user's private key is made up of partial private key generated by key generation center (KGC) [11,19,25] and a secret value selected by the user separately. So even if the malicious KGC leaks the partial private key created by KGC, the attacker also cannot get the entire private key to decrypt the associated ciphertext. Through this, certificateless cryptography not only eliminates the key escrow property, but also removes certificates.

Ring signature was first defined by Rivest *et al.* [23] in 2001. In a ring signature scheme, a signer can select some members to form a ring and produce a ring signature without the assitance of the other ring members. Any verifier can know that the message comes from a member of ring, but doesn't know exactly who the signer is. So it has a lot of important applications for revealing secrets. Some valuable information was found in the study of ring signature [4, 7, 10, 17, 21, 24]. Ring signcryption [15] is a cryptographic primitive motivated by ring signature. In a ring signcryption scheme, a user can anonymously signcrypt a message on behalf of ring members including himself. It is helpful for leaking secrets in an anonymous, authenticated and confidential way.

Huang *et al.* [13] extended ring signature to ring signcryption and proposed a concrete scheme in the identity-based cryptosystem, but the ciphertext of their scheme is too long. In 2009, Zhu *et al.* [33] proposed an efficient and provable secure identity based ring signcryption scheme. But Selvi *et al.* [26] pointed out that the scheme [33] is not semantically secure. Other schemes proposed including generalized ring signcryption [32], attribute-based ring signcryption [9,31], threshold ring signcryption [5], *etc.*

In 2007, Wang *et al.* [30] constructed a certificateless ring signcryption scheme, which is proved to be secure. Their scheme needs 3n+5 pairing operations. Zhu *et al.* [34] proposed a provably secure parallel certificateless ring signcryption scheme, but they did not give the concrete proof about security. In 2011, Qi *et al.* [22] proposed a provably secure certificateless ring signcryption scheme. In 2015, Sharma *et al.* [28] constructed a pairing-free certificateless ring signcryption scheme (PF-CLRSC). However, Shen *et al.* [29] pointed out that the scheme [28] is not secure in 2017.

In this paper, we propose a new certificateless ring signcryption scheme which has the following features:

1) The proposed scheme is proved to be secure in the random oracle model.

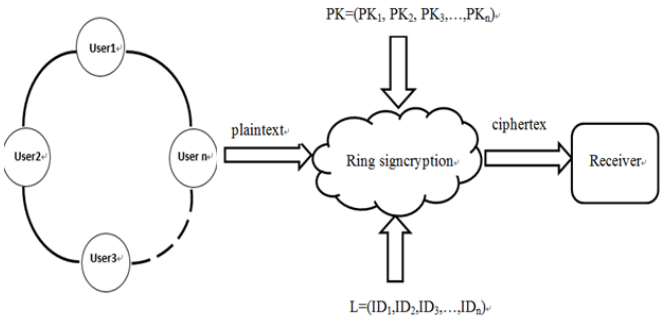


Figure 1: Process of a CLRSC scheme

2) The proposed scheme requires only 4 pairing operations and it is more efficient than the schemes [22,30, 34] in computation.

2 Preliminaries

2.1 Bilinear Pairing

Let G_1 be an additive group of prime order q and G_2 be a multiplicative group of the same order. And P is a generator of G_1. Let $e : G_1 \times G_1 \to G_2$ be a map with the following properties:

- Bilinearity: $e(aP_1, bP_2) = e(P_1, P_2)^{ab}$ for all $P_1, P_2 \in G_1$ and $a, b \in Z_q^*$.
- Non-degeneracy: There exist $P_1, P_2 \in G_1$ such that $e(P_1, P_2) \neq 1_{G_2}$.
- Computability: There is an efficient algorithm to compute $e(P_1, P_2)$ for all $P_1, P_2 \in G_1$.

Definition 1. *Given a generator P of a group G_1 and a tuple $(aP, bP, cP, X \in G_2)$ for unknown $a, b, c \in Z_q^*$, the Decisional Bilinear Diffie-Hellman problem (DBDHP) is to decide whether $X = e(P, P)^{abc}$.*

Definition 2. *Given a generator P of group G_1 and a tuple (aP, bP) for unknown $a, b \in Z_q^*$, the computational Diffie-Hellman problem (CDHP) is to compute abP.*

Definition 3. *Given a generator Q of group G_3 with prime order p, and a tuple $(aQ, bQ, X \in G_3)$ for unknown $a, b \in Z_q^*$, the Decisional Diffie-Hellman problem (DDHP) is to decide whether $X = abQ$.*

Definition 4. *Given a generator Q of group G_3 with prime order p, and an elements aQ, the discrete logarithm problem (DLP) is to compute a.*

2.2 Model of Certificateless Ring Signcryption

A certificateless ring signcryption scheme (CLRSC) is composed of six polynomial time algorithms, it is defined as follows:

- Setup: Input a security parameter ν, KGC outputs the system parameters *params* and a master secret key msk.
- Partial-Private-Key-Extract: Input the system parameters *params*, the master secret key msk and the identity $ID_i \in \{0,1\}^*$, KGC returns the user's partial private key D_i.
- Secret-Value-Set: The user ID_i randomly chooses a secret value $t_i \in Z_q^*$.
- User-Public-Key-Generate: Input the system parameters *params*, the user's secret value t_i and identity $ID_i \in \{0,1\}^*$, this algorithm outputs the public key T_i. It is run by user himself.
- Signcryption: To send the message m to the receiver ID_r, the actual signcrypter ID_s selects $n-1$ other users to form n users ring L including himself and represents members of the ring L to give a ciphertext σ on the message m.
- Unsigncryption: After receiving the ciphertext (σ, L), the receiver ID_r decrypts the ciphertext and obtains the message m or the symbol $\perp$ if σ was a invalid ciphertext.

Definition 5.

A CLRSC scheme is said to be indistinguishable under adaptive chosen ciphertext attacks (IND-CLRSC- CCA2) if the polynomial bounded adversary with a negligible advantage in the following game.

Game I. A challenger $\mathscr{C}$ and a Type I adversary $\mathscr{A}_1$ play the following game.

Initialization. $\mathscr{C}$ runs the setup algorithm to generate a master secret key msk and the public system parameters *params*. $\mathscr{C}$ sends *params* to $\mathscr{A}_1$. ($\mathscr{A}_1$ does not know msk).

Phase 1. $\mathscr{A}_1$ makes a polynomially bounded number of adaptive queries to $\mathscr{C}$.

- Hash functions query: $\mathscr{A}_1$ can query the values of any hash functions.
- Partial private key query: $\mathscr{A}_1$ chooses a user's identity ID_i, $\mathscr{C}$ runs this algorithm to generate the corresponding partial private key D_i, and sends to $\mathscr{A}_1$.
- User public key query: $\mathscr{A}_1$ chooses an identity ID_i, $\mathscr{C}$ returns public key T_i generated by the public key algorithm.
- User public key replacement: $\mathscr{A}_1$ chooses an identity ID_i and a new public key value T_i', $\mathscr{A}_1$ replaces the current public key T_i of the user ID_i with T_i'.
- Secret value query: $\mathscr{A}_1$ chooses an identity ID_i, $\mathscr{C}$ returns the corresponding secret value t_i to $\mathscr{A}_1$. If public key of the user ID_i was replaced, $\mathscr{A}_1$ cannot ask for the secret value of the user ID_i.

- Signcryption query: $\mathscr{A}_1$ chooses a message m, a receiver ID_r and a set $R = L \bigcup \{T_i : ID_i \in L\}$, where $L = \{ID_1, \cdots, ID_n\}$ is the set of n users' identities, and sends to $\mathscr{C}$. $\mathscr{C}$ returns the ciphtext σ to $\mathscr{A}_1$.
- Unsigncryption query: When $\mathscr{A}_1$ chooses a ciphertext σ, a receiver's identity ID_r and a set $L = \{ID_1, \cdots, ID_n\}$, $\mathscr{C}$ outputs plaintext m or the symbol $\perp$ if σ is an invalid ciphertext.

Challenge. $\mathscr{A}_1$ sends following information to the challenger: two equal length messages m_0, m_1, a specified receiver ID_r, a set $R = L \bigcup \{T_i : ID_i \in L\}$, where $L = \{ID_1, \cdots, ID_n\}$ is the set of n users, and fulfills the following conditions:

1) $\mathscr{A}_1$ should not have queried the partial private key to ID_r in Phase 1.
2) There exists at least a member $ID_s \in L$ whose public key has not been replaced by $\mathscr{A}_1$.

$\mathscr{C}$ takes randomly a bit $\mu \in \{0, 1\}$ and computes the ciphertext σ^* on the message m_μ under the set R.

Phase 2. $\mathscr{A}_1$ performs a polynomially bounded number of queries just like in Phase 1, and fulfills the following restrictions:

1) $\mathscr{A}_1$ can not have requested the partial private key for ID_r.
2) $\mathscr{A}_1$ can not have made the unsigncryption queries for the ciphertext σ^*.

Response. $\mathscr{A}_1$ outputs a bit μ' and wins the game if $\mu' = \mu$.

The advantage of $\mathscr{A}_1$ is defined as : $Adv_{\mathscr{A}_1}^{IND-CLRSC}(\nu) = |2\Pr[\mu' = \mu] - 1|$.

Game II. A Type II adversary $\mathscr{A}_2$ for a CLRSC scheme plays the following game with a challenger $\mathscr{C}$.

Initialization. $\mathscr{C}$ runs the setup algorithm to generate the master secret key msk and public system parameters $params$, then sends $params$ and msk to $\mathscr{A}_2$.

Phase 1. Same as that in the Game I.

Challenge. $\mathscr{A}_2$ sends following information to the challenger: two equal length messages m_0, m_1, a specified receiver ID_r and a set $R = L \bigcup \{T_i : ID_i \in L\}$, where $L = \{ID_1, \cdots, ID_n\}$ is the set of n users, and fulfills the following restrictions:

1) $\mathscr{A}_2$ can not have requested the secret value for ID_r in Phase 1.
2) $\mathscr{A}_2$ can not have replaced the user public key corresponding to ID_r in Phase 1.
3) There exists at least a member $ID_s \in L$ whose public key has not been replaced by $\mathscr{A}_2$.

$\mathscr{C}$ takes randomly a bit $\mu \in \{0, 1\}$ and computes the ciphertext σ^* on m_μ under the set R.

Phase 2. $\mathscr{A}_2$ performs a polynomially bounded number of queries just like in Phase 1, and fulfills the following conditions:

1) $\mathscr{A}_2$ can not have requested the secret value for ID_r.
2) $\mathscr{A}_2$ can not have made the unsigncryption queries for the ciphertext σ^*.

Response. $\mathscr{A}_2$ outputs a bit μ' and wins the game if $\mu' = \mu$.

The advantage of $\mathscr{A}_2$ is defined as: $Adv_{\mathscr{A}_2}^{IND-CLRSC}(\nu) = |2\Pr[\mu' = \mu] - 1|$.

Definition 6. *CLRSC is said to be unforgeable under adaptive chosen message attacks(EUF-CLRSC-CMA2) if the polynomial bounded adversary with a negligible advantage in the following game.*

Game III. Challenger $\mathscr{C}$ and type I adversary $\mathscr{A}_1$ play the following game:

Initialization, Query. Same as that in the Game I.

Forge. $\mathscr{A}_1$ produces a new ciphertext (σ, ID_r, R).

When the following conditions hold, $\mathscr{A}_1$ wins the game.

1) The symbol $\perp$ is not returned by unsigncryption query.
2) $\mathscr{A}_1$ cannot ask for the partial private keys of the users in L.
3) The forged ciphertext (σ, ID_r, R) is not obtained by signcryption query.

The advantage of $\mathscr{A}_1$ is defined as: $Adv_{\mathscr{A}_1}^{UNF-CLRSC} = \Pr[\mathscr{A}_1 \ win]$.

Game IV. Challenger $\mathscr{C}$ and type II adversary $\mathscr{A}_2$ play the following game:

Initialization, Query. Same as that in the Game II.

Forge. $\mathscr{A}_2$ produces a new ciphertext (σ, ID_r, R). When the following conditions hold, $\mathscr{A}_2$ wins the game.

1) The symbol $\perp$ is not returned by unsigncryption query.
2) $\mathscr{A}_2$ can not request the secret value of the users in L and replace the user public key of the members in L.
3) The forged ciphertext (σ, ID_r, R) is not obtained by signcryption query.

The advantage of $\mathscr{A}_2$ is defined as : $Adv_{\mathscr{A}_2}^{UNF-CLRSC} = \Pr[\mathscr{A}_2 \ win]$.

Definition 7. *A CLRSC scheme is anonymous if for any message m, any ring $L = \{ID_1, \cdots, ID_n\}$, receiver ID_r and ciphertext σ. The receiver ID_r ($ID_r \notin L$), even with unbounded computing resources, can identify the actual signcrypter with probability no better than $\frac{1}{n}$.*

3 Proposed Scheme

- Setup: Given the security parameter of the system ν, KGC chooses groups $G_1 = \langle P \rangle$, G_2 and $G_3 = \langle Q \rangle$ of prime order $q > 2^\nu$, and a bilinear pairing $e : G_1 \times G_1 \to G_2$. Then KGC chooses four hash function $H_1 : \{0,1\}^* \to G_1$, $H_2 : \{0,1\}^* \to \{0,1\}^l$, $H_3, H_4 : \{0,1\}^* \to Z_q^*$. The message space is $\Omega = \{0,1\}^l$. KGC randomly chooses its secret key $x \in Z_q^*$ and sets $P_{pub} = xP$ as its system public key. KGC publishes system parameters : $params = \{G_1, G_2, G_3, q, e, P, Q, P_{pub} = xP, H_1, H_2, H_3, H_4\}$.
- Partial-Private-Key-Extract: Given a user's identity $ID_i \in \{0,1\}^*$, KGC computes $E_i = H_1(ID_i), D_i = xE_i$ and sends D_i to the user via a secure channel.
- Secret value set: The user ID_i selects at random $t_i \in Z_q^*$ as his/her secret value.
- User public key generate: The user ID_i sets $T_i = t_iQ$ as his/her public key.
- Signcryption: Let $R = L \bigcup \{T_i, ID_i \in L\}$, where $L = \{ID_1, \cdots, ID_n\}$ is the set of n users' identities. The actual signcrypter $ID_s \in L$ outputs a ciphertext σ on the message m and sends it to the receiver ID_r as following:
  1) Randomly selects $\lambda_1, \lambda_2 \in Z_q^*$, computes $B_1 = \lambda_1 P$, $B_2 = \lambda_2 Q$, $U_1 = e(\lambda_1 P_{pub}, E_r)$, $U_2 = \lambda_2 T_r$, $C = H_2(R, U_1, U_2) \bigoplus m$.
  2) Randomly selects $A_i \in G_1, c_i \in Z_q^*$, computes $h_i = H_3(m, R, U_1, U_2, T_i, ID_i, A_i, c_i)$, $i = 1, 2, \cdots, s-1, s+1, \cdots, n$.
  3) Randomly selects $\delta_1 \in Z_q^*$, computes $A_s = \delta_1 E_s - \sum_{i=1, i\neq s}^{n}(A_i + h_i E_i)$.
  4) Randomly selects $\delta_2 \in Z_q^*$, computes $y = H_4(m, R, U_1, U_2, \delta_2 Q + \sum_{i=1, i\neq s}^{n} c_i T_i, \bigcup_{i=1}^{n}\{A_i\})$.
  5) Computes $c_s = y - \sum_{i=1, i\neq s}^{n} c_i \pmod q$, $h_s = H_3(m, R, U_1, U_2, T_s, ID_s, A_s, c_s)$.
  6) Computes $z = \delta_2 - c_s t_s \pmod q$, $V = (\delta_1 + h_s) D_s$.
  7) Outputs the ciphertext : $\sigma = \{z, V, B_1, B_2, C, \bigcup_{i=1}^{n}\{A_i\}, \bigcup_{i=1}^{n}\{c_i\}\}$.
- Unsigncryption: On receiving the ciphertext $\sigma = \{z, V, B_1, B_2, C, \bigcup_{i=1}^{n}\{A_i\}, \bigcup_{i=1}^{n}\{c_i\}\}$, the receiver ID_r decrypts the ciphertext as follows:
  1) Computes $U_1 = e(B_1, D_r)$, $U_2 = t_r B_2$, $m = C \bigoplus H_2(R, U_1, U_2)$.
  2) Checks if $\sum_{i=1}^{n} c_i = H_4(m, R, U_1, U_2, zQ + \sum_{i=1}^{n} c_i T_i, \bigcup_{i=1}^{n}\{A_i\})$. Proceed if the equality holds, reject otherwise.
  3) Computes $h_i = H_3(m, R, U_1, U_2, T_i, ID_i, A_i, c_i)$, $i = 1, 2, \cdots, n$.
  4) Checking whether $e(P, V) = e(P_{pub}, \sum_{i=1}^{n}(A_i + h_i E_i))$. If the equality holds, accepts m as a valid message. Otherwise, it returns $\perp$

4 Analysis of Proposed Scheme

4.1 Correctness Analysis

$$
\begin{aligned}
e(P,V) &= e(P,(\delta_1 + h_s)D_s) \\
&= e(P,(\delta_1 + h_s)xE_s) \\
&= e(xP,(\delta_1 + h_s)E_s) \\
&= e(P_{pub}, \delta_1 E_s + h_s E_s) \\
&= e(P_{pub}, A_i + \sum_{i=1, i\neq s}^{n}(A_i + h_i E_i) + h_s E_s) \\
&= e(P_{pub}, \sum_{i=1}^{n}(A_i + h_i E_i)); \\
U_2 &= t_r B_2 = t_r \lambda_2 Q = \lambda_2 t_r Q = \lambda_2 T_r; \\
U_1 &= e(B_1, D_r) \\
&= e(\lambda_1 P, xE_r) \\
&= e(\lambda_1 xP, E_r) \\
&= e(\lambda_1 P_{pub}, E_r); \\
\sum_{i=1}^{n} c_i &= y \\
&= H_4(m, R, U_1, U_2, zQ \\
&\quad + \sum_{i=1}^{n} c_i T_i, \bigcup_{i=1}^{n}\{A_i\}); \\
\delta_2 Q + \sum_{i=1, i\neq s}^{n} c_i T_i &= (z + c_s t_s)Q + \sum_{i=1, i\neq s}^{n} c_i T_i \\
&= zQ + c_s T_s + \sum_{i=1, i\neq s}^{n} c_i T_i \\
&= zQ + \sum_{i=1}^{n} c_i T_i.
\end{aligned}
$$

4.2 Security Analysis

Theorem 1. *In random oracle model, the scheme is indistinguishable against IND-CLRSC-CCA2 adversary $\mathscr{A}_1$ if the DBDHP is hard.*

Proof. Assume that the challenger $\mathscr{C}$ receives an instance (P, aP, bP, cP, X) of the DBDHP, the goal of $\mathscr{C}$ is to determine whether $X = e(P,P)^{abc}$ or not. $\mathscr{C}$ runs $\mathscr{A}_1$ as a subroutine and plays the role of the challenger in Game I.

Initialization. $\mathscr{C}$ runs the setup algorithm to generate system parameters. Then $\mathscr{C}$ sends the system parameters $params = \{G_1, G_2, G_3, q, e, P, Q, P_{pub} = aP, H_1, H_2, H_3, H_4\}$ to $\mathscr{A}_1$. ($\mathscr{A}_1$ does not know the value a).

Phase 1.Without losing generality, assuming that each query is different. $\mathscr{A}_1$ will ask for $H_1(ID_i)$ before the identity ID_i is used in any other queries. $\mathscr{C}$ will maintain some lists to store the queries and answers, all of the lists are initially empty.

- H_1 queries: $\mathscr{C}$ maintains the list L_1 of tuple (ID_i, d_i). When $H_1(ID_i)$ is queried by $\mathscr{A}_1$, $\mathscr{C}$ answers the query H_1 as follows.

  At the j^{th} H_1 query, $\mathscr{C}$ sets $H_1(ID^*) = bP$. For $i \neq j$, $\mathscr{C}$ selects a random $d_i \in Z_q^*$ and sets $H_1(ID_i) = d_iP$, the query and the respond will be stored in the list L_1.

- H_2 queries: $\mathscr{C}$ maintains the list L_2 of tuple (α_i, h_i). When $H_2(\alpha_i)$ is queried by $\mathscr{A}_1$, $\mathscr{C}$ selects a random $h_i \in \{0,1\}^l$, sets $H_2(\alpha_i) = h_i$ and adds (α_i, h_i) to list L_2.

- H_3 queries: $\mathscr{C}$ maintains the list L_3 of tuple (β_i, c_i). When $H_3(\alpha_i)$ is queried by $\mathscr{A}_1$, $\mathscr{C}$ selects a random $c_i \in Z_q^*$, sets $H_3(\beta_i) = c_i$ and adds (β_i, c_i) to list L_3.

- H_4 queries: $\mathscr{C}$ maintains the list L_4 of tuple $(\beta_i^{'}, c_i^{'})$. When $H_4(\alpha_i)$ is queried by $\mathscr{A}_1$, $\mathscr{C}$ selects a random $c_i^{'} \in Z_q^*$, sets $H_4(\beta_i^{'}) = c_i^{'}$ and adds $(\beta_i^{'}, c_i^{'})$ to list L_4.

- User public key queries: $\mathscr{C}$ maintains the list L_U of tuple (ID_i, t_i). When $\mathscr{A}_1$ makes this query, $\mathscr{C}$ picks a random $t_i \in Z_q^*$, sets $T_i = t_iQ$ and adds (ID_i, t_i) to list L_U.

- User public key replacement requests: $\mathscr{C}$ maintains the list L_R of tuple (ID_i, T_i, T_i'). When $\mathscr{A}_1$ makes this query, $\mathscr{C}$ replaces the current public key value T_i with a new value T_i' and adds (ID_i, T_i, T_i') to list L_R.

- Partial private key queries: $\mathscr{C}$ maintains the list L_D of tuple (ID_i, D_i). When $\mathscr{A}_1$ makes this query, $\mathscr{C}$ does as follows:

  If $ID_i = ID^*$, $\mathscr{C}$ fails and stops. Otherwise $\mathscr{C}$ looks up the tuple (ID_i, d_i) in list L_1, responds with $D_i = d_i \cdot (aP)$ and adds (ID_i, D_i) to list L_D.

- Secret value queries: $\mathscr{C}$ maintains the list L_E of tuple (ID_i, t_i). When $\mathscr{A}_1$ makes this query, $\mathscr{C}$ checks list L_U. If there exists the tuple (ID_i, t_i) in list L_U, $\mathscr{C}$ answers with t_i. Otherwise, $\mathscr{C}$ selects a random $t_i \in Z_q^*$, answers with t_i and adds (ID_i, t_i) to lists L_E and L_U.

- Signcryption queries: $\mathscr{A}_1$ selects a message m, a set $R = L \bigcup \{T_i : ID_i \in L\}$, where $L = \{ID_1, \cdots, ID_n\}$ is the set of n users' identities and a receiver ID_r and sends them to $\mathscr{C}$. $\mathscr{C}$ returns a signcryption as follows:

  If there exists an identity $ID_s \in L$ such that $ID_s \neq ID^*$ and $ID_s \notin L_R$, $\mathscr{C}$ gives a signcryption σ by calling the signcryption algorithm to answer $\mathscr{A}_1$, where ID_s is the actual signer. Otherwise, $\mathscr{C}$ does the following steps:

  1) Randomly selects $\lambda_1, \lambda_2 \in Z_q^*$, computes $B_1 = \lambda_1 P$, $B_2 = \lambda_2 Q$, $U_1 = e(\lambda_1 P_{pub}, E_r)$, $U_2 = \lambda_2 T_r$, $C = H_2(R, U_1, U_2) \bigoplus m$.
  2) Randomly selects $A_i \in G_1, c_i \in Z_q^*$, computes $h_i = H_3(m, R, U_1, U_2, T_i, ID_i, A_i, c_i)$, $i = 1, 2, s-1, s+1, \cdots, n$.
  3) Randomly selects $z, c_s \in Z_q^*$, computes $T = zQ + \sum_{i=1}^{n} c_i T_i$.
  4) Randomly selects $r, h_s \in Z_q^*$, computes $A_s = rP - h_s E_s - \sum_{i=1, i\neq s}^{n} (A_i + h_i E_i)$, $V = r(aP)$.
  5) Stores the relations: $\sum_{i=1}^{n} c_i = H_4(m, R, U_1, U_2, T, \bigcup_{i=1}^{n}\{A_i\})$, $h_s = H_3(m, R, U_1, U_2, T_s, ID_s, A_s, c_s)$.

     If collision occurs, repeats Steps (1)-(5).
  6) Outputs the ciphertext: $\sigma \doteq \{z, V, B_1, B_2, C, \bigcup_{i=1}^{n}\{A_i\}, \bigcup_{i=1}^{n}\{c_i\}\}$.

- Unsigncryption queries: $\mathscr{A}_1$ picks ciphertext $\sigma = \{z, V, B_1, B_2, C, \bigcup_{i=1}^{n}\{A_i\}, \bigcup_{i=1}^{n}\{c_i\}\}$, a set $R = L \bigcup \{T_i : ID_i \in L\}$ and a receiver ID_r. If $ID_r \neq ID^*$ and $ID_r \notin L_R$, $\mathscr{C}$ gives a message m by calling the unsigncryption algorithm. Otherwise, $\mathscr{C}$ notifies that σ is an invaild ciphertext.

Challenge. $\mathscr{A}_1$ chooses two equal length messages m_0, m_1, a specified receiver ID_r, and a set $R = L \bigcup \{T_i : ID_i \in L\}$, where $L = \{ID_1, \cdots, ID_n\}$ is the set of ring members, and sends them to the challenger $\mathscr{C}$. ($\mathscr{A}_1$ should not have queried the partial private key for ID_r in Phase 1). If $ID_r \neq ID^*$, $\mathscr{C}$ fails and stops. Otherwise, $\mathscr{C}$ picks $\mu \in \{0,1\}$, and computes ciphertext σ^* on the message M_μ under the set R as follows:

1) Randomly selects $c, \lambda_2 \in Z_q^*$, computes $B_1 = cP$, $B_2 = \lambda_2 Q$, $U_1 = X$, $U_2 = \lambda_2 T_r$, $C = H_2(R, X, U_2) \bigoplus m$.
2) Randomly selects $A_i \in G_1, c_i \in Z_q^*$, computes $h_i = H_3(m, R, U_1, U_2, T_i, ID_i, A_i, c_i)$, $i = 1, 2, \cdots, s-1, s+1, \cdots, n$.
3) Randomly selects $\delta_1 \in Z_q^*$, computes $A_s = \delta_1 E_s - \sum_{i=1, i\neq s}^{n} (A_i + h_i E_i)$.
4) Randomly selects $\delta_2 \in Z_q^*$, computes $y = H_4(m, R, U_1, U_2, \delta_2 Q + \sum_{i=1, i\neq s}^{n} c_i T_i, \bigcup_{i=1}^{n}\{A_i\})$.
5) Computes $c_s = y - \sum_{i=1, i\neq s}^{n} c_i \pmod q$. $h_s = H_3(m, R, U_1, U_2, T_s, ID_s, A_s, c_s)$.
6) Computes $z = \delta_2 - c_s t_s \pmod q$, $V = (\delta_1 + h_s) D_s$.
7) Outputs the ciphertext: $\sigma^* = \{z, V, B_1, B_2, C, \bigcup_{i=1}^{n}\{A_i\}, \bigcup_{i=1}^{n}\{c_i\}\}$.

Phase 2. $\mathscr{A}_1$ makes a polynomially bounded number of queries just like in the Phase 1 (but $\mathscr{A}_1$ should not have queried the partial private key for ID_r and requested the plaintext corresponding to the ciphertext σ^*).

Response. $\mathscr{A}_1$ outputs $\mu' \in \{0,1\}$. If $\mu' \doteq \mu$, $\mathscr{C}$ outputs 1. Otherwise, $\mathscr{C}$ outputs 0. If $X = e(P,P)^{abc}$, σ^* is a valid ciphertext. Then $\mathscr{A}_1$ can distinguish μ with the advantage ε. So $\Pr[\mathscr{C} \longrightarrow 1|X \doteq e(P,P)^{abc}] \doteq \Pr[\mu^{'} \doteq \mu|X \doteq e(P,P)^{abc}] \doteq \frac{1}{2} + \varepsilon$.

If $X \neq e(P,P)^{abc}$, when $\mu = 0$ or $\mu = 1$, each part of the ciphertext has the same probability distribution, so $\mathscr{A}_1$ has no advantage to distinguishing μ. So

$\Pr[\mathscr{C} \longrightarrow 1|X \neq e(P,P)^{abc}] \doteq \Pr[\mu^{'} \doteq \mu|X \neq e(P,P)^{abc}] \doteq \frac{1}{2}$.

Probability. Let $q_{H_i}(i = 1,2,3,4)$, q_U, q_R, q_D and q_S be the number of $H_i(i = 1,2,3,4)$ queries, user public key queries, user public key replacement requests, partial private key queries and signcryption queries, respectively.

Without loss of generality, we may assume that $L_E \cap L_R = \emptyset$, and denote some events as follows: π_1: $\mathscr{C}$ does not fail in partial private key queries; π_2: $\mathscr{C}$ does not fail in unsigncryption queries; π_3: $\mathscr{C}$ does not fail in challenge stage. It is easy to get following results:
$\Pr[\pi_1]=1-\frac{q_D}{q_{H_1}}$, $\Pr[\pi_2]=1-\frac{q_U}{2^\nu}$, $\Pr[\pi_3]=\frac{1}{q_{H_1}-q_D}$.

$$\begin{aligned}\Pr[\mathscr{C}\ success] &= \Pr[\pi_1 \wedge \pi_2 \wedge \pi_3] \\ &= \Pr[\pi_1] \cdot \Pr[\pi_2] \cdot \Pr[\pi_3] \\ &= (1-\frac{q_D}{q_{H_1}}) \cdot (1-\frac{q_U}{2^\nu}) \cdot \frac{1}{q_{H_1}-q_D} \\ &\approx \frac{1}{q_{H_1}}\end{aligned}$$

Therefore, if $\mathscr{A}_2$ can succeed with the probability ε, then $\mathscr{C}$ can solve the DBDHP with probability $\frac{\varepsilon}{q_{H_1}}$. □

Theorem 2. *In the random oracle model, the scheme is indistinguishable against IND-CLRSC-CCA2 adversary $\mathscr{A}_2$ if the DDHP is hard.*

Proof. Assume that the challenger $\mathscr{C}$ receives an instance (aQ, bQ, Y) of the DDHP, the goal of $\mathscr{C}$ is to determine whether $Y = abQ$ or not. $\mathscr{C}$ runs $\mathscr{A}_2$ as a subroutine and plays the role of the challenger in Game II.

Initialization. $\mathscr{C}$ performs the setup algorithm with the parameter ν, then sends the system parameters $params = \{G_1, G_2, G_3, q, e, P, Q, P_{pub} = xP, H_1, H_2, H_3, H_4\}$ and master secret key $msk = \{x\}$ to $\mathscr{A}_2$.

Phase 1. Without losing generality, assuming that each query is different. $\mathscr{A}_1$ will ask for $H_1(ID_i)$ before the identity ID_i is used in any other queries. $\mathscr{C}$ will maintain some lists to store the queries and answers, all of the lists are initially empty.

- H_1 queries: $\mathscr{C}$ maintains the list L_1 of tuple (ID_i, d_i). When $\mathscr{A}_2$ makes a query $H_1(ID_i)$, $\mathscr{C}$ randomly picks $d_i \in Z_q^*$, sets $H_1(ID_i) = d_iP$ and adds (ID_i, d_i) to list L_1.
- H_2, H_3 and H_4 queries: Same as those in the proof of Theorem 1.
- User public key queries: $\mathscr{C}$ maintains the list L_U of tuple (ID_i, t_i). When $\mathscr{A}_2$ makes this query, $\mathscr{C}$ responds as follows:

  At the j^{th} query, $\mathscr{C}$ sets $ID_j = ID^*$, $T^* = aQ$. For $i \neq j$, $\mathscr{C}$ randomly picks $t_i \in Z_q^*$, returns $T_i = t_iQ$ and adds (ID_i, t_i) to list L_U.
- User public key replacement requests: Same as that in the proof of Theorem 1.
- Partial private key queries: $\mathscr{C}$ maintains the list L_D of tuple (ID_i, D_i). When $\mathscr{A}_2$ makes this query, $\mathscr{C}$ finds the tuple (ID_i, d_i) in list L_1, responds with $D_i = d_i(xP)$ and adds (ID_i, D_i) to list L_D.
- Secret value queries: $\mathscr{C}$ maintains the list L_E of tuple (ID_i, t_i). When $\mathscr{A}_2$ makes this query, $\mathscr{C}$ does as follows:

  If $ID_i = ID^*$, $\mathscr{C}$ fails and stops. Otherwise, $\mathscr{C}$ looks up (ID_i, t_i) in list L_U, responds with t_i and adds (ID_i, t_i) to list L_E.
- Signcryption, Unsigncryption queries: Same as that in the proof of Theorem 1.

Challenge. $\mathscr{A}_2$ chooses two equal length messages m_0, m_1, and a specified receiver ID_r, a set $R = L \bigcup \{T_i : ID_i \in L\}$, where $L = \{ID_1, \cdots, ID_n\}$ is the set of n ring members, and sends them to the challenger $\mathscr{C}$.($\mathscr{A}_2$ should not have queried the secret value for ID_r). if $ID_r \neq ID^*$, $\mathscr{C}$ fails and stops. Otherwise, $\mathscr{C}$ picks $\mu \in \{0,1\}$, and computes ciphertext σ^* on message M_μ under the set R as follows:

1) Randomly chooses $\lambda_1, b \in Z_q^*$, computes $B_1 = \lambda_1 P$, $B_2 = bQ$, $U_1 = e(\lambda_1 P_{pub}, E_r)$, $U_2 = Y$, $C = H_2(R, U_1, Y) \bigoplus m$.
2) Randomly chooses $A_i \in G_1, c_i \in Z_q^*$, computes $h_i = H_3(m, R, U_1, U_2, T_i, ID_i, A_i, c_i)$, $i = 1, 2, \cdots, s-1, s+1, \cdots, n$.
3) Randomly chooses $\delta_1 \in Z_q^*$, computes $A_s = \delta_1 E_s - \sum_{i=1, i\neq s}^{n}(A_i + h_i E_i)$.
4) Randomly chooses $\delta_2 \in Z_q^*$, computes $y = H_4(m, R, U_1, U_2, \delta_2 Q + \sum_{i=1, i\neq s}^{n} c_i T_i, \bigcup_{i=1}^{n}\{A_i\})$.
5) Computes $c_s = y - \sum_{i=1, i\neq s}^{n} c_i \pmod q$, $h_s = H_3(m, R, U_1, U_2, T_s, ID_s, A_s, c_s)$.
6) Computes $z = \delta_2 - c_s t_s \pmod q$, $V = (\delta_1 + h_s) D_s$.

7) Outputs the ciphertext:
$\sigma = \{z, V, B_1, B_2, C, \bigcup_{i=1}^{n}\{A_i\}, \bigcup_{i=1}^{n}\{c_i\}\}$.

Phase 2. $\mathscr{A}_2$ performs a polynomially bounded number of queries just like in Phase 1. ($\mathscr{A}_2$ should not have queried the secret value for ID_r and requested the plaintext corresponding to the ciphertext σ^*).

Response. $\mathscr{A}_2$ outputs $\mu' \in \{0,1\}$. If $\mu' \doteq \mu$, $\mathscr{C}$ outputs 1. Otherwise, $\mathscr{C}$ outputs 0. If $Y = abQ$, σ^* is a valid ciphertext. Then $\mathscr{A}_2$ distinguishes μ with the advantage ε. So

$$\Pr[\mathscr{C} \longrightarrow 1 | Y = abQ] = \Pr[\mu' \doteq \mu | Y = abQ] = \tfrac{1}{2} + \varepsilon.$$

If $Y \neq abQ$, when $\mu = 0$ or $\mu = 1$, each part of the ciphertext has the same probability distribution, so $\mathscr{A}_2$ has no advantage to distinguishing μ. So

$$\Pr[\mathscr{C} \longrightarrow 1 | Y \neq abQ] = \Pr[\mu' \doteq \mu | Y \neq abQ] = \tfrac{1}{2}.$$

Probability. Let $q_{H_i}(i = 1,2,3,4)$, q_U, q_R, q_D and q_S be the number of $H_i(i = 1,2,3,4)$ queries, user public key queries, user public key replacement requests, partial private key queries and signcryption queries, respectively.

Without loss of generality, we may assume that $L_E \cap L_R = \emptyset$, and denote some events as follows: π_1: $\mathscr{C}$ does not fail in secret value queries; π_2: $\mathscr{C}$ does not fail in unsigncryption queries; π_3: $\mathscr{C}$ does not fail in challenge stage. It is easy to get following results:

$\Pr[\pi_1]=1-\frac{q_T}{q_Q}$, $\Pr[\pi_2]=1-\frac{q_U}{2^\nu}$, $\Pr[\pi_3]=\frac{1}{q_Q-q_T}$.

$$\begin{aligned}\Pr[\mathscr{C}\ success] &= \Pr[\pi_1 \wedge \pi_2 \wedge \pi_3] \\ &= \Pr[\pi_1]\cdot\Pr[\pi_2]\cdot\Pr[\pi_3] \\ &= (1-\frac{q_T}{q_Q})\cdot(1-\frac{q_U}{2^\nu})\cdot\frac{1}{q_Q-q_T} \\ &\approx \frac{1}{q_Q}.\end{aligned}$$

Therefore, if $\mathscr{A}_2$ can succeed with the probability ε, then $\mathscr{C}$ can solve the DDHP with probability $\frac{\varepsilon}{q_Q}$. □

Theorem 3. *In random oracle model, the scheme is unforgeable against EUF-CLRSC-CMA2 adversary $\mathscr{A}_1$ if the CDHP is hard.*

Proof. Assume that the challenger $\mathscr{C}$ receives an instance (P, aP, bP) of the CDHP. The goal of $\mathscr{C}$ is to compute the value of abP. $\mathscr{C}$ will run $\mathscr{A}_1$ as a subroutine and play the role of challenger in Game III.

Initialization, Phase 1. Same as that in the Theorem 1.

Forge. $\mathscr{A}_1$ outputs a forged signcryption $\sigma = \{z, V, \bigcup_{i=1}^{n}\{A_i\}, \bigcup_{i=1}^{n}\{c_i\}\}$ on message m^* under the set $R = L \bigcup \{P_i : P_i \in L\}$, and fulfills the requirements as defined in Game III.

Solve CDHP. Using the forking lemma for ring signature schemes [6], after replays $\mathscr{A}_1$ with the same random tape except the λ^{th} result returned by H_2 query of the forged message, $\mathscr{C}$ gets two valid ring signncryptions with probability $\frac{\varepsilon^2}{66C_{q_{H_2}}^n}$: $\{z, V, \bigcup_{i=1}^{n}\{A_i\}, \bigcup_{i=1}^{n}\{c_i\}\}$ and $\{z, V', \bigcup_{i=1}^{n}\{A_i\}, \bigcup_{i=1}^{n}\{c_i\}\}$. It follows that $h_\lambda \neq h'_\lambda$ and $h_i = h'_i$ for $i \neq \lambda$. If ID^* is the actual signer and $\lambda = s$, then $V = (r_1 + h_s)abP$ and $V' = (r_1 + h'_s)abP$, $\mathscr{C}$ solves CDHP by computing: $abP = (h'_s - h_s)^{-1}(V' - V)$.

Probability. Let $q_{H_i}(i = 1,2,3,4)$, q_U, q_D and q_S be the number of $H_i(i = 1,2,3)$ queries, user public key queries, partial private key queries and signcryption queries, respectively.

We denote some events as follows: π_1: $\mathscr{C}$ does not fail during the queries; π_2: $ID^* \in L$; π_3: ID^* is the actual signer; π_4: $\lambda = s$. It is easy to get following results:

$$\begin{aligned}\Pr[\pi_1] &= \frac{q_{H_1} - q_D}{q_{H_1}}, \\ Pr[\pi_2|\pi_1] &= \frac{n}{q_{H_1} - q_D}, \\ Pr[\pi_3|\pi_1 \wedge \pi_2] &= \frac{1}{n}, \\ Pr[\pi_4|\pi_1 \wedge \pi_2 \wedge \pi_3] &= \frac{1}{n}.\end{aligned}$$

$$\begin{aligned}\Pr[\mathscr{C}\ success] &= \Pr[\pi_1 \wedge \pi_2 \wedge \pi_3 \wedge \pi_4] \\ &= \Pr[\pi_1]\cdot Pr[\pi_2|\pi_1]\cdot\Pr[\pi_3|\pi_1 \wedge \pi_2] \\ &\quad \cdot\Pr[\pi_4|\pi_1 \wedge \pi_2 \wedge \pi_3] \\ &= \frac{q_{H_1} - q_D}{q_{H_1}}\cdot\frac{n}{q_{H_1} - q_D}\cdot\frac{1}{n}\cdot\frac{1}{n} \\ &= \frac{1}{n \cdot q_{H_1}}\end{aligned}$$

Therefore, if $\mathscr{A}_1$ can succeed with the probability ε, then $\mathscr{C}$ can solve CDHP with the probability $\frac{\varepsilon^2}{66C_{q_{H_3}}^n} \cdot \frac{1}{n \cdot q_{H_1}}$. □

Theorem 4. *In random oracle model, the scheme is unforgeable against the Type II adversary if the DLP is hard.*

Proof. Assume that the challenger $\mathscr{C}$ receives an instance (P, aP) of the DLP and the goal of $\mathscr{C}$ is to compute the value of a. $\mathscr{C}$ will run $\mathscr{A}_2$ as a subroutine and play the role of challenger in the Game IV.

Initialization, Phase 1. Same as that in the Theorem 2.

Forge. $\mathscr{A}_2$ outputs a forged signcryption $\sigma = \{z, V, \bigcup_{i=1}^{n}\{A_i\}, \bigcup_{i=1}^{n}\{c_i\}\}$ on message m^* under the set $R = L \bigcup \{P_i : P_i \in L\}$, and fulfills the requirements as defined in Game IV.

Solve DLP. Using the forking lemma for ring signature schemes [6], after replays $\mathscr{A}_2$ with the same random tape except the result returned by H_3 query of the forged message, $\mathscr{C}$ gets two valid ring signcryptions with probability $\frac{\varepsilon^2}{66C^n_{q_{H_3}}}$: $\{z, V, \bigcup_{i=1}^n\{A_i\}, \bigcup_{i=1}^n\{c_i\}\}$ and $\{z', V, \bigcup_{i=1}^n\{A_i\}, \bigcup_{i=1}^n\{c'_i\}\}$. It follows that $c_s \neq c'_s$, $c_i = c'_i$ for $i \neq s$. If ID^* is the actual signer, then $z = r_2 - c_s a \pmod q$ and $z' = r_2 - c'_s a \pmod q$, $\mathscr{C}$ solves DLP by computing: $a = (c'_s - c_s)^{-1}(z - z') \bmod q$.

Probability. Let $q_{H_i}(i = 1, 2, 3, 4)$, q_U, q_R, q_D and q_S be the number of $H_i(i = 1, 2, 3)$ queries, user public key queries, user public key replacement requests, partial private key queries and signcryption queries, respectively.

Without loss of generality, we may assume that $L_E \cap L_R = \emptyset$, and denote some events as follows: π_1: $\mathscr{C}$ does not fail during the queries; π_2: $ID^* \in L$; π_3: ID^* is the actual signer. It is easy to get following results:
$\Pr[\pi_1]=\frac{q_U-q_E}{q_U}$, $\Pr[\pi_2|\pi_1]=\frac{n}{q_U-q_E-q_R}$, $\Pr[\pi_3|\pi_1\wedge\pi_2]=\frac{1}{n}$.

$$\begin{aligned}\Pr[\mathscr{C}\ success] &= \Pr[\pi_1 \wedge \pi_2 \wedge \pi_3] \\ &= \Pr[\pi_1] \cdot \Pr[\pi_2|\pi_1] \cdot \Pr[\pi_3|\pi_1 \wedge \pi_2] \\ &= \frac{q_U - q_E}{q_U} \cdot \frac{n}{q_U - q_E - q_R} \cdot \frac{1}{n} \\ &\geq \frac{1}{q_U}\end{aligned}$$

Therefore, if $\mathscr{A}_2$ can succeed with the probability ε, then $\mathscr{C}$ can solve the DLP with probability $\frac{\varepsilon^2}{66C^n_{q_{H_3}}} \cdot \frac{1}{q_U}$. □

Theorem 5. *The scheme is anonymous.*

Proof. In the scheme, because A_i, c_i are randomly selected from G_1 and Z_q^* for $i \neq s$, respectively. h_i are hash functions values for $i \neq s$, and δ_1 is randomly selected from Z_q^*, so $A_s = \delta_1 E_s - \sum_{i=1, i\neq s}^n (A_i + h_i E_i)$ is distributed uniformly. Since δ_2 is chosen uniformly at random from Z_q^* and y is the output of the random oracle, then $c_s = y - \sum_{i=1, i\neq s}^n c_i \pmod q$ is distributed uniformly. By h_s is the output of the random oracle, then h_s is distributed uniformly. Further, z and V are also distributed uniformly over Z_q^* and G_1, respectively. □

In conclusion, no matter who is the actual signer, all the mentioned parameters are independent and uniformly distributed for any message m, receiver and the user ring L. Therefore, even an adversary with all the private keys corresponding to the set of identities L and unbounded computing resources has no advantage in identifying the actual signer over random guessing.

5 Efficiency and Comparison

By using a famous encryption library (MIRACL) on a mobile device (Samsung Galaxy S5 with a Quad-core 2.45G processor, 2G bytes memory and the Google Android 4.4.2 operating system), He *et al.* [12] obtained the running time for cryptographic operations. The running time are listed in Table 1.

For the CLRSC scheme based on bilinear pairing, we use the Tate bilinear pairing $G_1 \times G_1 \longrightarrow G_2$, where G_1 with prime order $\hat{q}$ is an additive group defined on a super sigular elliptic curve $E/E_p : y^2 = x^3 + x$ over the finite field $F_{\hat{p}}$, and $\hat{p}$ and $\hat{q}$ are 512 bits and 160 bits, respectively. To achieve the same level of security, for the CLRSC based on the non-singular elliptic curve cryptography, we use an additive group G_3 with the prime order $\hat{q}$, which is defined on a non-sigular elliptic curve over the finite field $F_{\hat{p}}$, where both $\hat{p}$ and $\hat{q}$ are 160 bits. We define some notations as follows:

- P: a pairing operation.
- M_{G_1}: a scalar multiplication operation in G_1.
- M_{G_3}: a scalar multiplication operation in G_3.
- E_{G_2}: a exponentiation operation in G_2.
- n: the number of members in the ring.

We use a simple method to evaluate the computation efficiency of different schemes. For example, the scheme [30] needs $3n+5$ pairing operations, $3n+2$ scalar multiplication operation in G_1. Therefore, the resulting operation time is $(3n+5) \times 32.713 + (3n+2) \times 13.405 = 190.375 + 138.354n$. We now let n = 10, and then the computation time is $190.375 + 138.354 \times 10 = 1573.915$.

According to the above ways, the detailed comparisom results of other schemes [22, 34] are shown in Table 2.

Table 1: Cryptographic operation time (in milliseconds)

P	M_{G_1}	M_{G_3}	E_{G_2}
32.713	13.405	3.335	2.249

6 Conclusion

In recent years, some good results have been achieved in speeding up the computation of pairing function. However, the pairing operation is still relatively expensive. So it is still quite significant to design CLRSC scheme with less pairing operations. In this paper, we constructe a new CLRSC scheme and prove the security against the Type I/II adversary in the random oracle model.

Our proposed scheme is proved to be indistinguishable against adaptive chosen ciphertext attacks, existentially unforgeable against adaptive chosen message attacks and anonymous. The proposed scheme based on certificateless cryptography, it avoids the storage problem of public

Table 2: Comparison of several CLRSC schemes

Scheme	Signcryption	Unsigncryption	Time(n = 10)
Qi [22]	$P + (2n+3)M_{G_1} + E_{G_2}$	$3P + (n+1)M_{G_1}$	588.871
Wang [30]	$(n+2)P + (2n+2)M_{G_1}$	$(2n+3)P + nM_{G_1}$	1573.915
Zhu [34]	$3nP + (n+4)M_{G_1} + nE_{G_2}$	$(2n+1)P + M_{G_1} + nE_{G_2}$	1914.418
Our scheme	$P + (n+3)M_{G_1} + (n+2)M_{G_3}$	$3P + nM_{G_1} + (n+2)M_{G_3}$	519.207

key certificate of public key infrastructure and the key escrow problem in identity based system. Our scheme only requires four pairing operations. Compared with other schemes [22,30,34], our CLRSC scheme is more efficient in computation. Because of the good nature of our scheme, it should be useful for practical application in the ring signcryption.

Acknowledgments

The authors are grateful to the anonymous referees for their helpful comments and suggestions. The research is supported by the National Natural Science Foundation of China under Grants 61562012, the Innovation Group Major Research Projects of Department of Education of Guizhou Province under Grant No. KY[2016]026.

References

[1] R. S. Abdeldaym, H. M. A. Elkader, R. Hussein, "Modified RSA algorithm using two public key and Chinese remainder theorem," *International Journal of Electronics and Information Engineering*, vol. 10, no. 1, pp. 51-64, 2019.

[2] S. S. Al-Riyami and K. G. Paterson, "Certificateless public key cryptography," in *9th International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT'03)*, LNCS 2894, pp. 452-473, 2003.

[3] C. C. Chang, M. S. Hwang, "Parallel computation of the generating keys for RSA cryptosystems", *Electronics Letters*, vo1. 32, no. 15, pp. 1365–1366, 1996.

[4] S. Y. Chen, P. Zeng, K. K. R. Choo and X. L. Dong, "Efficient ring signature and group signature schemes based on q-ary identification protocols," *The Computer Journal*, vol. 61, no. 4, pp. 545-560, 2018.

[5] L. Z. Deng, S. W. Li and Y. F. Yu, " Identity-based threshold ring signcryption from pairing," *International Journal of Electronic Security and Digital Forensics*, vol. 6, no. 2, pp. 333-342, 2014.

[6] L. Z. Deng, C. Liu and X. Wang, " An improved identity-based ring signcryption scheme," *Information Security Journal*, vol. 22, no. 1, pp. 46-54, 2013.

[7] L. Z. Deng, "Certificateless ring signature scheme based on RSA problem and DL problem," *RAIRO-Theoretical Informatics and Applications*, vol. 49, no. 4, pp. 307-318, 2015.

[8] M. Dissanayake, "A new modular multiplication method and its application in RSA cryptosystem," *International Journal of Electronics and Information Engineering*, vol. 10, no. 1, pp. 24-33, 2019.

[9] T. Feng, N. N. Liu, "A sensitive information protection scheme in named data networking using attribute-based ring signcryption," in *IEEE Second International Conference on Data Science in Cyberspace (DSC'17)*, pp. 187-194, 2017.

[10] C. Gritti, W. Susilo and T. Plantard, "Logarithmic size ring signatures without random oracles," *IET Information Security*, vol. 10, no. 1, pp. 1-7, 2016.

[11] Z. Guo, "Cryptanalysis of a certificateless conditional privacy-preserving authentication scheme for wireless body area networks," *International Journal of Electronics and Information Engineering*, vol. 11, no. 1, pp. 1-8, 2019.

[12] D. B. He, H. Wang, L. Wang, J. Shen and X. Yang, "Efficient certificateless anonymous multi-receiver encryption scheme for mobile devices," *Soft Computing*, vol. 21, no. 22, pp. 6801-6810, 2017.

[13] X. Y. Huang, W. Susilo, Y. Mu and F. T. Zhang, "Identity-based ring signcryption schemes: cryptographic primitives for preserving privacy and authenticity in the ubiquitous world," in *19th International Conference on Advanced Information Networking and Applications (AINA'05)*, IEEE, vol. 2 pp. 649-654, 2005.

[14] M. S. Hwang, J. W. Lo, S. C. Lin, "An efficient user identification scheme based on ID-based cryptosystem", *Computer Standards & Interfaces*, vol. 26, no. 6, pp. 565–569, 2004.

[15] A. A. Jothi and D. B. Srinivasan, "Security analysis in boby area networks using attribuute-based ring signcryption scheme," *Research Journal of Applied Sciences, Engineering and Technology*, vol. 13, no. 1, pp. 48-56, 2016.

[16] A. V. N. Krishna, A. H. Nareyana, K. M. Vani, "Window method based cubic spline curve public key cryptography," *International Journal of Electronics and Information Engineering*, vol. 4, no. 2, pp. 94-102,

2016.
[17] T. C. Lin, T. Y. Yeh, M. S. Hwang, "Cryptanalysis of an ID-based deniable threshold ring authentication", *International Journal of Network Security*, vol. 21, no. 2, pp. 298–302, 2019.
[18] L. H. Liu, Z. Z. Guo, Z. J. Cao and Z. Chen, "An improvement of one anonymous identity-based encryption scheme," *International Journal of Electronics and Information Engineering*, vol. 9, no. 1, pp. 11-21, 2018.
[19] L. H. Liu, Z. Z. Guo, Z. J. Cao and Z. Chen, "Anonymity and certificateless property could not be acquired concurrently," *International Journal of Electronics and Information Engineering*, vol. 7, no. 2, pp. 61-67, 2017.
[20] L. H. Liu, W. P. Kong, Z. J. Cao and J. B. Wang, "Analysis of one certificateless encryption for secure data sharing in public clouds," *International Journal of Electronics and Information Engineering*, vol. 6, no. 2, pp. 110-115, 2017.
[21] M. J. Qin, Y. L. Zhao and Z. J. Ma, "Pratical constant-size ring signature," *Journal of Computer Science and Technology*, vol. 33, no. 3, pp. 533-541, 2018.
[22] Z. H. Qi, G. Yang and X. Y. Ren, "Provably secure certificateless ring signcryption scheme," *China Communications*, vol. 8, no. 3, pp. 99-106, 2011.
[23] R. L. Rivest, A. Shamir and Y. Tauman, "How to leak a secret," in *International Conference on the Theory and Application of Cryptology and Information Security*, LNCS 2248, pp. 552-565, 2001.
[24] J. L. Salazar, J. L. Tornos and J. J. Piles, "Efficient ways of prime number generation for ring signatures," in *IET Information Security*, vol. 10, no. 1, pp. 33-36, 2016.
[25] S. Shan, "An efficient certificateless signcryption scheme without random oracles," *International Journal of Electronics and Information Engineering*, vol. 11, no. 1, pp. 9-15, 2019.
[26] S. S. D. Selvi, S.S. Vivek and C.P. Rangan, "On the security of identity based ring signcryption schemes," in *proceedings of ISC*, pp. 310-325, 2009.
[27] A. Shamir, "Identity-based cryptosystem and signature scheme," in *Advances in Cryptology (Crypto'84)*, LNCS 196, pp. 47-53, 1984.
[28] G. Sharma, S. Bala and A. K. Verma, "Pairing-free certificateless ring signcryption (PF-CLRSC) scheme for wireless sensor networks," *Wireless Personal Communications*, vol. 84, no. 2, pp. 1469-1485, 2015.
[29] H. Shen, J. Chen, D. He and J. Shen, "Insecurity of a pairing-free certificateless ring signcryption scheme," *Wireless Personal Communications*, vol. 96, no. 4, pp. 5635-5641, 2017.
[30] L. L. Wang, G. Y. Zhang and C. G. Ma, "A secure ring signcryption scheme for private and anonymous communication," in *IFIP International Conference on Network and Parallel Computing Workshops*, pp. 107-111, 2007.
[31] H. Xiong, J. Geng, Z. G. Qin and G. B. Zhu, "Cryptanalysis of attribute-based ring signcryption scheme," *International Journal of Network Security*, vol. 17, no. 2, pp. 224-228, 2015.
[32] C. X. Zhou, Z. M. Cui and G. Y. Gao, " Efficient identity-based generalized ring signcryption scheme," *Ksii Transactions on Internet and Information Systems*, vol. 10, no. 12, pp. 6116-6134, 2016.
[33] Z. C. Zhu, Y. Zhang and F. J. Wang, " An efficient and provable secure identity-based ring signcryption scheme," *Computer Standard and Interfaces*, vol. 31, no. 6, pp. 1092-1097, 2009.
[34] L. J. Zhu, F. T. Zhang and S. Q. Miao, " A provably secure parallel certificatelesss ring signcryption scheme," in *International Conference on Multimedia Information Networking and Security (MINES'10)*, pp. 423-427, 2010.

Biography

Hui Guo received her B.S. from Guizhou Normal University, Guiyang, PR China, in 2016; She is now a master student in the School of Mathematical Sciences, Guizhou Normal University, Guiyang, PR China. Her recent interest include cryptography and information safety.

Lunzhi Deng received his B.S. from Guizhou Normal University, Guiyang, PR China, in 2002; M.S. from Guizhou Normal University, Guiyang, PR China, in 2008; and Ph.D. from Xiamen, PR China, in 2012. He is now a professor in the School of Mathematical Sciences, Guizhou Normal University, Guiyang, PR China. His recent interest include algebra and information safety.

A Modified Advanced Encryption Standard for Data Security

Lin Teng, Hang Li, Shoulin Yin, and Yang Sun
(Corresponding author: Hang Li)

Software College, Shenyang Normal University
No. 253, HuangHe Bei Street, Huang Gu District, Shenyang 110034, China
(Email: lihangsoft@163.com)

(Received June 2, 2018; Revised and Accepted Nov. 10, 2018; First Online June 26, 2019)

Abstract

With the continuous development of society and economic progress, when a large amount of data enters the cloud computing system, people will pay more attention to data security. In order to make the stored data in the cloud more secure, according to the characteristics of cloud computing, we study the modified data encryption algorithm in cloud computing. First traditional advanced encryption standard (AES) is analyzed. Then a modified advanced encryption standard for data security in cloud computing is proposed by introducing random disturbance information to improve the data security. What's more, column mix operation and key choreography in AES are improved. Finally, experiments are conducted on Hadoop. Formal security analysis and performance comparisons indicate that the proposed solutions simultaneously ensure attribute privacy and improve decryption efficiency for outsourced data storage in mobile cloud computing.

Keywords: AES; Key Choreography; Random Disturbance Information

1 Introduction

Hamming Codes are one of the EDC codes which are used to protect the registers and memories from soft errors. As technology scales, radiation particles can create more soft error likely to affect the more than one bit binary number. Big-data storage poses significant challenges to anonymization of sensitive information against data sniffing. Not only will the encryption bandwidth be limited by the I/O traffic, the transfer of data between the processor and the memory will also expose the input-output mapping of intermediate computations on I/O channels that are susceptible to semi-invasive and non-invasive attacks.

Cloud computing [6,9,12,17] is an emerging computing model applied to the Internet. It provides basic resource facilities, application systems, and software platforms as services to users.Cloud computing is also a virtualization-based architecture that virtualizes resources and builds large-scale resource pools and manages services externally.

With the development of cloud computing, amounts of user data and large-scale data are put into cloud computing systems. Because of the distributed and virtualized nature of cloud computing, users cannot intuitively determine the storage location and division of data, *etc.*, so the security of data becomes very important. In the cloud computing, data security is generally ensured through data encryption and identity management [3, 10]. At present, the common encryption algorithms are classified into symmetric encryption algorithm and public key encryption algorithm. Among them, DES algorithm and AES algorithm are two widely used algorithms in symmetric encryption algorithms [1, 4, 15, 19].

Therefore, many researchers proposed many new schemes to solve the above issue. Deng [2] proposed an effective PKC-based certificateless group authenticated key agreement protocol, the certificateless mechanism of the protocol simplified the complex certificate management problem and key escrow problem in ID-based protocols. The security of the scheme was proved and its computational cost was discussed. The result showed that the new protocol was secure and effective. Shan [13] proposed an improved protocol to append a signature in the second round to eliminate weakness of certificateless group key agreement protocol. The signature was related to the group identity, the broadcast messages in the first round and the computed message in the second round, to ensure the protocol freshness and the entity authenticity. The message in the second round guarantees that the adversary could not attack the protocol by corrupting neighboring entities. Zhang [20] studied authenticated AGKA in certificateless and identity-based public key cryptosystems. They formalized the security model of certificateless authenticated asymmetric group key agreement and realized a one-round certificateless authenticated asymmetric group key agreement protocol to resist active attacks in the real world. They also investigated the re-

lation between certificateless authenticated AGKA and identity-based authenticated AGKA. So a concrete conversion from certificateless authenticated AGKA was proposed to session key escrow-free identity-based authenticated AGKA. Yin [18] introduced the concept of distributed Searchable asymmetric encryption, which was useful for security and could enable search operations on encrypted data.

This paper proposes a modified data encryption algorithm in cloud computing. First traditional advanced encryption standard is analyzed. Then a modified advanced encryption standard for data security in cloud computing is proposed by introducing random disturbance information to improve the data security. What's more, column mix operation and key choreography in AES are improved. In terms of security, the protocol can prove safety in the random prediction model; For performance, the new protocol requires only one round to complete authentication and key negotiation.

And for computation, compared with state-of-the-art schemes, the calculation of new protocols is also significantly reduced. The rest of the paper is organized as follows. Section 2 introduces the Hadoop Framework in this paper. Traditional encryption is explained in Section 3. Section 4 outlines the proposed scheme to analyze detailed processes. Experiments and performance analysis are given in Section 5. Finally, Section 6 concludes this paper.

2 Hadoop Framework

Hadoop is a distributed computing framework developed by the Apache storage and calculations for massive amounts of data. The core design of Hadoop framework is distributed file system (HDFS) and parallel computing framework (MapReduce). HFDS is responsible for the distribution and storage of data, and MapReduce is responsible for the calculation of data [11].

2.1 HDFS System

The essence of HDFS [16] is a distributed file system, which can divide a large data into small data sets and back them up, distributed and stored on different nodes in the cloud environment. However, for a single user, HDFS is like a traditional hierarchical file system. When used, HDFS can operate on big data just like a single file.

The HDFS framework is built on a set of specific nodes, which includes a unique NameNode to provide metadata services, guide computing nodes and data nodes to handle assigned tasks. Multiple DataNode is mainly for HDFS to provide storage blocks, and to perform read and write operations for distributed files. The data redundancy in the Hadoop platform is three, and each piece of data is stored in three DataNodes.

In the cloud computing environment, HDFS ensures the reliable storage of massive data through the following measures. DataNode sends a "heartbeat" message to NameNode regularly and sends the data block list information to determine whether the node is normal to provide a secure mode, only read views in this mode, it does not allow for additional or deletions and modification operations, record detailed log files, and test the integrity of the data taken by the user.

2.2 MapReduce Framework

MapReduce is a software framework that processes large data sets in parallel [14]. The root of MapReduce is the *map* and *reduce* functions in functional programming, corresponding to the mapping and specification in the calculation process. The Map process accepts a set of data and converts it into a key/value pair list, then transmits and reorder it. The *Reduce* process takes a list generated by the Map and then shrinks the list of key/value pairs based on their keys (generating a key/value pair for each key). That is, the Reduce process processes the integration and sorting of the intermediate results generated by the Map process, and then forms the final result.

3 Traditional Encryption Algorithm

The amount of data in cloud computing is very large, and often scattered on different computing nodes. The security protection is very important. Encrypting and decrypting data through encryption algorithms is one of the most effective methods to ensure data security. This article mainly discusses and improves the symmetric encryption algorithm in traditional encryption algorithms. Symmetric encryption algorithms use the same key for encryption and decryption, such as DES and AES algorithms. A symmetric encryption system can be represented as $CS = M, C, K, e, d$, where: $m \in M$ represents a plaintext message set; $C = c$ represents a ciphertext message set; $K = k$ represents the key set; E represents the encryption mapping process, *i.e.* $E : K*M = C$; D represents the decryption mapping process, *i.e.* $D : K*C = M$.

During the execution of the DES algorithm, the plaintext is grouped in 64 bits, and the last group with less than 64 bits is patched according to a specific method. The key length is 8 bytes, but 8 bits are the check bits. In the encryption phase, the plaintext is first divided into 32 parts by initial replacement, represented by the left half and the right half. Then perform 16 rounds of operations to combine the data and the key. The key is shifted in each round of operations, 48 bits out of the 56 bits of the key are selected, the original 32 bits of the right half are replaced by 48 bits through expansion, and then the XOR operation is combined with the 48-bit key Then, the 48 bits are converted to 32 bits by the S box, and then XOR with the original 32 bits of the left half.Finally, the final ciphertext is obtained by inverse initial permutation. The algorithm flow chart is shown in Figure 1.

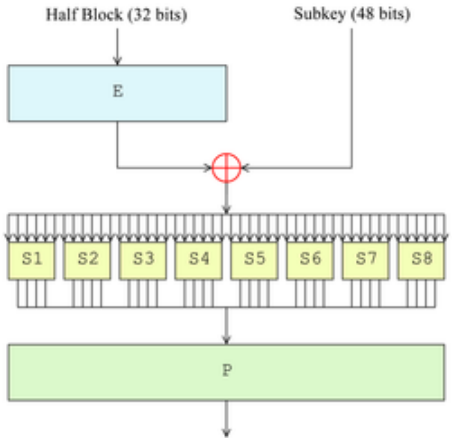


Figure 1: DES process

The AES algorithm is also a classical algorithm for symmetric key encryption. The length of the block in AES is 128 bits, and the key length can be 128 bits, 192 bits, or 256 bits.The AES encryption process operates on a 4×4 byte matrix.The AES encryption process mainly performs four types of operations:

1) SubBytes means that each byte is replaced by a lookup table through the S box.

2) ShiftRows performs a cyclic shift operation on each row in the 4x4 matrix.

3) MixColumns, uses linear conversion to mix 4 bytes per column.

4) AddRoundKey refers to the XOR operation between each round of input and round keys in the encryption process, and the XOR key is used in the decryption process.

With the continuous improvement of cloud computing capabilities and the rapid development of computer hardware, the shortest key for the DES encryption algorithm is too short. The key length is 64 bits and 8 check bits are removed. The actual effective number of bits is only 56 bits. If the brute-force method is used to crack, only 256 possibilities need to be calculated. Particularly, the computing power of the cloud platform is used to complete the cracking in a short time. Keys have become possible. For the AES algorithm, the key length is up to 256 bits, and the using of the brute force method is less likely to forcibly crack, but this algorithm is not absolutely secure, if an attacker designs different keys to measure the exact time required for the encryption process. Once the encryption routine is carelessly encoded, the execution time depends on the key value, and it is possible to derive information on the key.

4 Modified AES

AES adopts group iteration, patch size is 4×4 matrix. Each element is 8 bits. In order to make the algorithm applicable encryption, achieve better security and improve the encryption efficiency, we make the following improvements based on the AES algorithm framework.

4.1 Improved Key Sequence Generation Method

Chaotic dynamic system has pseudo randomness and is extreme sensitivity to initial conditions and system parameters. Therefore, it provides a good way for image information encryption. The improved algorithm adopts the following skew tent map to generate the key sequence.

$$F_a(x) = \begin{cases} x/a, x \in (0,a). \\ (1-x)/(1-a), x \in (a,1). \end{cases} \quad (1)$$

When $a \in [0,1]$, the system is in a chaotic state. The correlation of this mapping iterative trajectory sequence decreases exponentially, and the distribution of chaotic variables is uniform with good pseudo-random characteristics.

The method of generating pseudo-random sequences based on oblique tent mapping is as follows: one $M \times N$ image needs to encrypt R rounds. First, it iterates the oblique tent mapping and gets R sequence $X_r x_{r,0}, x_{r,1}, \cdots, x_{r,MN-1}, 1 \leq r \leq R$. X will be expanded to 0-255 integer sequence $K_r k_{r,0}, k_{r,1}, \cdots, k_{r,MN-1}$ according to following equation.

$$k_{r,i} = \llcorner x_{r,i} \times 255 \lrcorner.$$

where $\llcorner \lrcorner$ denotes round to-infinite.

4.2 Improved Encryption and Decryption

The encryption way of AES is that matrix D and key sequence K execute XOR operation. In order to increase the sensitivity to plaintext, the algorithm is improved. The encryption process is:

$$C[i][j] = \begin{cases} D[i][j] \bigoplus k_{r,i \times N+j}, i = M-1, j = N-1. \\ D[i][j] \bigoplus (k_{r,i \times N+j} \bigoplus D[i+1][0]), \\ \qquad i \neq M-1, j = N-1. \\ D[i][j] \bigoplus (k_{r,i \times N+j} \bigoplus D[i][j+1]), others. \end{cases} \quad (2)$$

Where $i \in [0, M-1]$, $j \in [0, N-1]$, $D[i][j]$ are plaintext pixels. $C[i][j]$ is obtained cipher pixel.

4.3 Key Tree

The plaintext matrix is encrypted from left to right. Ciphertext matrix is decrypted from top to bottom and from right to left. After XOR operation, key and plaintext are combined. Two different images even if use the same initial conditions, but the generated key sequences are different.

25	13	242	79
66	24	35	176
242	13	71	233
195	140	125	72
78	238	97	248

(a) 5×4 matrix

25	38	255	65
66	91	58	212
242	255	84	49
195	79	8	196
78	60	79	89

(b) Row diffusion result

25	13	242	79
41	10	49	97
201	3	22	136
250	137	102	191
83	102	252	58

(c) Column diffusion result

Figure 2: Diffusion enhancement result

4.4 Improved Column Mixing

In AES, Column mixing operation $MixColumns$ adopts the matrix operation, each pixel takes shift and XOR operation. In order to reduce the computational complexity and achieve good mixing effect, in the improved algorithm, we changed the $MixColumns$ matrix operations, the simple addition and subtraction are adopted to strengthen the relationship between pixels. The concrete measures are as follows: for each row, the first pixel remains the same, and the current pixel is updated with adjacent pixels from the second pixel (as shown in Equation (3)); For each column, the first pixel remains the same, and the current pixel is updated with the value of adjacent pixels starting from the second pixel (as shown in Equation (4)).

$$MixColumns = \begin{cases} D[i][j] = D[i][j], j = 0. \\ D[i][j] = (D[i][j] - D[i][j-1] \\ \qquad \bmod 256, others. \end{cases} \quad (3)$$

$$MixColumns = \begin{cases} D[i][j] = D[i][j], i = 0. \\ D[i][j] = (D[i][j] - D[i-1][j] \\ \qquad \bmod 256, others. \end{cases} \quad (4)$$

With an example of 5×4 matrix, the operation result is as shown in Figure 2. From this figure, we can know that when $D[0][0]$ is changed, it will affect all pixels. When $D[M-1][N-1]$ is changed, it does not affect other pixels in the same round. So in the row and column transformation operations, each line should be moved to the left, each column moves up. After several encryption round, it has obvious diffusion effect. Improved row and column mixed operations, it uses simple addition and subtraction, each pixel needs only two additive operations, the operation is not only reduces the computational complexity, but strengthens the connection between the pixels. After several rounds of encryption, it can achieve better mixing effect.

5 Experiment and Analysis

This experiments are conducted in MATLAB platform with SSH framework and clusters simulation cloud computing environment. Clusters are composed of six computers with one computer CPU I7, memory 8GHz, frequency 3.2GHz, this computer is as $NameNode$. To better simulate the cloud environment, the other five computers are selected five different machines as $Slave$ and $DataNode$.

Because the biggest advantage of cloud computing is that it can process large data's storage and calculation, this experiment chooses the size of 1.5GB text file as the experimental data. The data calculated by MapReduce in Hadoop platform, we test the performance of proposed algorithm and make comparison with other encryption methods including RSAE [5], CTME [8] and SUE [7]. Because the performance of AES is poorer than RSAE, we did not compare with AES.

5.1 Performance Analysis

We do the following performance analysis.

1) Plaintext sensitivity analysis. If no interference information is added, the plaintext sensitivity of the transformation is the same as AES algorithm if it falls into the M_D segment. If it falls into the M_A segment, the sensitivity is the same as AES algorithm too. However, new algorithm is more sensitive to plaintext than AES because random variable interference information is added to both sections of plaintext.

2) Key sensitivity analysis. The key sensitivity of new algorithm is determined by AES algorithm. If the key K_1 is changed, the middle plaintext M_{D1} is changed. If the key K_2 is changed, the middle plaintext M_{A1} will be changed too. When the key changes slightly, the final ciphertext will be greatly changed. This algorithm has better key sensitivity.

3) Against attack analysis. Attackers attack new algorithm which means that it needs to defeat AES algorithm. For the ciphertext attack, the obtained ciphertext just locates in the segment point, the probability is very small. Even getting the key to decrypt the ciphertext C_D and C_A, the plaintext M_A and M_D are more difficult to obtain, so the original plaintext M is difficult to acquire too. And that plaintext attack can also be difficult to speculate the original plaintext message.

5.2 Time Analysis

MapReduce calculates data by default in 64MB block. To intuitive display performance of the new algorithm, we set file block size with 64MB, 32MB, 16MB, 8MB, 4MB and 2MB. We conduct 8 encryption experiments and 8 decryption experiments. The average execution time of the algorithm was taken as shown in Figures 3 and 4. AE: average time.

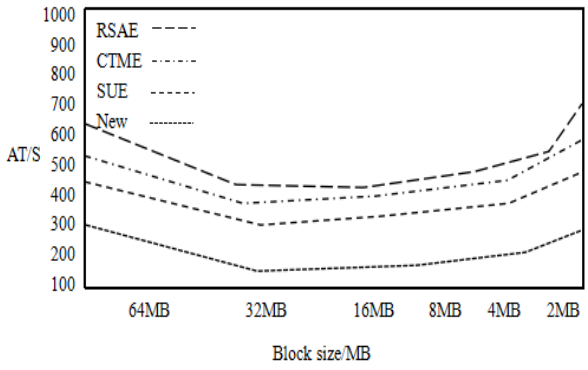


Figure 3: Average encryption time

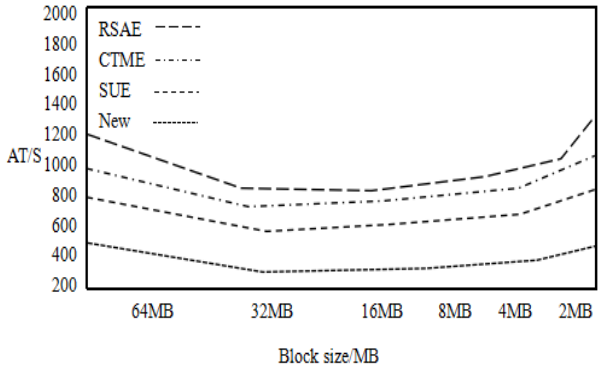


Figure 4: Average decryption time

In Hadoop platform, block size of the original data has the similar effect for the four algorithms. The influence of partitioned values changed from 64MB to 32MB or 16MB, the algorithms have the highest execution efficiency, if it continues to reduce the block value, especially when the block value is 2MB, four algorithms' execution time are rising sharply, this is because when the data block unit is too small, the block number will surge. In MapReduce calculation, Reduce process can consume time for the integrate ordering of Map. So in the cloud computing, the division of big data should take appropriate units, otherwise, it would affect the computation time. We also can get that new method has better performance in encryption and decryption process than other three methods. In summary, the proposed method has high security in cloud computing.

6 Conclusion

Cloud computing is a widely promising commercial calculation model based on virtualization of resources. Large huge amounts of data can be calculated and managed. It provides service according to the customer's demand. This paper analyzes the Hadoop technology and constructs experimental platform. Firstly, traditional data encryption algorithm is introduced. Aiming at the shortcomings of the raw algorithms in cloud computing environment, this paper puts forward a modified encryption algorithm by introducing random disturbance information to improve the data security. Finally, the experiments results prove that proposed algorithm is suitable for encryption in cloud computing environment.

Acknowledgments

This study was supported by the Natural Science Fund Project Guidance Plan in Liaoning Province of China (No. 20180520024). The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

References

[1] D. S. AbdElminaam, "Improving the security of cloud computing by building new hybrid cryptography algorithms," *International Journal of Electronics and Information Engineering*, vol. 8, no. 1, pp. 40–48, 2018.

[2] F. Deng, Y. Zhu, "Novel one-round certificateless group authenticated key agreement protocol," *Computer Engineering & Applications*, vol. 53, no. 5, pp. 111-115, 2017.

[3] S. T. Hsu, C. C. Yang, and M. S. Hwang, "A study of public key encryption with keyword search", *International Journal of Network Security*, vol. 15, no. 2, pp. 71–79, Mar. 2013.

[4] S. H. Islam, A. Singh, "Provably secure one-round certificateless authenticated group key agreement protocol for secure communications," *Wireless Personal Communications*, vol. 85, no. 3, pp. 879-898, 2015.

[5] S. Kaufmann, "RSA public-key encryption," *Journal of Biological Chemistry*, vol. 280, no. 5, pp .3636-44, 2018.

[6] S. Khatoon, T. Thakur, B. Singh, "A provable secure and escrow-able authenticated group key agreement protocol without NAXOS trick," *International Journal of Computer Applications*, vol. 171, no. 3, pp. 1-8, 2017.

[7] K. S. Lee, S. G. Choi, D. H. Lee, "Self-updatable encryption: Time constrained access control with hidden attributes and better efficiency," *Theoretical Computer Science*, vol. 667, pp .51-92, 2017.

[8] C. Li, G. Luo, K. Qin, et al, "An image encryption scheme based on chaotic tent map," *Nonlinear Dynamics*, vol. 87, no. 1, pp .127-133, 2017.

[9] J. Liu, S. L. Yin, H. Li and L. Teng, "A density-based clustering method for k-anonymity privacy protection," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 8, no. 1, pp. 12-18, Jan. 2017.

[10] L. Liu, Z. Cao, C. Mao, "A note on one outsourcing scheme for big data access control in cloud," *International Journal of Electronics and Information Engineering*, vol. 9, no. 1, pp. 29–35, 2018.

[11] R. R. Parmar, S. Roy, D. Bhattacharyya, "Large-scale encryption in the hadoop environment: Challenges and solutions," *IEEE Access*, vol. 5, pp. 7156-7163, 2017.

[12] S. Rezaei, M. Ali Doostari, and M. Bayat, "A lightweight and efficient data sharing scheme for cloud computing," *International Journal of Electronics and Information Engineering*, vol. 9, no. 2, pp. 115–131, 2018.

[13] C. Shan, H. U. Kangwen, J. Xue, "Improved pairing-free constant round certificateless authenticated group key agreement protocol," *Journal of Tsinghua University*, vol. 57, no. 6, pp. 580-585, 2017.

[14] B. Sheintz, A. Chandra, R. K. Sitaraman, "End-to-end optimization for geo-distributed mapReduce," *IEEE Transactions on Cloud Computing*, vol. 4, no. 3, pp. 293-306, 2017.

[15] L. Teng, H. Li, S. L. Yin, "A multi-keyword search algorithm based on polynomial function and safety inner-product method in secure cloud environment," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 8. no. 2, pp, 413-425, 2017.

[16] S. Wu, W. Zhu, B. Mao, "PP: Popularity-based proactive data recovery for HDFS RAID systems," *Future Generation Computer Systems*, vol. 86, pp. 1146-1153, 2018.

[17] S. L. Yin and J. Liu, "A k-means approach for map-reduce model and social network privacy protection," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 7, no. 6, pp. 1215-1221, Nov. 2016.

[18] S. L. Yin, L. Teng, J. Liu, "Distributed searchable asymmetric encryption," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 4, no. 3, pp. 684-694, 2016.

[19] Q. C. Zhang, T. L. Yang, X. G. Liu, Z. K. Chen, and P. Li, "A tucker deep computation model for mobile multimedia feature learning," *ACM Transactions on Multimedia Computing, Communications and Applications*, vol. 13, no. 3, pp. 1-39:18, 2017.

[20] L. Zhang, Q. Wu, B. Qin, "Certificateless and identity-based authenticated asymmetric group key agreement," *International Journal of Information Security*, vol. 16, no. 5, pp. 559-576, 2017.

Biography

Lin Teng received the B.Eng. degree from Shenyang Normal University, Shenyang , Liaoning province, China in 2016. Now, she is studying for Master degree in Software College, Shenyang Normal University. Her research interests include Multimedia Security, Network Security, Filter Algorithm and Data Mining.

Email:910675024@qq.com.

Hang Li is a full professor in Software College, Shenyang Normal University. He received his B.S. and M.S. degrees from Northeastern University. He is also a master's supervisor. He has research interests in wireless networks, mobile computing, cloud computing, social networks, network security and quantum cryptography. Professor Li had published more than 30 international journal papers (SCI or EI journals) on the above research fields. Email: lihangsoft@163.com.

Shoulin Yin received the B.Eng. and M.Eng. degree from Shenyang Normal University, Shenyang , Liaoning province, China in 2016 and 2013 respectively. Now, he is a doctor in Harbin Institute of Technology. His research interests include Multimedia Security, Network Security, Filter Algorithm, image processing and Data Mining. Email:yslinhit@163.com.

Yang Sun obtained his master degree in Information Science and Engineering from Northeastern University. Yang Sun is a full professor of the Kexin software college at Shenyang Normal University. He is also a department head of network engineering. He has research interests in wireless networks, mobile computing, cloud computing, social networks and network security. Yang Sun had published more than 20 international journal and conference papers on the above research fields. Email:17247613@qq.com.

Ensuring Users Privacy and Mutual Authentication in Opportunistic Networks: A Survey

Cossi Blaise Avoussoukpo, Chunxiang Xu, and Marius Tchenagnon

(Corresponding author: Cossi Blaise Avoussoukpo)

School of Computer Science and Engineering, University of Electronic Science and Technology of China
No.2006, Xiyuan Ave, West High-Tech Zone, Chengdu 611731, Sichuan, China
(Email: omramson@yahoo.fr)

(Received June 8, 2018; Revised and Accepted Oct. 20, 2018; First Online Feb. 16, 2019)

Abstract

The Opportunistic Communication main goal is to use short-term, simple, easy, convenient, and quick actions to communicate when limited or no traditional Communications infrastructure is available. For users' altruism represents the heart of any OppNets, using Communications Technologies such as Bluetooth, Wi-Max, or Wi-Fi to communicate poses not only routing challenges but also users privacy challenges. However, most researches on OppNets domain, focus more on routing security than users mutual authentication and privacy. This work provides a review of the state of the art proposals on users privacy and mutual authentication techniques with three main contributions. First, it clarifies the concept of OppNets. Second, it Points out the differences between OppNets and some communications models that emerged from Mobile Ad hoc Networks research. Third, it succinctly reviews the main existing techniques proposed for users mutual authentication and privacy protection in OppNets, organising them in a taxonomy. Finally, it discusses, the limits of the techniques studied.

Keywords: Mutual Authentication; OppNets; Opportunistic Communication; Privacy Protection

1 Introduction

Hitherto recently, mobile Ad Hoc network has captured the attention of researchers for years. It follows, the advent of Delay Tolerant Networks, Unstructured Networks, and Peer to Peer Communications. However, with the pervasiveness of mobile and fixed smart devices or systems equipped with various kinds of communication media such as Bluetooth, Wired Internet, Wi-Fi, Ham Radio, Satellite, a new type of network based on devices discovery called Opportunistic Networks surfaced. Moreover, the increasing number of mobile smart devices in use, together with the gregarious nature of human mobility, gives not only the idea of the creation of smart city [10, 20, 21] but also opens up the idea to use the mobility of devices for opportunistic communications when mobile devices come into contact. Here, mobility is an opportunity, not a challenge.

Opportunistic Networks [1] as a natural evolution of mobile Ad-Hoc network, are self–configured and made up of diverse systems, not formerly employed as components, which join dynamically to exploit the resources of separate networks according to the needs of a specific application task. Opportunistic Networks do not have an end-to-end path and rely solely on a Seed node (supernode, source node or root note) that invites other nodes called Helpers to form together, the opportunistic networks, whenever needs are. Here, both Seed node and Helpers that form the Opportunistic Network are not predefined, in other words, there are no fixed architectures like other networks to manage the Opportunistic Networks. For users represent the heart of Opportunistic Networks, OppNets can be useful across many domains such as crises management, info-mobility services and intelligent transportations, and pervasive healthcare. However, the wireless communication is not a safe environment [33]. Therefore, operating in OppNets poses not only routing challenges but also users privacy challenges. However, Opportunistic Networks related research tends to focus on routing.

There are various type of surveys on OppNets routing, among others the most useful and recent Nessrine Chacchouk's work [9]. There is less work dedicated to mutual authentication and users privacy for Opportunistic Networks. Meanwhile, users might be reluctant to join an Opportunistic Network if their identity, social links, or location can be compromised when operating in an Opportunistic Network environment. Moreover, due to the user-centric nature of such networks, users mutual authentication, when addresses rigorously can allow more users to join an Opportunistic network with confidence. That justifies the importance of this survey that reviews

the state of the art techniques used for users privacy (location, social links, identity) and mutual authentication schemes providing three main contributions. First, it clarifies the concept of OppNets. Second, it Points out the differences between OppNets and some communications models that emerged from Mobile Ad hoc Networks research. Third, it succinctly reviews the main existing techniques proposed for users mutual authentication and privacy protection in OppNets, organising them in a taxonomy. Finally, it discusses, the limits of the techniques studied.

The remainder of this paper goes as follows. Section 2 provides useful definitions. Section 3 clarifies the concept of Opportunistic Networks, characterises OppNets, and Points out the differences between OppNets and some communications models that emerged from Mobile Ad hoc Networks research. Section 4 provides a taxonomy of the most significant existing proposals in the domain of Opportunistic Networks on Users Privacy and Mutual Authentication respectively; describing the most relevant approaches, and discussing their pros and cons. Section 5 provides an insightful summary of the works presented within each class. Finally, Section 6 concludes the paper and gives future research directions.

2 Background

2.1 Multidimensional Scaling

Multidimensional scaling (MDS) is one of several multivariate schemes that study the similarity or distance between two objects (data) which are presented in a low dimensional space.MDS visualises the results to reveal the hidden structure in the data [11]. MDS uses the distance between each pair of the objects as input and generate (2D or 3D)-points as output.

2.2 Bloom Filter

Burton H. Bloom conceived Bloom filter [3] in 1970. Bloom filters are kinds of hash tables, probabilistic space-efficient data structures that verify whether an element is a member of a set [14]. The raison d'être of Bloom filters is that; they are more space-efficient than hash tables, super fast insert and super fast lookups. Bloom filters concede false positive but no false negative. For Broder and Mitzenmacher [6], on any occasion, a list or set is used, and space is case-sensitive, one can resort to Bloom filter if the false positive can be solved.

2.3 Dynamic Clustering

A cluster is a subset of data with common characteristics. Clustering, also called unsupervised learning is the process of making the difference between similar and dissimilar dataset dividing the dataset into groups. As opposed to static clustering, in dynamic clustering, the clusters are formed, and cluster heads are selected [5].

2.4 Opportunistic Network Contact Graph

A contact graph reveals keen pieces of information about social links. Two elements characterise the contact graph G; G={V,E}. V is a set of nodes and E a set of edges [16]. Figure 1 gives an idea of how an opportunistic contact graph can look like.

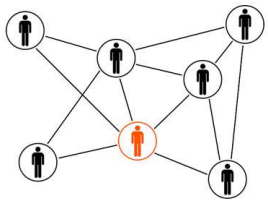


Figure 1: Network contact graph

2.5 K-Anonymity

K-Anonymity is a model for protecting data privacy. It relies on the principle that if at least K people share the same quasi-identifiers in the same table, then no individual can be individually tracked [25].

2.6 Markov Models

Markov models depend on Markov processes that are memoryless chains of events for which the next event depends on the current event but not the past event [17]. Markov models are composed of a set of states, state transition probability, and an initial state distribution. Figure 2 is an example of a Markov model with the states A, B, and C .

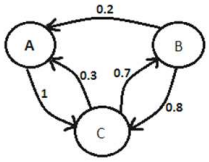


Figure 2: A markov model

2.7 Decisional Bilinear Diffie-Hellman Problem

Let $\mathbb{G}$, $\mathbb{G}_T$ be two multiplicative cyclic groups of prime order p, g a generator of $\mathbb{G}$ and e, a bilinear map; $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$. Let $x,y,z,c \in \mathbb{Z}_p$ be randomly chosen. The Decisional Bilinear Diffie-Hellman (DBDH) assumption [26] holds in $\mathbb{G}$ if no probabilistic polynomial-time algorithm can distinguish the tuples (g , g^x,g^y,g^z , $e(g,g)^{(xyz)}$) from the tuple (g , g^x,g^y,g^z , g^c) with non-negligible advantage [30].

2.8 Identity-Based Cryptography

Shamir [27] was the first person to propose the idea of Identity-Based Cryptography (IBC) in 1984. IBC main goal was to simplify the certification management of conventional PKI supported security schemes. However, it was until 2001 that Boneh and Franklin introduced the first practical solution of IBC based on the Diffie-Hellman Problem from Weil pairing. An IBC scheme is made up of four randomised algorithms [4].

Setup : Generate the master secret key S and the system parameters.

Extract : Given a user's identity, generate the corresponding private key by using the master secret key.

Encrypt : To encrypt a message m for a user, take the user's identity and m as input, and generate the corresponding ciphertext.

Decrypt : To decrypt a ciphertext c, take the user's private key and c as input, and recover the corresponding message.

2.9 Threshold Secret Sharing

Threshold secret sharing allows a secret to be shared among multiple parties or users in such a way that only a sufficient number of users together can reconstruct the secret.

2.10 Mutual Authentication

Generally, authentication is the process of establishing an identity; the process of proving that a user or a process is, who or what it claims to be. Mutual authentication, also called two-way authentication refers to two parties authenticating each other at the same time [23, 32]. In a Network, TLS and mTLS are examples of mutual authentication protocols.

3 Opportunistic Networks

3.1 What are Opportunistic Networks (OppNets)?

Leszek Lilien *et al.* were the first to clearly and formally define the concept of Opportunistic Networks (OppNets) [1]. Opportunistic Networks, characterised as the most challenging evolution of mobile Ad hoc Networks rely on limited or no infrastructure. Opportunistic Networks are self–configured and made up of diverse systems, not employed initially as components, which join dynamically to exploit the resources of separate networks according to the needs of a specific application task. Opportunistic Networks do not have an end-to-end path and rely solely on a Seed node(s) (Supernode(s) or Source node(s)). The Seed node(s) or Seed OppNet is an essential part of OppNets for everything starts with the Seed OppNet that expands by inviting other nodes called Helpers [31].

3.2 Significant Differences between OppNets and other Networks

The first and most important fact to understand is that most people mistake Opportunistic Communications For OppNets. Although nodes within an OppNets also communicate opportunistically, the "Opportunistic" referred to by other Networks is limited because for opportunistic communication to happen, devices wait till they are in each other range. In contrast, OppNets should realise opportunistic growth and opportunistic use of resources acquired by this opportunistic growth. Second, Delay Tolerant Networks routing algorithms, always look for an existing end to end route first. If there is no end to end route, Delay Tolerant Networks routing algorithms resort to opportunistic communications. On the other hand, for OppNets, messages are always sent opportunistically, and an existing end to end path is never considered.

3.3 Important Applications for OppNets

OppNets can be useful in all emergency situations, healthcare, and military. For example, OppNets can effectively and efficiently; help inform people before a disaster, organise rescue operation during and after a disaster [24,29]. Also, With an ageing society, and people living in remote areas with no access to proper medical facilities, people with chronic medical conditions [2] can enjoy remote healthcare assistance. Moreover, OppNets can be useful in the military for security operations.

3.4 Users Privacy Challenges in OppNets

Users are at the heart of OppNets for users are the ones who carry devices. So, users privacy and devices privacy are related. The most critical users privacy challenges for OppNets are Helpers privacy and OppNet privacy on the one hand, and authentication and mutual authentication of nodes within an OppNet on the other hand.

4 Taxonomy of Users Privacy and Mutual Authentication in OppNets

Since the advent of OppNets that is a natural evolution of Mobile Ad Hoc Networks, Researchers have achieved great things to advance the new domain of research, OppNets. However, research works tend to focus more on routing; there are even many surveys on routing in OppNets. Moreover, mutual authentication within an OppNet on the one hand, users privacy, on the other hand, get less attention. Meanwhile, within such a hostile environment like OppNets, these questions are worth to consider. The

following sections provided a succinct review of the existing literature based on the proposed taxonomy which is schematically illustrated in Figure 3 and sorted in Table 1.

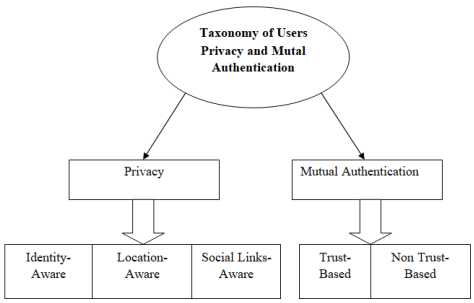


Figure 3: Taxonomy of users privacy and mutual authentication proposals for OppNets

Table 1: Users privacy and mutual authentication schemes taxonomy

schemes	schemes
Users Privacy	Mutual Authentication
Social Links Aware	Trust-Based
Distl [13]	Cao [7]
Distl [12]	Singh [28]
Location Aware	Non-Trust-Based
Zakhary [34]	Ma [22]
Zakhary [35]	Carver [8]]
Identity Aware	Guo [15]
Kaur [18]	Kumar [19]

4.1 Mutual Authentication in OppNets

Before two nodes enter an OppNet environment, they register at the Seed OppNet. They resort to mutual authentication before engaging in communication. Two classes of mutual authentication techniques are considered in this study: Non Trust-Based and Trust-Based. The literature on this topic is limited for it is a relatively new domain of research.

4.1.1 Trust-Based Mutual Authentication

Xiamei and Ying [7] proposed an authentication scheme based on trust and rooted on Multidimensional scaling. Their scheme depends on a trust model called M-Trust that relies on an integrate trust value $\mathbb{Q}_{\alpha}^{\beta}$ obtained by combining direct and indirect trust values. First of all, each node generates its private key; a root node participates slightly in that process. Also, each node sets the relationship intensity threshold δ. When two nodes α and β come into contact, α queries the local repository, calculate the integrate trust value. Then, after considering the value-at-risk, the node β can get a certificate from α if $\mathbb{Q}_{\alpha}^{\beta} \leq \delta$. The proposed scheme is an excellent job for it meets most of the requirements of the opportunistic networks. However, the root node(seed OppNet) does play a third party role which is not desirable for OppNets. Umesh Pal Singh and Naveem Chauhan [28] proposed an authentication scheme for opportunistic communication within a trust framework. The proposed scheme is a variant of Ming Huang Guo's work. Here, the authors added to Guo's work, the notion of dynamic registration where ordinary authenticated nodes become semi-super nodes. Seed node or static nodes appoint authenticated nodes as demi-super nodes by their trust and threshold values. The trust value depends on two parameters; encounter value and number of messages. Nevertheless, the trust value does not serve much in the process of mutual authentication.

4.1.2 Non-Trust-Based Mutual Authentication

Ma and Jamalipour [22] combined both (t, n) Threshold Secret Sharing and Identity-Based Cryptography and proposed a scheme that aims to mitigate malicious attacks through opportunistic nodes authentication. The proposed scheme considered the use of (t, n) secret sharing to solve not only the key escrow problem of Identity-Based Cryptography but also the single point of failure of PKG. Any node that is waiting for authentication must reveal its unique and unchangeable identity that could be its IP address, MAC address or a combination of them. Afterwards, the authenticating node, from direct encounters of t unique PKGs can reconstruct its private key. In the process of getting its private key, the authenticating node should forward both its identity and a self- generated public key to an encountered PKG. Also, Authors evaluated the delay performance of their scheme, studying, on the one hand, the trade-off between security and reliability, and on the other hand the trade-off between security and convenience. Although the proposed scheme can solve major issues such as key escrow problem and single point of failure, how to choose n PKGs, remain a crucial problem for resorting to a third party may raise other concerns.

Christ and Xiaodong [8] proposed a scheme that, with Opportunistic Networking, a mobile phone user finds friends nearby, using both Bluetooth and 3G technologies. Here, friends' identity privacy is protected. The proposed scheme uses three phases to notify friends nearby: system initialisation, notification generation and opportunistic forwarding, notification receiving. A trusted party does the system initialisation. Any user that wants to discover proximity friends must contact the trusted party for authentication. Afterwards, the user sends a packet notification with a time to live to friends. Upon reception and verification of the packet notification, friends choose at will to join the packet sender. For Christ and Xiaodong 'scheme is based on the Decisional Bilinear Diffie-Hellman problem, the proposed scheme is semantically secure under chosen plaintext attack. The scheme performance

Table 2: Users privacy and mutual authentication proposals overview

schemes	Year	Type	Techniques	Pros	Cons
Ma [22]	2010	Authentication	Threshold Secret Sharing+Identity-Based Cryptography	Solve the Key Escrow problem and the Single Point of Failure.	Third Party issue.
Guo [15]	2015	Authentication	Cryptography Principles	Uses Simple Cryptography Principles. Achieves Privacy	Registration at Seed node
Kumar [19]	2017	Authentication	RSA+ Diffie-Hellman Key exchange Protocol	Designed after Guo [15]	Third-party issue
Cao [7]	2014	Authentication	Trust+Multidimensional scaling	Users are considered	Third-party issue
Singh [28]	2017	Authentication	Trust Framework+Guo [15]	Use of Trust Framework	Third-party issue
Carver [8]	2012	Authentication	Decisional Bilinear Diffie-Hellman problem	Achieves Privacy	Third-party issue
Distl [13]	2014	Social Links protection	Contact Graph	Satisfactory result after simulation	Does not scale to more massive graphs
Distl [12]	2015	Social Links protection	Bloom Filter	Satisfactory result after simulation	Designed for free routing
Zakhary [34]	2012	Location protection	Social Links+ K Anonymity technique	Satisfactory result after simulation	The Social Links Problem
Zakhary [35]	2013	Location protection	K-Anonymity technique, lightweight Markov-based location prediction model	Satisfactory result after simulation	The Social Links Problem
Kaur [18]	2015	Identity protection	Dynamic Clustering	Dynamic Concept	Clustering Concept

analysis also shows satisfactory results. However; the role of the trusted party in the scheme is too critical for opportunistic networking aims to promote the direct contact between users.

Ming Huang Guo *et al.* [15] proposed an authentication scheme that also protects users' privacy for Opportunistic Networks. The proposed scheme has two main phases: registration and authentication. Any node or user that wish to communicate with another node should first register at the supernode. The registration process of any unauthenticated node A at the supernode S involves A's virtual identifier ID_a, public key PK_a, secret key SK_a; the supernode's public key $PKsn$, secret key $SKsn$.The supernode uses a symmetric key, an arithmetic function $f()$, and a timestamp Tsn . If the registration is successful, node A can move within the network with its authentication credentials M_j, $f()$, Tsn.

Two nodes A and B that have already completed their registration at the super node can then engage in mutual authentication. The proposed scheme achieves anonymity and privacy due to the techniques used for registration and authentication processes. It also mitigates tapping, forgery, resend, and Man-in-the-middle attacks. Despite an excellent job, the supernode job appears as a major single point of failure. Prashant Kumar *et al.* [19] proposed a scheme for authentication and privacy protection for opportunistic communications. This scheme is a variant of the scheme proposed by Ming Huang Guo *et al.* The proposed scheme stresses the use of RSA and Diffie-Hellman for key generations and key exchange which is useless. Also, the role of the seed node is too much for it generates all the Public and private keys pair for the nodes. The seed node also does the mutual authentication for the nodes because for mutual authentication, nodes look through a list.

4.2 Users Privacy Proposals for OppNets

4.2.1 Social Aware Proposals

An opportunistic contact graph is of great importance for it encoded social information that is used to solve challenging opportunistic networking problems. Considering the trade-off between privacy and utility in the contact

graph, Distl and Hossmann [13] proposed a scheme that changes the contact graph by adding and removing edges.

The algorithm works as follows. First, consider an unweighted and undirected contact graph $G = \{V, E\}$ as input. Second, output a modified contact graph G'={V,E'} such that $|E| = |E'|$. Here, it is hard for an attacker operating on a graph level to know any hidden information in $G = \{V, E\}$. Although the proposed algorithm shows satisfactory results, it does not scale to larger contacts graphs. Considering the importance of users for opportunistic networking, and balancing the trade-off between social links and users privacy, Bernhard Distl and Stephan Neuhaus [12] proposed a scheme that uses the social connections to improve performance without revealing users private information.

The key component of the proposed algorithm is Bloom filters that help achieve privacy for users. Here, Authors are interested in pre-established social links. The algorithm detects social links, uses social links for mutual authentication revealing no personal information. The proposed work is an excellent achievement for it found a way to overturn the concern over social connection into an asset that can be available for other applications. However, more attacker models are yet to be studied.

4.2.2 Location Aware Proposals

Zakhary and Radenkovic [34] were principally interested in how location privacy can influence communications in opportunistic networks. To solve the location matter, they resort to social links. Assuming that users trust their social links, the proposed scheme offers location privacy through request/reply location obfuscation techniques. A user (U_a) that wants a location-based service looks for proximate friends and forwards a copy of their request to an available friend (U_b). With a social forwarding protocol, (U_b) will contribute to help (U_a) achieve his goal without revealing its location. However, the social links privacy was not addressed adequately.

The quest for location –privacy in opportunistic mobile social networks [35] is a variant of a previous scheme that Zakhary and Radenkovic designed. The goal is almost the same. Only the techniques differ. Here, Zakhary *et al.* proposed a stochastic model for location prediction using a lightweight Markov model to drive the privacy protection scheme. The scheme depends on the fact that users trust their contact (friends and relatives) in their social network. The scheme detects users' contact and uses it to obfuscate requests and hide the original sender's location from the location-based service. The proposed work is a collaborative and distributed protocol that offers location K anonymity for each node participates in the anonymisation process. Authors' scheme achieves better than many other protocols. Still, social links privacy should also be considered carefully.

4.2.3 Identity Aware Proposals

Motivated by the fact that opportunistic networks could be of great help if privacy is maintained, Kaur and Singh [18]proposed a scheme that protects users' identity. The proposed scheme relies on dynamic clustering. The algorithm follows the following steps. First, it characterises the network with a finite number of nodes, divides the network into clusters, and generates of cluster heads of each cluster. Second, cluster heads store the information of all its neighbouring nodes, and nodes communicate with each other through the cluster heads. Third, each transmission will be formed along with the new cluster heads. Although the use of dynamic clustering enhances the privacy of the network, the notion of the cluster, on the one hand, an the key role of the base station, on the other hand, do not match opportunistic networks characteristics.

5 Summary

This Section, through table 2 provides a concise and insightful summary of the works studied within Mutual Authentication and Users Privacy classes respectively.

5.1 On Mutual Authentication

5.1.1 Basis on Comparing Mutual Authentication Schemes

OppNets are self-configured and depend on little or no infrastructure with the Seed OppNet as a vital component. On the mutual authentication schemes proposed, this paper, not only described the achievement of those proposals but most importantly compared those proposals concerning the role of the Seed OppNet. For OppNets schemes, it is not desirable for the Seed OppNet to play the role of a Central authority or third party. The (Cons) column in Table 2 gives an idea of the degree of involvement of the Seed OppNet for each scheme.

5.1.2 Summary on Mutual Authentication Schemes

From the works studied within Mutual Authentication class, Xiamei and Ying [7], Ma and Jamalipour [22], and Ming Huang Guo *et al.* [15] impacted the domain Significantly. Other proposals are variant of the works in [15]. Ming Huang Guo et. al used general cryptographic principles to demonstrate the mutual authentication. Xiamei and Ying used trust and multidimensional scaling. Ma and Jamalipour on the other hand, used threshold secret sharing and identity-based cryptography.

5.2 On Users Privacy

5.2.1 Basis on Comparison

As challenging as OppNets are, achieving privacy is tantamount to compromising something. Thus, this paper identified what it took for each proposed scheme to achieve their goal. The (Cons) column in Table 2 gives an idea of the compromise made in each users privacy scheme.

5.2.2 Summary on Privacy Schemes

On Identity protection, Kaur and Singh [18] did a remarkable work using dynamic clustering. Zakhary and Radenkovic [34], [35] did the most work on Location protection using social links. On Social links protection, Distl and Hossmann [13] and Distl and Neuhaus [28] did the most work using contact graph and bloom filter respectively.

6 Conclusion and Future Research Directions

This work clarifies the OppNets concept and Points out the differences between OppNets and some communications models that emerged from Mobile Ad hoc Networks research. What's more, this paper provides a comprehensive survey on users Mutual Authentication in OppNets on the one hand; and Location, Identity, and Social links protection within OppNets on the other hand. The different proposals were organised in a taxonomy.OppNets are the most challenging evolution of Mobile Ad hoc Networks research due to their infrastructure-less nature and their ability to expand from a Seed OppNet.For users represent the heart of OppNets, much effort should be put on Mutual Authentication and privacy protection within OppNets. As future works, an existing multipurpose communication system that can be beneficial to OppNets will be studied and presented. Also, a trust-based mutual authentication mechanism will be proposed.

Acknowledgments

This study was supported by the National Key&D Program under the Grant 2017YFB0802000, and the National Natural Science Foundation of China under the Grant 61370203. The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] M. A. Alduailij and L. T. Lilien, "A Collaborative healthcare application based on Opportunistic resource utilization networks with OVM primitives," in *International Conference on Collaboration Technologies and Systems (CTS'15)*, pp. 426–433, 2015.

[2] A. S. Bleda, R. Maestre, A. Jara, "Ambient assisted living tools for a sustainable aging society in modeling and optimization in science and technologies," *New York : Springer*, pp. 193–220, 2014.

[3] B. H. Bloom, "Space/time trade-offs in hash coding with allowable errors," *Communications of the ACM*, vol. 13, pp. 422–426, 1970.

[4] D. Boneh and M.Franklin, "Identity-based encryption from the weil pairing," *Advances in Cryptology (Crypto'01)*, LNCS 2139, pp. 213 – 229, 2001.

[5] A. Bouchachia, "Dynamic clustering," *Evolving Systems*, vol. 3, no. 3, pp. 133–134, 2012.

[6] A. Z. Broder and M. Mitzenmacher, "Network applications of bloom filters: A survey," *Internet Mathematics*, vol. 1, no. 4, pp. 485–509, 2003.

[7] X. Cao and Y. Yin, "An identity authentication scheme for opportunistic network based on multidimensional scaling," in *International Conference on Cyber-Enabled*, pp. 87-93, 2014.

[8] C. Carver and X. Lin, "A privacy-preserving proximity friend notification scheme with opportunistic networking," in *IEEE International Conference on Communications*, pp. 5387–5392, 2012.

[9] N. Chakchouk, "Communication Networks," vol. 17, no. 4, pp. 2214–2241, 2015.

[10] M. Conti, F. Delmastro, V. Arnaboldi, "People-centric computing and communications in smart cities," *IEEE Communications Magazine*, vol. 54, no. 7, pp. 122–128, 2016.

[11] T. Cox and M. Cox, "Multidimensional Scaling, 1994. (`https://ncss-wpengine.netdna-ssl.com/wp-content/themes/ncss/pdf/Procedures/NCSS/Multidimensional_Scaling.pdf`)

[12] B. Dist and S. Neuhaus, "Social power for privacy-protected opportunistic networks," in *7th International Conference on Communication Systems and Networks (COMSNETS'15)*, pp. 1–8, 2015.

[13] B. Distl and T. Hossmann, "opportunistic network contact graphs," in *Proceeding of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks*, pp. 1–3, 2014.

[14] R. Ganesan, D. Thiyagarajan, "Cryptographically imposed model for efficient multiple keyword-based search over encrypted data in cloud by secure index using bloom filter and false random bit generator," *International Journal of Network Security*, vol. 19, No.3, pp. 413–420, 2017.

[15] M. H. Guo, H. T. Liaw, and M. Y. Chiu, "Authenticating with privacy protection in opportunistic networks Ming-Huang," *EAI International Conference on Heterogeneous, Networking for Quality, Reliability, Security and Robustness (QSHINE'15)*, pp. 375–380, 2015.

[16] T. Hossmann, G. Nomikos, Spyropoulos, and F.Legendre, "Collection and Analysis of Multi-dimensional Network data for Opportunistic Networking research," *Elsevier Computer Communication*, 2012. (`http://www.`

eurecom.fr/en/publication/3751/detail/collection-and-analysis-of-multi-dimensional-network-data-for-opportunistic-networking-research-1)

[17] C. Hu, F. Al-Ayed and H. Liu, "An efficient practice of privacy implementation: Kerberos and markov chain to secure file transfer sessions," *International Journal of Network Security*, vol. 20, no. 4, pp. 655–663, 2018.

[18] P. Kaur and J. Singh, "Ensuring privacy in opportunistic networks using dynamic clustering," in *International Conference on Advances in Computer Engineering and Applications,*, pp. 866–869, 2015.

[19] P. Kumar, N. Chauhan, and N. Chand, "Authentication with privacy preservation in opportunistic networks," in *Proceedings of the International Conference on Inventive Communication and Computational Technologies (ICICCT'17)*, pp. 183–188, 2017.

[20] C. T. Li, M. S. Hwang, "A lightweight anonymous routing protocol without public key en/decryptions for wireless ad hoc networks",*Information Sciences*, vol. 181, no. 23, pp. 5333–5347, Dec. 2011.

[21] C. T. Li, M. S. Hwang, C. Y. Liu, "An electronic voting protocol with deniable authentication for mobile Ad hoc networks," *Computer Communications*, vol. 31, no. 10, pp. 2534–2540, June 2008.

[22] Y. Ma, A. Jamalipour, "Opportunistic node authentication in intermittently connected mobile ad hoc networks," in *16th Asia-Pacific Conference on Communications (APCC'10*, pp. 453–457, 2010.

[23] Y. Ma, "NFC communications-based mutual athentication scheme for the internet of things," *International Journal of Network Security*, vol. 19, No.4, pp. 631–638.

[24] R. Martí, A. Martín-Campillo, J. Crowcroft, E. Yoneki, "Evaluating opportunistic networks in disaster scenarios," *Journal of Network and Computer Applications*, vol. 36, no. 2, pp. 870–880, 2013.

[25] Q. Qian, S. Ni, M. Xie, "Clustering based K-anonymity algorithm for privacy preservation," *International Journal of Network Security*, vol. 19, No. 6, pp. 1062–1071, 2017.

[26] A. Saha and B. Waters, "Fuzzy identity-based encryption," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques* pp. 457-473, 2005.

[27] A. Shamir, "Identity-based cryptosystems and signature schemes," *Advances in Crptohraphy*, LNCS 196, pp. 47–53, 1985.

[28] U. P. Singh and N. Chauhan, "Authentication using trust framework in opportunistic networks," in *8th International Conference on Computing, Communications and Networking Technologies (ICCCNT'17)*, 2017. (https://ieeexplore.ieee.org/document/8203956)

[29] M. Turoff, "The paradox of emergency management," *Conference–Kristiansand (ISCRAM'15)*, 2015. (https://pdfs.semanticscholar.org/d49c/f309f520312fc55d90d11195cbc84b76c738.pdf)

[30] S. F. Tzeng, M. S. Hwang, "Digital signature with message recovery and its variants based on elliptic curve discrete logarithm problem", *Computer Standards & Interfaces*, vol. 26, no. 2, pp. 61–71, Mar. 2004.

[31] Y. Wang, G. Xu, M. Zhang, H. H. Jin, "Research on the topological evolution of uncertain social relations in opportunistic networks," in *IEEE 1st International Conference on Edge Computing*, 2017. (https://ieeexplore.ieee.org/document/8029276)

[32] C. H. Wei, M. S. Hwang, A. Y. H. Chin, "A mutual authentication protocol for RFID", *IEEE IT Professional*, vol. 13, no. 2, pp. 20–24, Mar. 2011.

[33] C. Y. Yang, J. S. Chen, M. S. Hwang, "The capacity Analysis in the Secure cooporative communication System," *International Journal of Network Security*, vol. 19, no. 6, pp. 863–869, 2017.

[34] S. Zakhary, M. Radenkovic, "Utilizing social links for location privacy in opportunistic delay-tolerant networks," in *IEEE International Conference on Communications (ICC'12)*, pp. 1059–1063, 2012.

[35] S. Zakhary, M. Radenkovic and A. Benslimane, "The quest for location-privacy in opportunistic mobile social networks," in *9th International Wireless Communications and Mobile Computing Conference (IWCMC'13)*, vol. 667-673, 2013.

Biography

Cossi Blaise Avoussoukpo is currently a PhD candidate at the University of Electronic Science and Technology of China (UESTC). His research area includes Wireless communications, Opportunistic communications, Cryptography, and Information security.

Chunxiang Xu received her PhD degree from Xidian University in 2004, in P.R.China.She is currently a Professor at the University of Electronic Science and Technology of China (UESTC). Her research area includes cloud computing security, cryptography and information security. She is a member of the IEEE organisation.

Marius Tchenagnon received his MSc degree in computer science from the University of Electronic Science and Technology of China (UESTC). His research area includes Wireless Communications. Opportunistic Communication, cryptography and information security.

Reversible Data Hiding Scheme Based on Fully Exploiting the Orientation Combinations of Dual Stego-images

Xiaofeng Chen[1,2] and Wenlong Guo[1]

(Corresponding author: Xiaofeng Chen)

School of Electronic Information Science, Fujian Jiangxia University[1]

Digital Fujian, Internet-of-Things Key Lab of Information Collection and Processing in Smart Home[2]

Fuzhou 350108, China

(Email: dragon_ball@fjjxu.edu.cn)

(Received Aug. 4, 2018; Revised and Accepted Dec. 7, 2018; First Online Mar. 2, 2019)

Abstract

In order to increase the embedding capacity and achieve good visual quality, this paper proposes a novel reversible data hiding scheme based on fully exploiting the combinations of pixel pair orientations in two stego-images. We labelled these combinations from 0 to 24, each combination representing for embedding a base-25 digit. The embedding capacity of the proposed scheme is approximate to 1.14 bit per pixel (bpp). Since the modification of the cover pixel value is tiny, the two generated stego-images also have a good visual quality of 49.92 dB. Moreover, not any overhead messages are required in this scheme. In the experiment, the proposed scheme outperforms some state-of-the-art methods in terms of measuring in embedding ratio (ER) or peak-signal-to-noise ratio (PSNR). In addition, the proposed scheme has good performance on resisting static attacks on pixel-value differencing (PVD) histogram.

Keywords:Dual Stego-images; Orientation Combinations; Reversible Data Hiding (RDH)

1 Introduction

With the fast development of the Internet, one can transmit information to communicate with people around the world simply by a few clicks or touches on the screen. However, due to the public nature of the network, our transmitted messages can be easily stolen or destroyed by attackers. The most of our concern is focused on enhancing the security of the transmitted data. Cryptography is a traditional method for protecting the confidential data. It will encrypt the to-be-transmitted data into ciphertext by a secret key. The receiver who has the secret key can decrypt the ciphertext to obtain the secret data [11]. Yet, the data after encryption remain out there where they are in a meaningless state that will attract the attention of illegal users. Data hiding is an alternative technique which can conceal the confidential information into a to-be-transmitted image which is referred to as a cover image to obtain a stego-image. Since the difference between the cover image and the stego-image is very small, the human eyes can't tell whether the stego-image contains the secret information or not.

Data hiding can be divided into two categories, *i.e.*, irreversible data hiding [3–5,15,20,21,28,29] and reversible data hiding (RDH) [1,6,10,12,14,16,22,25–27]. The difference between them depends on whether the cover image can be retrieved from the stego-image or not [13]. Numerous RDH methods have been introduced and have been successful in some lossless applications such as military communications and medical cares. Among these RDH methods, difference expansion (DE) [26] and histogram-shift (HS) [22] are the two earliest major techniques. DE was first proposed by Tian in 2003 [26] which calculated the difference of two consecutive pixel values, doubled the result and concealed one secret bit into it. In 2006, Ni [22] first introduced the HS based technique that generated a histogram based on the frequency of each pixel value. The secret data were then embedded into the bin with the highest frequency.

Besides the DE and HS methods which embedded the secret data into the cover image to generate one stego-image, secret sharing [2,24] divided the secret data into n parts and concealed each part into the same cover image to obtain n stego-images. The secret data can be retrieved by the corporation of k or more stego-images, while insufficient number of stego-images can cause the leak of any information about the secret. Dual-image hiding techniques [8,9,17–19,23] can be considered as a special case of the secret sharing when k=2 and n=2. The secret data would not be obtained without two stego-images being processed simultaneously.

Dual image technology has attracted a lot of attention

in recent years. In 2007, Chang *et al.* [8] was pioneered in developing a reversible data hiding scheme by using two steganographic images. In their method, the secret data was first converted into a base-5 numeral system. Then each pixel pair of the cover image was modified to embed two base-5 digits according to the exploiting modification direction (EMD) magic matrix [29]. After processing the whole pixel pair in cover image, two stego-images were obtained. The capacity of their method is almost 1 bit per pixel (bpp) and the quality of generated stego-images can reach 45 dB.

In 2009, Lee *et al.* [18] considered each pixel pair as the center point and embedded two consecutive of two secret bits using the four directions of it to obtain the stego-pixel pair of the two stego-images. In order to implement the reversibility, the orientation relationship between the pixel pairs of the two images was utilized to determine whether the second two secret bits could be concealed in the second stego-pixel pair or not. Though the quality of two setgo-images could reach up to 52 dB, but the payload was no more than 0.75 bpp because only half of the whole orientation relationship could be used for embedding four secret bits.

In 2013, Lee and Huang [17] proposed a novel dual stego-images hiding scheme to increase the embedding capacity. They first converted the secret data into base-5 secret symbols by enhancing base-5 numeral system, every two secret symbols were considered as a set to embed in the identical cover pixel pair to obtain the stego-pixel pair through pre-defined embedding rules. Their embedding capacity was improved to 1.07 bpp.

In 2018, Liu and Chang [19] proposed a dual image hiding scheme based on turtle shell reference matrix. Each pixel in the cover image is duplicated to a pixel pair first. Three secret bits will be embedded when the pixel pair belongs to the back element and only one secret bit will be concealed in the edge type element of the cover pixel pair to obtain two stego-pixel value. The quality of the first stego-image can reach up to 51 dB while the second one can remain at 45 dB and the embedding capacity is almost 1 bpp.

In this paper, we proposed dual image hiding scheme to increase the embedding capacity and exploit the 25 orientation combinations in two stego-pixel pairs to achieve the reversibility. After labelling these combinations from 0 to 24, each combination can represent to embed a base-25 digit into a cover pixel pair. The image quality evaluated by peak-signal-to-noise ratio (PSNR) can reach up to 49.9 dB and the pure payload is around 1.14 bpp. Moreover, the proposed scheme has good performance for resisting static attacks of pixel-value differencing (PVD) histogram.

The rest of this paper is organized as follows. Section 2 describes the method proposed by Lee and Huang, and Section 3 then introduces the proposed scheme. Section 4 summarizes the experimental results and conclusions drawn can be found in Section 5.

2 Review of Lee and Huang's Method

Inspired by the EMD method introduced by Zhang and Wang in [29], Lee and Huang [17] proposed a reversible data hiding scheme by using the orientation combinations of dual stego-images. Pick up a pixel pair (x, y) from cover image, it can be mapped into a two-dimensional space. Each pixel value is ranging from 0 to 255 while the grayscale value is an 8-bits pixel intensity. Furthermore, the first dimension x represents the coordinate value in X-axis and the second dimension y represents the coordinate value in Y-axis. The pixel pair (x, y) is modified to (x_1, y_1) to embed a base-5 digit $m1$, and the other pixel pair (x_2, y_2) will be gained for embedding the second base-5 digit $m2$. When the secret data embedding procedure is implemented, two corresponding stego-images are both constructed. The embedding algorithm is described as below.

The original pixel pair is modified, according to the first pattern, to be the first pixel (x_1, y_1) for embedding $m1$. The value of (x_1, y_1) depends on the value of $m1$, as shown in Figure 1. Once the first secret digit is hidden, $m2$ can be concealed by modifying the original pixel pair according to the second pattern to produce the second pixel pair (x_2, y_2), as shown in Figure 1. The second pattern chosen to embed $m2$ depends on the value of $m1$. For example, if $m1$=3, then the second pattern marked as $P3$ will be selected to conceal $m2$. After the second pattern is determined, the value of (x_2, y_2) will be gained by the value of $m2$.

To explain this algorithm in more details, the embedding rules are created as shown in Table 1. It should be noticed that the overflow and underflow problem will arise in these embedding rules. Since the difference value of d_x (calculated by d_x=x_1-x_2) and d_y (calculated by d_x=y_1-y_2) all range from -2 to 2, Lee and Huang [17] set (x_1, y_1) equal to (x, y) and d_x=3 (or -3) or d_y=3 (or -3) when the overflow and underflow problem occurs.

When two stego-images are acquired, the sequential base-5 digit and the cover image can be retrieved by the following way.

1) Generate the extracting rules as shown in Table 2.

2) Pick up pixel pairs (x_1, y_1) and (x_2, y_2) from the first and the second stego-images, respectively. Calculate the value of d_x and d_y. The two embedded base-5 digits and the cover pixel pair can be determined by Table 2 directly.

3) If d_x or d_y equals to 3 (or -3), these mean that overflow and underflow has occurred. For these situations, no secret data is embedded and the cover pixel value equals to (x_1, y_1).

According to what's mentioned above, the embedding secret data and the cover image can be retrieved without any error. While enhancing the base-5 numeral system,

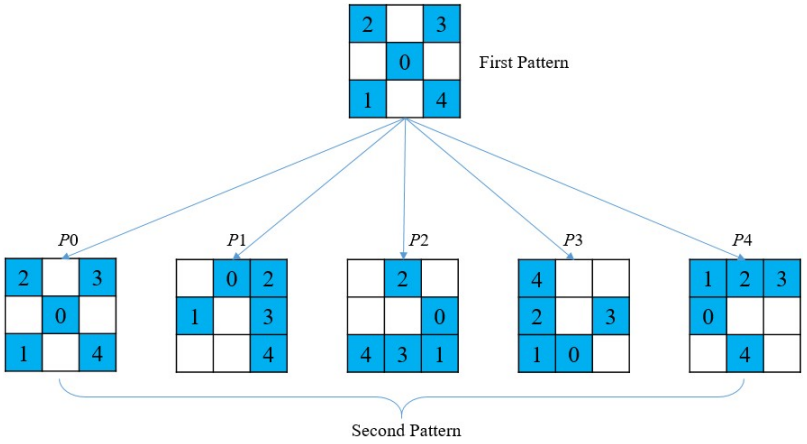


Figure 1: The first pattern and the five second patterns are marked as $P0$, $P1$, $P2$, $P3$ and $P4$

Table 1: The embedding rules of Lee and Huang's method

$m1$	(x_1, y_1)	$m2$	(x_2, y_2)
0	(x,y)	0	(x,y)
		1	$(x\text{-}1,y\text{-}1)$
		2	$(x\text{-}1,y\text{+}1)$
		3	$(x\text{+}1,y\text{+}1)$
		4	$(x\text{+}1,y\text{-}1)$
1	$(x\text{-}1,y\text{-}1)$	0	$(x,y\text{+}1)$
		1	$(x\text{-}1,y)$
		2	$(x\text{+}1,y\text{+}1)$
		3	$(x\text{+}1,y)$
		4	$(x\text{+}1,y\text{-}1)$
2	$(x\text{-}1,y\text{+}1)$	0	$(x\text{+}1,y)$
		1	$(x\text{+}1,y\text{-}1)$
		2	$(x,y\text{+}1)$
		3	$(x,y\text{-}1)$
		4	$(x\text{-}1,y\text{-}1)$
3	$(x\text{+}1,y\text{+}1)$	0	$(x,y\text{-}1)$
		1	$(x\text{-}1,y\text{-}1)$
		2	$(x\text{-}1,y)$
		3	$(x\text{+}1,y)$
		4	$(x\text{-}1,y\text{+}1)$
4	$(x\text{+}1,y\text{-}1)$	0	$(x\text{-}1,y)$
		1	$(x\text{-}1,y\text{+}1)$
		2	$(x,y\text{+}1)$
		3	$(x\text{+}1,y\text{+}1)$
		4	$(x,y\text{-}1)$

Table 2: The exacting rules of Lee and Huang's method

d_x	d_y	$m1$	$m2$	(x, y)
2	2	3	1	$(x_1\text{-}1,y_1\text{-}1)$
	1	3	2	$(x_1\text{-}1,y_1\text{-}1)$
	0	3	4	$(x_1\text{-}1,y_1\text{-}1)$
	-1	4	0	$(x_1\text{-}1,y_1\text{+}1)$
	-2	4	1	$(x_1\text{-}1,y_1\text{+}1)$
1	2	3	0	$(x_1\text{-}1,y_1\text{-}1)$
	1	0	1	(x_1,y_1)
	0	4	4	$(x_1\text{-}1,y_1\text{+}1)$
	-1	0	2	(x_1,y_1)
	-2	4	2	$(x_1\text{-}1,y_1\text{+}1)$
0	2	2	4	$(x_1\text{+}1,y_1\text{-}1)$
	1	3	3	$(x_1\text{-}1,y_1\text{-}1)$
	0	0	0	(x_1,y_1)
	-1	1	1	$(x_1\text{+}1,y_1\text{+}1)$
	-2	4	3	$(x_1\text{-}1,y_1\text{+}1)$
-1	2	2	3	$(x_1\text{+}1,y_1\text{-}1)$
	1	0	4	(x_1,y_1)
	0	2	2	$(x_1\text{+}1,y_1\text{-}1)$
	-1	0	3	(x_1,y_1)
	-2	1	0	$(x_1\text{+}1,y_1\text{+}1)$
-2	2	2	1	$(x_1\text{+}1,y_1\text{-}1)$
	1	2	0	$(x_1\text{+}1,y_1\text{-}1)$
	0	1	4	$(x_1\text{+}1,y_1\text{+}1)$
	-1	1	3	$(x_1\text{+}1,y_1\text{+}1)$
	-2	1	2	$(x_1\text{+}1,y_1\text{+}1)$

$(P_i\text{-}1, P_{i+1}\text{+}1)$	$(P_i, P_{i+1}\text{+}1)$	$(P_i\text{+}1, P_{i+1}\text{+}1)$
$(P_i\text{-}1, P_{i+1})$	(P_i, P_{i+1})	$(P_i\text{+}1, P_{i+1})$
$(P_i\text{-}1, P_{i+1}\text{-}1)$	$(P_i, P_{i+1}\text{-}1)$	$(P_i\text{+}1, P_{i+1}\text{-}1)$

Figure 2: The value of each location in a 3*3 block

8	1	5
4	0	2
7	3	6

Figure 3: The mark of each location in a 3*3 block

their scheme can achieve 1.07 bpp in the embedding capacity. Obviously, the embedding procedure just modifies the original pixel pair to be, at most, plus or minus one, a good visual quality of two stego-images is maintained.

3 Propose Scheme

In this section, we will introduce a novel reversible data hiding scheme to conceal secret data into a grayscale cover image to generate two shadows. It begins with a discussion of the orientation combinations in a 3*3 block, followed by a description of the shadow construction procedure and the data extraction and image recovery procedure of the proposed scheme.

3.1 Orientation Combinations in a 3*3 Block

Suppose we have an cover pixel pair $(P_i,\ P_{i+1})$ and draw a 3*3 block around it. The value of each location is shown in Figure 2. we also mark each location in the block, as shown in Figure 3. It is defined that the smaller mark means a higher priority.

If we embedded some secret data into the pixel pair $(P_i,\ P_{i+1})$ to obtain dual stego-pixel pairs of $(M_i,\ M_{i+1})$ which is denoted as the major one and $(A_i,\ A_{i+1})$ which is denoted as the auxiliary one. Certainly, these stego-pixel pairs are all located within the 3*3 block. In order to achieve reversibility, the orientation combinations of these dual stego-pixel pairs are exploited. To the best of our knowledge, there are a total of 25 combinations which can uniquely determine the center pixel pair $(P_i,\ P_{i+1})$ as shown in Figure 4. Thus, each combination of the dual stego-pixel pairs represents some secret data was embedded in the cover pixel pair. For the convenience, we label these combinations arranging from 0 to 24. The embedding and extracting rules corresponding to these 25 combinations are shown in Table 3. In Table 3, the column labelled as (d_i, d_{i+1}) means the difference between the major and the auxiliary pixel pair which can be acquired by Equation (1). The column labelled as $(P_i,\ P_{i+1})$ is defined as the original pixel pair gained from the major pixel pair in the extracting phase.

$$(d_i,\ d_{i+1}) = (M_i - A_i,\ M_{i+1} - A_{i+1})\ . \tag{1}$$

From Table 3, we can find out that the label of the combination of two dual stego-pixel pairs are ordered by their mark in the 3*3 block. That is, combinations with a smaller major mark is labelled smaller. When two combinations have the same major marks, the smaller auxiliary mark is the smaller label will be. Follow this order, we can create a unique 25 combinations of dual stego-pixel pairs.

According to what's mentioned above, we can embed a base-25 digit into a pixel pair $(P_i,\ P_{i+1})$ to create dual stego-pixel pairs $(M_i,\ M_{i+1})$ and $(A_i,\ A_{i+1})$. Furthermore, the orientation relationship of the dual stego-pixel pairs will depict the embedded digit and the central position where they originated from. The detail of the shadow construction is described in the next section.

3.2 The Shadow Construction Procedure

Assume that a size of R^*C grayscale cover image is divided into a set of pixel pairs $(P_i,\ P_{i+1})$ in raster-scan order, where i$\in$1, 3, $\cdots$, R^*C-1 and a size of n binary stream S is defined as $S = \{s_k\,|k = 1,\ 2, ..., n\}$. We first constructs the table of embedding and extracting rules as mentioned in Section 3.1. Then convert the binary stream into a base-25 digit sequence $S' = \{s'_k\,|k = 1,\ 2, ..., m\}$, where s'_k is arranged from 0 to 24. The converting rules are shown as below:

RULE 1: Get five secret bits $(s_k,\ s_{k+1}, s_{k+2}, s_{k+3}, s_{k+4}\ )$ from binary stream S. Convert them into a decimal value v. If v is less than or equal to 17, then the v is exactly the required digit; Otherwise, read four secret bits $(s_k,\ s_{k+1}, s_{k+2}, s_{k+3}\ )$ from binary stream S. Convert them into the decimal form and increased by "9" to obtain the digit v.

After the sequential of base-25 digits is generated, we will start to process each pixel pair $(P_i,\ P_{i+1})$ in the cover image. If $(P_i,\ P_{i+1})$ belongs to the border, *i.e.*, P_i=0 or P_i=255 or P_{i+1}=0 or P_{i+1}=255, we keep it intact to generate the major pixel pair $(M_i,\ M_{i+1})$ and the auxiliary one $(A_i,\ A_{i+1})$, *i.e.*, M_i=P_i, M_{i+1}=P_{i+1}, A_i=P_i and A_{i+1}=P_{i+1}; otherwise, read a base-25 digit v from S', then rule-v is utilized for creating the two stego-pixel pairs as Equations (2) and (3).

$$(M_i,\ M_{i+1}) \ = \ (M_i,\ M_{i+1})_v\ . \tag{2}$$

$$(A_i,\ A_{i+1}) \ = \ (A_i,\ A_{i+1})_v\ . \tag{3}$$

Where $(M_i,\ M_{i+1})_v$ and $(A_i,\ A_{i+1})_v$ represent the major and auxiliary pixel pairs lying in the embedding rule v in Table 3, respectively.

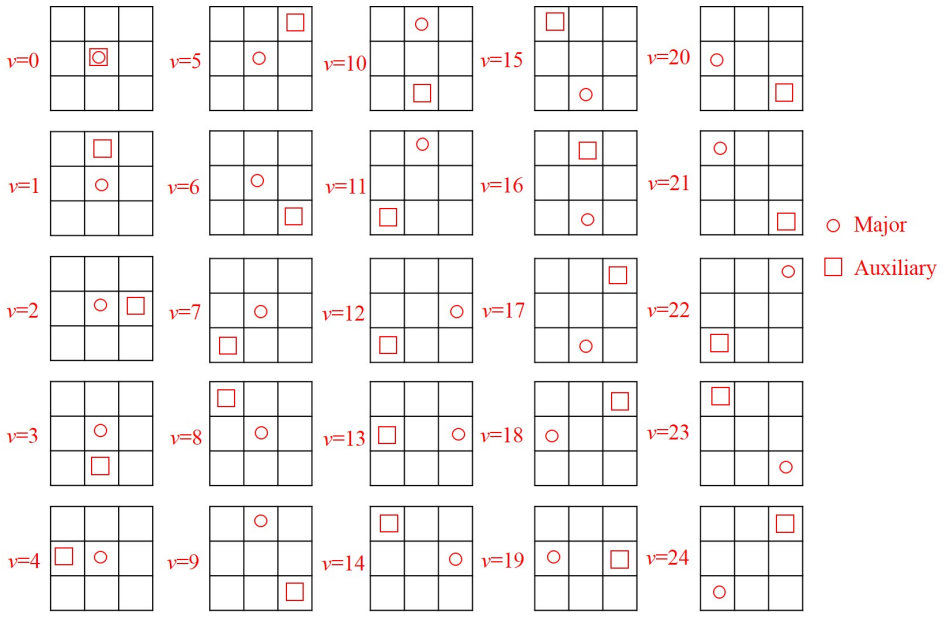


Figure 4: The 25 combinations of dual stego-pixel pairs in a 3*3 block with their labels

Table 3: The embedding and extracting rules of the proposed scheme

v	Secret Bits	(M_i, M_{i+1})	Major Mark	(A_i, A_{i+1})	Auxiliary Mark	(d_i, d_{i+1})	(P_i, P_{i+1})
0	00000	(P_i, P_{i+1})	0	(P_i, P_{i+1})	0	(0,0)	(M_i, M_{i+1})
1	00001			$(P_i, P_{i+1}+1)$	1	(0,1)	
2	00010			(P_i+1, P_{i+1})	2	(1,0)	
3	00011			$(P_i, P_{i+1}-1)$	3	(0,-1)	
4	00100			(P_i-1, P_{i+1})	4	(-1,0)	
5	00101			$(P_i+1, P_{i+1}+1)$	5	(1,1)	
6	00110			$(P_i+1, P_{i+1}-1)$	6	(1,-1)	
7	00111			$(P_i-1, P_{i+1}-1)$	7	(-1,-1)	
8	01000			$(P_i-1, P_{i+1}+1)$	8	(-1,1)	
9	01001	$(P_i, P_{i+1}+1)$	1	$(P_i, P_{i+1}-1)$	3	(0,2)	$(M_i, M_{i+1}-1)$
10	01010			$(P_i+1, P_{i+1}-1)$	6	(-1,2)	
11	01011			$(P_i-1, P_{i+1}-1)$	7	(1,2)	
12	01100	(P_i+1, P_{i+1})	2	(P_i-1, P_{i+1})	4	(2,0)	(M_i-1, M_{i+1})
13	01101			$(P_i-1, P_{i+1}-1)$	7	(2,1)	
14	01110			$(P_i-1, P_{i+1}+1)$	8	(2,-1)	
15	01111	$(P_i, P_{i+1}-1)$	3	$(P_i-1, P_{i+1}+1)$	1	(1,-2)	$(M_i, M_{i+1}+1)$
16	10000			$(P_i+1, P_{i+1}+1)$	5	(-1,-2)	
17	10001			$(P_i, P_{i+1}+1)$	8	(0,-2)	
18	1001	(P_i-1, P_{i+1})	4	(P_i+1, P_{i+1})	2	(-2,0)	(M_i+1, M_{i+1})
19	1010			$(P_i+1, P_{i+1}-1)$	5	(-2,1)	
20	1011			$(P_i+1, P_{i+1}+1)$	6	(-2,-1)	
21	1100	$(P_i+1, P_{i+1}+1)$	5	$(P_i-1, P_{i+1}-1)$	7	(2,2)	$(M_i-1, M_{i+1}-1)$
22	1101	$(P_i+1, P_{i+1}-1)$	6	$(P_i-1, P_{i+1}+1)$	8	(2,-2)	$(M_i-1, M_{i+1}+1)$
23	1110	$(P_i-1, P_{i+1}-1)$	7	$(P_i+1, P_{i+1}+1)$	5	(-2,-2)	$(M_i+1, M_{i+1}+1)$
24	1111	$(P_i-1, P_{i+1}+1)$	8	$(P_i+1, P_{i+1}-1)$	6	(-2,2)	$(M_i+1, M_{i+1}-1)$

After all the pixels are processed, the major stego-image (denoted as M) and auxiliary stego-image (denoted as A) are constructed. For a better understanding of the proposed scheme, an example of the shadow construction procedure is described as follows.

In this example, three cover pixel pairs (0, 2), (5, 6), (8, 8) are used. Suppose the binary stream and the secret stream $(1111\ 00100)_2$ are to be embedded into the pixel pairs. Firstly, we convert binary stream into a base-25 digit sequence of $(24\ 4)_10$ by the converting rule mentioned above. Then, we give more details on how the secret messages are embedded into these pixel pairs to construct two shadows.

1) Since the pixel pair (0, 2) locate in the border, the two stego-pixel pairs are both set to (0, 2) and no secret data is embedded in this case. Thus, the pixel pair of M and A both equal to (0, 2).

2) When embedding the digit 24 into the pixel pair (5, 6). According to Equations (2) and (3), the major and the auxiliary pixel pair are set to $(M_i,\ M_{i+1}) = (M_i,\ M_{i+1})_{24} = (P_i - 1, P_{i+1} + 1) = (4,\ 7)$ and $(A_i,\ A_{i+1}) = (A_i,\ A_{i+1})_{24} = (P_i + 1, P_{i+1} - 1) = (6,\ 5)$. Thus, the pixel pairs of M are (0, 2), (4, 7), while the pixel pairs of A are (0, 2), (6, 5).

3) Follow the similar idea, when embedding the digit 4 into pixel pair (8, 8), According to Equations (2) and (3), the major and the auxiliary pixel pairs are set to $(M_i,\ M_{i+1}) = (M_i,\ M_{i+1})_4 = (P_i, P_{i+1}) = (8,\ 8)$ and $(A_i,\ A_{i+1}) = (A_i,\ A_{i+1})_4 = (P_i - 1, P_{i+1}) = (7,\ 8)$. Finally, the pixel pairs of M are (0, 2), (4, 7), (8, 8), while the pixel pairs of A are (0, 2), (6, 5), (7, 8).

3.3 The Data Extraction and Image Recovery Procedure

In this phase, the receiver can carry out the extraction and restoration by using the corporation of the two shadows. The receiver first generates the identical table of embedding and extracting rules as mentioned in Section 3.1. Then pick up a pixel pair $(M_i,\ M_{i+1})$ from major stego-image M and a pixel pair $(A_i,\ A_{i+1})$ from auxiliary stego-image A at the identical location. The embedded secret bits and the original cover pixel pair can be retrieved by the following way.

1) If M_i=A_i, M_{i+1}=A_{i+1} and M_i=0 or M_i=255, that means the original pixel pair is located at the border, thus P_i=M_i and P_{i+1}=M_{i+1}. Meanwhile, no secret data was embedded in this situation.

2) Calculate d_i and d_{i+1} by Equation (1). The embedding secret base-25 digit v can be determined by $(d_i,\ d_{i+1})$ according to Table 3. Convert v into the binary form to obtain the secret bits, the converting rules is shown as below.

RULE 2: If v is greater than 17, then let v minus 9 and convert the result to a 4-bits binary stream to obtain 4 bits of secret information; otherwise, convert v directly to a 5-bits binary stream to obtain 5 bits of secret information.

Additionally, the cover pixel pair $(P_i,\ P_{i+1})$ is retrieved by

$$(P_i,\ P_{i+1}) = (P_i,\ P_{i+1})_v. \tag{4}$$

Where $(P_i,\ P_{i+1})_v$ represents the cover pixel pair lying in the embedding rule v in Table 3.

After all pixel pairs of two stego-images have been processed, the secret binary stream S and the original cover image can be can be retrieved exactly. Continue the example described in Section 3.2, we will illustrate how to extract the secret data and retrieve the cover image by using the pixel pairs of M(0, 2), (4, 7), (8, 8) and A(0, 2), (6, 5), (7, 8).

1) Obviously, for the identical pixel pair (0, 2) from M and A, the cover pixel pair is located at the border and equals to (0, 2). No secret data was concealed in this situation.

2) Pick up the next pixel pairs (4, 7) and (6, 5) from M and A, respectively. According to Equation (1), the $(d_i,\ d_{i+1})$ can be calculated by $(d_i,\ d_{i+1}) = (4 - 6,\ 7 - 5) = (-2,\ 2)$. Examine $(d_i,\ d_{i+1})$ in Table 3 which can determine that digit 24 is embedded in this situation. According to the *RULE* 2, digit 24 is greater than 17, thus convert (24-9) =15 into a 4-bit binary stream "1111" which is exactly the secret data. Meanwhile, the cover pixel can be retrieved by Equation (4) which is $(P_i,\ P_{i+1}) = (P_i,\ P_{i+1})_{24} = (M_i + 1,\ M_{i+1} - 1) = (4 + 1,\ 7 - 1) = (5,\ 6)$. Thus, the binary secret bit stream S becomes $(1111)_2$ and the cover pixel pairs are (0, 2), (5, 6).

3) Continue to take pixel pairs (8, 8) and (7, 8) into consideration. Calculate the $(d_i,\ d_{i+1})$ by Equation (1) which is $(d_i,\ d_{i+1}) = (8 - 7,\ 8 - 8) = (1,\ 0)$. According to Table 3, we can determine that digit 4 is embedded by the value of $(d_i,\ d_{i+1})$. Secret data "00100" is retrieved by digit 4 according to *RULE* 2. Similarly, the cover pixel can be retrieved by Equation (4) that $(P_i,\ P_{i+1}) = (P_i,\ P_{i+1})_4 = (M_i,\ M_{i+1}) = (8,\ 8)$. Finally, the binary secret bit stream S becomes $(1111\ 00110)_2$ and the cover pixel pairs are (0, 2), (5, 6), (8, 8). The secret data and cover pixel pairs are all retrieved without any error.

4 Experimental Results

The simulation is implemented by Matlab R2012b software on the Intel Core (TM) i5-4210U at 1.70 GHz, 8 GB main memory. Additionally, the operating system is

Windows 7. The binary secret bit stream S is randomly generated using a secret key, *i.e.*, $key1$=2018.

Two conventional measurements are used to evaluate the performance of a data hiding scheme, *i.e.*, the embedding capacity and visual quality of the generated stego-image. In our experiment, the embedding ratio ξ is employed to estimate the embedding capacity (bpp) of a data hiding scheme to carry the pure secret data which is defined as

$$\xi = NUM/\left(2 \times R \times C\right) \ . \tag{5}$$

Where NUM represents the total number of secret bits which are embedded into two stego-images. Additionally, parameters R and C refer to the height and width of the cover image. ξ is divided by 2 because we finally generated two stego-images. Furthermore, the PSNR is used to evaluate the image quality (dB) which is defined as

$$PSNR = 10\log_{10}\left(\frac{255^2 \times R \times C}{\sum_{i=1}^{R}\sum_{j=1}^{C}\left(O_{ij} - SH_{ij}\right)^2}\right) \tag{6}$$

Where O_{ij} and SH_{ij} are referred to the pixels located at the i-th row and the j-th column of cover image O and stego-image SH, respectively.

4.1 Security Enhancement

To enhance the security of the proposed scheme, we generate a random binary bit string which is denoted as $F = \{f_1 f_2 ... f_n,\ f_i = \{0,1\},\ i \in [1,n]\}$, where n represents the number of pixel pairs in original image. Similarly, the binary stream is randomly generated by a secret key, *i.e.*, $key2$=2018. The random bit in private key F is utilized to determine which pixel pair to play what kind of role, *i.e.*, which one is the major and which is the auxiliary. In this paper, we set up that f_i=1 indicates the i-th pixel pair of first stego-image which is considered to be the major pixel pair and the same sequential pixel pair of the second stego-image served as auxiliary. The setting is exactly the opposite to the situation of f_i=0. Figure 5 shows the principle of the enhancement of one cover pixel pair.

4.2 Comparison with Previous Schemes

In the simulation analysis, we employed eight images with size of 512*512 (as shown in Figure 6) to show the embedding capacity and the visual quality of each stego-image of the proposed scheme compared to some previous methods.

To verify the efficiency of the proposed scheme after the security enhancement, comparative results with some state-of-the-art methods in [17–19] measured by and PSNR are given in Table 4 and Table 5, respectively. Notices that, the methods in [17, 18] are using the security enhancement while the scheme in [19] is not.

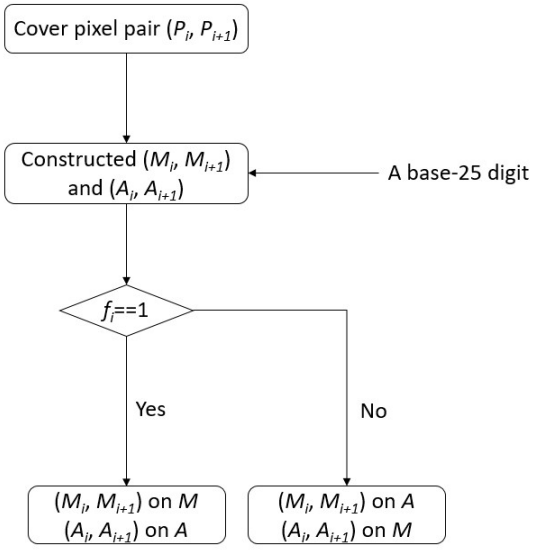


Figure 5: Security enhancement for one cover pixel pair

Figure 6: Eight 512*512 grayscale images

Table 4: Comparative embedding ratio of different schemes

Images	[18]	[17]	[19]	Proposed scheme
Baboon	0.74	1.07	1	1.14
Barbara	0.74	1.07	1	1.14
Lena	0.74	1.07	1	1.14
Pepper	0.74	1.07	1	1.14
Elaine	0.74	1.07	0.99	1.14
Goldhill	0.74	1.07	1	1.14
Airplane	0.74	1.07	1	1.14
Wine	0.74	1.07	1	1.14
Average	**0.74**	**1.07**	**1**	**1.14**

Table 5: Comparative image qualities of the shadows of different schemes

Images	[18]		[17]		[19]		Proposed scheme	
	M	*A*	*M*	*A*	*M*	*A*	*M*	*A*
Baboon	52.47	52.47	49.38	49.38	51.72	45.71	49.91	49.92
Barbara	52.47	52.48	49.38	49.38	51.73	45.70	49.91	49.92
Lena	52.47	52.48	49.38	49.38	51.69	45.70	49.91	49.92
Peppers	52.47	52.48	49.38	49.38	51.73	45.70	49.91	49.92
Elaine	52.47	52.48	49.38	49.38	51.84	45.66	49.91	49.92
Goldhill	52.47	52.48	49.38	49.38	51.72	45.71	49.91	49.92
Airplane	52.47	52.48	49.38	49.38	51.68	45.73	49.91	49.92
Wine	52.47	52.47	49.38	49.38	51.67	45.72	49.91	49.92
Average	**52.47**	**52.48**	**49.38**	**49.38**	**51.72**	**45.70**	**49.91**	**49.92**

From Table 4, it is shown that the proposed scheme acquires the best embedding ratio ξ of 1.14 bpp. The gains of the average ξ of the proposed scheme are 0.4 bpp and 0.07 bpp, and 0.14 bpp compared to these three comparative schemes in [17–19], respectively.

In Table 5, we focused on the average level. Though the scheme in [18] gains the best visual quality of up to 52.48 dB, his embedding ratio ξ is smaller than the proposed scheme of 0.4 bpp. The results also show that the proposed scheme achieves a PSNR of 0.55 dB higher than [17]. While comparing to [19], the proposed scheme has 1.81 dB lower in the main image, but 4.22 dB higher for the auxiliary one.

4.3 PVD Histogram Analysis

The robustness of the proposed scheme can be measured by PVD histogram. It creates a histogram through the different value of two consecutive pixels. The closer the shadow PVD histogram is to the cover PVD histogram, the better the scheme is. Figure 7 shows the PVD histograms of the four cover images (a)-(d) and their shadows. Obviously, the shape of the cover PVD histogram is well preserved on two shadows.

5 Conclusions

This work exploits the combinations of pixel pair orientations in two stego-images. There are at most 25 combinations which can uniquely determine where the two stego-pixel pair are originated from. While labelling these combinations from 0 to 24 by a particular order, each combination can represent the embedding of a base-25 digit and no overhead message is needed for secret data extraction and original image recovery. The stego-pixel pairs are only altering the cover value by at most plus one or minus one, so the generated stego-images can achieve high image quality. Experimental results indicate that a high embedding capacity of 1.14 bpp can be achieved in the proposed scheme and the average image quality of 49.92 dB of the two stego-images is gained. Moreover, the proposed scheme can resist the static attacks on PVD histogram.

Acknowledgments

This study was supported by Fujian Province Education and Science Foundation of young teachers (JAT170619). The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] A. M. Alattar, "Reversible watermark using the difference expansion of a generalized integer transform," *IEEE Transactions on Image Processing*, vol. 13, no. 8, pp. 1147-1156, 2004.

[2] G. R. Blakley, "Safeguarding cryptographic keys," in *Proceedings of American Federation of Information Processing Societies National Computer Conference*, pp. 313–317, Nov. 1979.

[3] C. K. Chan and L. M. Cheng, "Hiding data in images by simple lsb substitution," *Pattern Recognition*, vol. 37, no. 3, pp. 469–474, 2004.

[4] C. C. Chang, Y. C. Chou and T. D. Kieu, "An information hiding scheme using sudoku," in *Proceedings of Third International Conference on Innovative Computing, Information and Control*, June 2008.

[5] C. C. Chang, Y. J. Liu and T. S. Nguyen, "A novel turtle shell based scheme for data hiding," in *Proceedings of Tenth International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pp. 89–93, Aug. 2014.

[6] C. C. Chang, Y. C. Chou and T. D. Kieu, "Information hiding in dual images with reversibility," in *Proceedings of the Third International Conference on Multimedia and Ubiquitous Engineering*, pp. 145–152, June 2009.

[7] C. C. Chang, M. S. Hwang, "Parallel computation of the generating keys for RSA cryptosystems", *Electronics Letters*, vol. 32, no. 15, pp. 1365–1366, 1996.

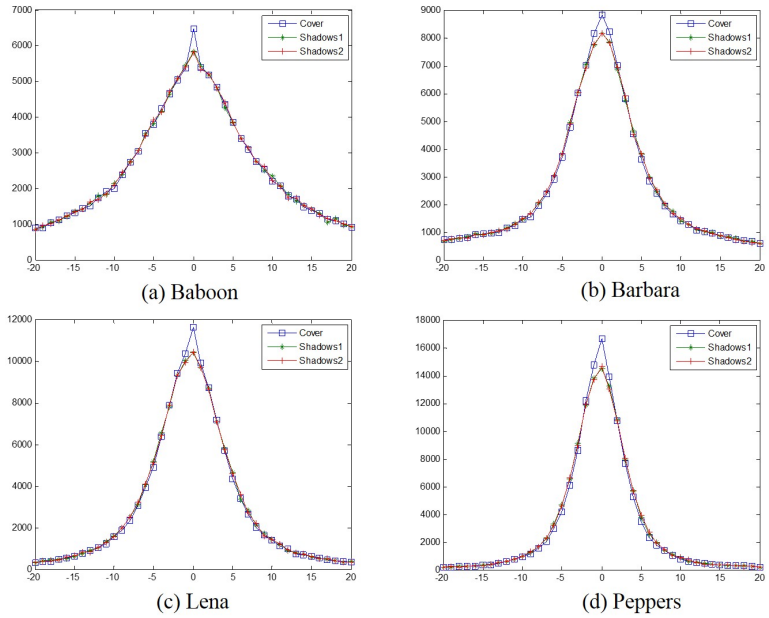


Figure 7: Four PVD histograms of the cover images and their shadows

[8] C. C. Chang, T. D. Kieu and Y.C. Chou, "Reversible data hiding scheme using two steganographic images," in *Proceedings of IEEE Region 10 International Conference (TENCON'07)*, pp. 1–4, Nov. 2007.

[9] C. C. Chang, T. C. Lu, G. Horng, Y. H. Huang and Y. M. Hsu, "A high payload data embedding scheme using dual stego-images with reversibility," in *Proceedings of the Third International Conference on Information, Communications and Signal Processing*, pp. 1–5, Dec. 2013.

[10] I. C. Chang, Y. C. Hu, W. L. Chen and C. C. Lo, "High capacity reversible data hiding scheme based on residual histogram shifting for block truncation coding," *Signal Processing*, vol. 108, pp. 376–388, 2015.

[11] T. Gulom, "The encryption algorithms GOST28147-89-IDEA8-4 and GOST28147-89-RFWKIDEA8-4," *International Journal of Electronics and Information Engineering*, vol. 8, no. 1, pp. 20–31, 2018.

[12] L. C. Huang, L. Y. Tseng, M. S. Hwang, "A reversible data hiding method by histogram shifting in high quality medical images", *Journal of Systems and Software*, vol. 86, no. 3, pp. 716–727, Mar. 2013.

[13] L. C. Huang, L. Y. Tseng, M. S. Hwang, "The study on data hiding in medical images", *International Journal of Network Security*, vol. 14, no. 6, pp. 301–309, 2012.

[14] B. Jana, "Dual image based reversible data hiding scheme using weighted matrix," *International Journal of Electronics and Information Engineering*, vol. 5, no. 1, pp. 6–19, 2016.

[15] H. J. Kim, C. Kim, Y. Choi, S. Wang and X. Zhang, "Improved modification direction schemes," *Computer and Mathematics with Applications*, vol. 60, no. 2, pp. 319–325, 2010.

[16] C. F. Lee, J. J. Li, Y. H. Wu and C. C. Chang, "Generalized pvo-k embedding technique for reversible data hiding," *International Journal of Network Security*, vol. 20, no. 1, pp. 65–77, 2018.

[17] C. F. Lee and Y. L. Huang, "Reversible data hiding scheme based on dual stegano-images using orientation combinations," *Telecommunication Systems*, vol. 52, no. 4, pp. 2237–2247, 2011.

[18] C. F. Lee, K. H. Wang, C. C. Chang and Y. L. Huang, "A reversible data hiding scheme based on dual steganographic images," in *Proceedings of the Third International Conference on Ubiquitous Information Management and Communication*, pp. 228–237, Jan. 2009.

[19] Y. Liu and C. C. Chang, "A turtle shell-based visual secret sharing scheme with reversibility and authentication," *Multimedia Tools and Applications*, vol. 77, no. 19, pp. 25295–25310, 2018.

[20] Y. J. Liu, C. C. Chang and T. S. Nguyen, "High capacity turtle shell-based data hiding," *IET Image Processing*, vol. 10, no. 2, pp. 130–137, 2016.

[21] J. Mielikainen, "Lsb matching revisited," *IEEE Signal Processing Letters*, vol. 13, no. 6, pp. 1129–1143, 2009.

[22] Z. Ni, Y. Q. Shi, N. Ansari and W. Su, "Reversible data hiding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 3, pp. 354–362, 2006.

[23] C. Qin, C. C. Chang and T. J. Hsu, "Reversible data hiding scheme based on exploiting modification direction with two steganographic images," *Multimedia Tools and Applications*, vol. 74, no. 15, pp. 5861–5872, 2015.

[24] A. Shamir, "How to share a secret," *Communications of the Association for Computing Machinery*, vol. 22, no. 11, pp. 612–613, 1979.
[25] J. J. Shen, Y. L. Wang and M. S. Hwang, "An improved dual image-based reversible hiding technique using lsb matching," *International Journal of Network Security*, vol. 20, no. 4, pp. 801–804, 2018.
[26] J. Tian, "Reversible data embedding using a difference expansion," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 8, pp. 890–896, 2003.
[27] P. Y. Tsai, Y. C. Hu and H. L. Yeh, "Reversible image hiding scheme using predictive coding and histogram shifting," *Signal Processing*, vol. 89, no. 6, pp. 1129–1143, 2009.
[28] D. C. Wu and W. H. Tsai, "A steganographic method for images by pixel-value differencing," *Pattern Recognition Letters*, vol. 24, no. 9-10, pp. 1613–1626, 2013.
[29] X. P. Zhang and S. Z. Wang, "Efficient steganographic embedding by exploiting modification direction," *IEEE Communications Letters*, vol. 10, no. 11, pp. 781–783, 2006.

Biography

Xiaofeng Chen. Received the B.S degree from Fujian Normal University in 1999 and the M.S degree from Fuzhou University, Fujian, China, in 2008. He is currently a lecturer with the School of Electronic Information Science, Fujian Jiangxia University. His research interests include information hiding, privacy protection and quantum cryptography.

Wenlong Guo. Received the B.S degree from Southwest University for Nationalities, Sichuan, China, in 2002 and the M.S degree from Fuzhou University, Fujian, China, in 2011. He is currently an associate professor with the School of Electronic Information Science, Fujian Jiangxia University. His research interests include information hiding , cryptography and data mining.

A Multibit Representation of Bloom Filter for Simultaneous Acquisition of Membership and Attribute Information

Ying-Chih Tseng[1,2] and Heng Ma[3]
(Corresponding author: Heng Ma)
Department of Obstetrics and Gynecology, Hsinchu Cathay General Hospital, Hsinchu, Taiwan[1]
Department of Information Management, Yuanpei University of Medical Technology, Hsinchu, Taiwan[2]
Department of Industrial Management, Chung Hua University, Hsinchu, Taiwan[3]
707, Sec.2, WuFu Rd., Hsinchu, 30012 Taiwan
(Email: hengma@chu.edu.tw)
(Received Sept. 10, 2018; Revised and Accepted Apr. 13, 2019; First Online Aug. 10, 2019)

Abstract

As dataflow on Internet is growing exponentially, processes that can efficiently extract meaningful information have become a crucial factor for many successful applications. For example, the purpose of membership determination is to discriminate whether a fragment of the dataflow is an element of a specific dataset. The Bloom filter has been well recognized for dealing with such a problem, but it can only provide the membership information. Recently, membership determination that can accompany with additional attribute information has becoming increasingly important, because it could save considerable time for the secondary querying once the membership is confirmed. Therefore, this study proposes a multibit representation of the original Bloom filter by encoding the attribute codes, instead of binary counterpart, for resolving such a situation. Simulation results show that the querying efficiency and false-positive ratios are fairly competitive with reasonable memory-space usages.

Keywords: Attribute Information; Bloom Filter; Membership Determination

1 Introduction

Emerging business models on the Internet, such as blogs, social groups, instant messaging and on-line shopping, have brought the world to a completely different look in recent years. These models usually require unique processes for efficiently manipulating information in real time, such as keyword searching, frequency measuring and account-password matching for accommodating the growing Internet speed. Membership determination could be accounted for a branch of such processes, which could be characterized as: Given a large dataset composed of elements of a certain feature, *e.g.*, login accounts or email addresses, the purpose is to discriminate whether a random query is an element of the dataset. The size of the dataset could grow even larger as time proceeds, and the expansion, however, could influence the querying time to some degrees, so a means that can perform such a process without compromising the querying speed is very critical.

The Bloom filter [2] has been widely employed as a core engine for dealing with the membership determination problem, which includes two phases of processes. In the programming phase, a set of hash functions are utilized for mapping each set element to a one-dimensional bit array. Because of the employment of the hashing functions, in the second phase the querying time could achieve nearly constant regardless the size of the dataset. The bit array is considerably large, in which each bit is initially set to zero and turned to one once hit by the hash process. Such a programming behavior guarantees that all the set elements could get all ones in the querying phase to be considered as a member. Such a process, however, could introduce errors called "false positives", where all the cells hit by a non-member element are all ones. Consequently, the element is falsely determined as a member, which is also referred to as false-positive (FP) errors.

The FP errors could be critical for some applications that requires stringent membership discrimination, which must be within an acceptable level. In [9], the authors show that it is important how such a matter could influence cloud computing, in which they point out that attribute-based encryption is a promising cryptographic solution in cloud environment because access control is important as far as data security is concerned. For example, Thiyagarajan and Ganesan [14] proposes an architecture of multiple keyword search by building index using Bloom filter, taking advantage of its consistent pro-

cessing time for querying through hash functions. The engagements of hash functions, due to their uniqueness have also given themselves a crucial role for defending malicious intrusions, also referred to as intrusion detection systems (IDS), as in [1,12]. For our proposed method, we employ a special yet simple hash function, designed for elements of texts and symbols with various lengths. Such a design was intended to accommodate packet headers on the network, making recognition of known malicious packets with certain characteristics possible.

The two-phase mechanism of Bloom filter is very similar to the process of artificial neural network (ANN), where a certain large amount of known data are iteratively presented to a specific paradigm, and the results are utilized as an on-line component for determining the outcome of inquiring data. In our previous work, an ANN paradigm (CMAC) was employed for resolving the issue of membership determination with multiple attributes [11]. Being a supervised type of ANN, a number of target values of CMAC must be specified for the paradigm to operate properly. In our implementation, each target value designated a specific attribute code. When the training process is complete, all the set data belonging to an identical attribute code would approach that target value, where a recognition zone could be formed for a querying element to be considered as member if they pass through the zones. Furthermore, the associated attribute code could be immediately identified. The scheme works well in simultaneously obtaining membership and the associated attribute information, while the FP errors could also be kept at an acceptable level; however, ANN requires extensively computational time for converging the recognition zones to a sufficiently small width for low FP errors, which is not suitable for dynamic membership insertion. Furthermore, ANN uses floating-point numbers to represent the content of cells in the array, which could consume considerably large memory in real time. In this paper, we change the contents of the array to integer numbers as the attribute codes. Although multiple bits are still required for each cell, the memory space is far less than the CMAC-based approach. Our objective is to demonstrate a feasible approach for applications that require membership determination with simultaneous attribute information. We focus on the computational efficiency instead of hardware implementations because hardware technology is progressing from time to time. Nevertheless, we still take the memory overhead as a crucial factor in the proposed approach. In the following, we present recent efforts in the literature for the addressed problem.

The difficulty of the addressed problem lies in that additional structures or computational components could be involved in the determination process for providing the attribution information at the same time. Intuitively, using a set of parallel Bloom filters could solve the problem, where each Bloom filter could represent an attribute code. The main drawback of such an approach has been that the size of each Bloom filter is difficult to decide because the number of elements for each attribute code could vary to some extent. Using the same size of the parallel Bloom filters could cause dramatic memory waste. Furthermore, a querying element must go through all the filters, which is not only a time-consuming process but could also lead to extra errors when multiple filters respond true membership. In [15], variants of the original Bloom filter were proposed for dealing with multi-attribute membership problem. In this approach, PBF (Parallel-Bloom filter) is responsible for defining attributes for a number of counting parallel Bloom filters when an element could be associated with multiple attributes. PBF-HT (PBF with a Hash Table) and PBF-BF (PBF with a Bloom Filter) were designed for verification purposes, whose objective was to compensate the extra false positive errors that could be introduced by PBF. Consequently, the number of attribute codes could become very large in particle applications. Although PBF-HT and PBF-BF could compensate time and space losses, the memory usage and additional false-positive errors could lead PBF to an unfit situation. In 2012, the concept of approximate membership query (AMQ) [8] was raised based on its previous work [15]. The AMQ is an approach referred to as that the degree of an element is within a certain range of a member's boundary. This approach could be meaningful for some prediction models especially in networking, however, the addressed problem could not be resolved by this approach because each element is associated with only one attribute, and the membership allows no ambiguous regions.

Due to the success of the original Bloom filter for a variety of applications, a number of approaches tackled the addressed problem with modified architectures. For example, the invertible Bloom lookup table proposed in [6] is to provide key-value pairs upon querying. The data structure was designed to accommodate both keys and values of integers, where inserting an element would always be successful because distinct keys are used. In the querying mode, an inquiring element x would receive a value y that composes a pair with x, and then information could be obtained given that y is not null, which is an extra step with conditions in obtaining the attribute information. As it becomes increasingly important to acquire the membership with its additional attribute information at the same time, especially in the networking applications, the following work had been done in an effort to resolve this problem under a packet routing situation. The routing mechanisms have been a key factor for keeping the Internet not only fluent but healthy, which requires accurate and timely dispatching of millions of arriving packets in a fraction of second. In [7], the authors proposed combinatorial Bloom filters, in which a considerably large group of hashing functions were employed. They defined a unique binary vector code for each hashing group for differentiating the attributes; therefore, an inquiring element must go through all the hashing groups before being sent to the general Bloom filter for the membership determination. In this approach, different hashing group combinations represent a certain attribute, but the size

of the binary vector for hashing could be very long to accommodate a large number of attributes. Qiao *et al.* [13] use two data structures for determining membership and the ID information. The first one, index filter, is a general Bloom filter, whose purpose is solely for membership determining, while the second one, the set-id table, stores the ID information for each member using multiple hash functions. The process proceeds when one of the multiple hash functions hit a zero in the set-id table, the ID information is then inserted into that cell and the identical hash function is used for coding the index filter. In the querying phase, the membership with the ID information could be obtained if one hash function reports one on the index filter and an ID on the set-id table. In this method, the lengths of both structures must be sufficiently large to accommodate a good portion of zeros for better performances.

More recently, using multiple bits instead of a single bit for each cell of the Bloom filter has been proposed to incorporate the set IDs. Xu *et al.* [16] suggested encoding both information in the same data structure could provide more efficient query processing speed. Therefore, they proposed multi-bit array, where insertion and lookup procedures could be achieved by bitwise operations including union and intersection. The approach, however, could result in additional false-positive errors when the set number is fairly large. Although several remedy techniques were proposed, they still compromise the lookup speed by dividing the original structure into several levels. Dai *et al.* [4] also proposed multi-bit structure for encoding the set IDs, where bitwise operations were employed for the insertion and lookup purposes. This approach, unlike [16], was dealing with determining membership of multiple disjoint sets; however, it also suffered from errors at the lookup phase because any query was given a set ID response, which could result in additional false-positive errors. The ID Bloom filter [10] shares the same bitwise operations for element insertion and lookup procedures as in [4], so it could also inherit the chances of misjudgment in the lookup phase. The approach designated the set IDs with decimal numbers instead of bit streams, whose purpose was to save the number of memory accesses in the querying phase; however, using the bitwise operations seems to lack scalability because it is difficult to encode a large number of sets into the array structure.

In this paper, we adopt the concept that both membership and attribute information are coded in the same array to avoid additional structures that consume memory usage, remain the querying speed as in the original Bloom filter, and keep the false-positive ratio within an acceptable level. The multibit representation allows each cell in the array to designate a number of attribute codes. Programming for both membership and attribute information in the same array is a better approach because it does not require a great number of hash functions. Secondly, fast querying could be achieved because the number of memory accesses is mainly dependent on the hash number. Furthermore, the false-positive rate is anticipated to be acceptable because only a single array is addressed by the hashing process. The multibit representation suggests that the attributes would be coded as integer numbers, so the input data in the programming phase include member elements and their associated attribute codes.

2 Methodology

In the proposed approach, since each cell of the array is represented by multiple bits for designating various attribute codes, we must first determine the number of bits for each cell to best conserve the memory space. Of the integer numbers that the multiple bits could represent, we reserve 0 and the largest one for the use of the programming as well as querying phases. For example, if 4 bits are used in each cell, there are 16 integer numbers ranged from 0 to 15, in which 0 and 15 would be reserved, and the remaining 1 14 could be used as attribute codes. Consequently, the relationship between the number of bits in a cell and the number of attribute codes could be characterized as in Equation (1).

$$b = \lceil \log_2(a+2) \rceil \tag{1}$$

where b is the number of bits in each cell, a is the total amount of attribute codes, $\lceil \cdot \rceil$ is the ceiling operator that finds the smallest integer greater or equal to the content.

2.1 The Proposed Approach

The original Bloom filter turns a cell's value from 0 to 1 for those hash hit by all the member elements, which is an important process for expediting membership determination in the querying mode. In the proposed approach, however, since the array is composed of attribute codes instead of binary ones, a specific procedure is employed for programming to achieve simultaneous retrievals of membership and attribute information in the querying phase. However, it is not an easy task because if an element's attribute code is assigned to all the hash hit cells, "collision" could happen, where multiple elements with different attribute codes hit the same cell. Therefore, we introduce two techniques for resolving such a situation: (1) the universal code U using the reserved largest number to represents all the attribute codes, and (2) the threshold rate T to pass for being considered as a member with the associated attribute code. In a sense, (1) is to ensure that the programming process could successfully proceed, while (2) is to save as much memory space as possible. We use examples in the following to further explain how these two techniques work.

As U designates all the attribute codes, the T is the hurdle to pass for being considered as a member, the hash number is h and the total attribute number is a, we describe the membership discrimination rules in the querying phase. The rules are designed to make the querying phase as fast as possible, including:

1) If zero is encountered, it is not a member;

2) If there are all U's, it is not a member;

3) Find the largest count c of the hashed cells of the same attribute code a plus the number of U, and if $c/h \geq T$, the element is recognized as a member of the attribute code a; otherwise, it is not a member.

We further deliberate the procedure with the following four examples with $h = 5$, $a = 8$ and $T = 0.5$, where the sequence is the attribute codes of the hash hit cells by an element is:

1) 2, 0, 3, 5, 6. The sequence includes zero, so the element is immediately rejected for being a member;

2) U, U, U, U, U. The sequence contains all U's, so it is not a member;

3) 4, 2, U, 4, 8. The largest count $c = 3$ with $a = 4$, and element is considered as a member with the attribute code of 4 because $3/5 \geq T$;

4) 3, 5, 6, 8, 6. $c = 2$ with $a = 6$, but $2/5 < T$, so the element is rejected;

5) 3, 4, 4, 3, U. There is a tie between $a = 3$ and 4, and both surpass T, so it's ambiguous and no conclusion is drawn.

With the discrimination rules, we now describe the programming process in the proposed approach.

It's important to describe the hash functions employed in the proposed approach before we go to the programming process. We call it "sequential hashing", which was inspired by [5] that suggests a simple logarithm function disregarding the integer and the first few digits of the decimal parts could form an effective hash function. In our implementation, we employ such a concept with some specific factors including the sum and the accumulated sum the element, the array size m and a position indicator for the array. When the position indicator exceeds m during the hashing process, it is set to the remainder position divided by m. The proposed hashing function is easy to implement and with operational efficiency, and could produce any specified h hash numbers.

Because collisions could happen at any cell through the hashing process, we investigate the magnitude of the collision for each cell, where a pilot run is executed for all the member elements with their attribute codes. The information is recorded in a "hit table" as in Table 1, which is particularly useful for allocating appropriate attribute codes for the array. The first column of the table is the sequential cell number up to the array size m, the second one is the element(s) that hit the cell with the attribute code in the parentheses, and the last column shows the hit frequency of the cell by elements with different attribute codes. Although the table claims memory space, it is only required in the programming phase.

The hit table could somehow depict our programming procedure whose purpose is to determine the arrangement

Table 1: The hit table in the programming phase

C_1	$e_1(5)$	1
C_2		0
C_3	$e_2(8)$	1
C_4	$e_1(5)$	1
C_5	$e_1(5)$	1
C_6	$e_1(5), e_2(8)$	2
C_7	$e_2(8), e_3(2)$	2
C_8	$e_2(8)$	1
C_9	$e_1(5), e_3(2)$	2
C_{10}	$e_2(8), e_3(2)$	2
C_{11}	$e_3(2)$	1
C_{12}	$e_3(2)$	1
⋮	⋮	⋮

of the attribute codes in the array, allowing all the elements to pass the threshold rate T with the smallest number of U. Therefore, we start with cells with the lowest hit frequency, and move the way up by excluding those elements that are already qualified as a member. We demonstrate this concept with a simplified example in Figure 1.

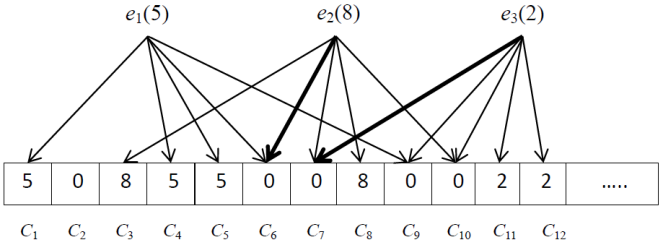


Figure 1: A simplified example for the proposed programming procedure

In Figure 1, like the original Bloom filter, all the cells in the array are initially set to zeros, designating the cell is unused. In the example, the hash number h is 5, meaning each element hits 5 cells after the hashing process. Supposing the threshold rate T is 0.5, at least 3 cells with the same attribute code or the universal code must be hit (3/5=0.6) for the element to be admitted as a member with that attribute code. Consulting the hit table in Table 1, the second column shows the elements that hit this cell with different attribute codes. Therefore, the first step of the programming procedure is to fill the attribute code to those cells with a single hit. For example, C1 is only hit by e1, whose attribute code 5 is then settled in C1, so are the other cells with only one hit. When a cell is hit by multiple elements, *e.g.* C6, since e1 is already qualified as a member but e2 is not, the attribute code 8 of e2 would be issued for C6. The thick arrow in Figure 1 represents the winner of the cell. After C6 is settled with the attribute code 8, the next cell with multiple hits is C7,

which would be set to the attribute code of e3 because e2 has become a member. For both C9 and C10 cells, since both hit elements are members, a randomly selected attribute code of the hit elements would be assigned to the cell. When no more attribute code is inserted into the array after an iteration, the uncertain cells are stored with the universal code U.

2.2 Theoretical Analysis

In this section, we provide theoretical analysis for the proposed approach, especially on the universal code U and the threshold rate T because they play key roles for programming effectiveness and memory conservation to the addressed problem. More specifically, the universal code U could solve collisions in the programming process, and thus keep the array at a reasonable size. However, it is conceivable that a large number of U could greatly increase the false-positive errors. The threshold rate T is to conserve the memory space because it represents the fraction of the hash cells of an element to be considered as a member. The fraction must contain either the element's attribute code or U. we suggest that it should be set at least 0.5 to avoid excess false-positive errors. Although T is pre-specified, its value could substantially affect the number of U, meaning a large T could dramatically increase the number of U. Therefore, we investigate the occurrence of U with regard to T as well as other related factors. The notations of these related factors are given as follows: the array size m, the number of attribute codes a, the number of elements with each attribute n, and the hash number h. We assume the number of elements n is the same for all attribute codes for the preliminary analysis.

The universal code U is only engaged when a cell is hit by multiple elements, and the probability of a cell hit by multiple elements is as in Equation (2).

$$P_x = 1 - P_0 - P_1, \tag{2}$$

where P_x is the probability of multiple hits, P_0 and P_1 are that of zero and single hit respectively.

Let H be the total number of hash hits by all the elements, so $H = a \cdot n \cdot h$. Therefore, we could derive P_0 and P_1 as in Equations (3) and (4).

$$P_0 = \frac{\binom{m}{1}}{\binom{H}{0}} = \frac{m}{H} \tag{3}$$

$$P_1 = \frac{\binom{m}{1}}{\binom{H}{1}} = \frac{m}{H}. \tag{4}$$

We then rewrite Equation (2) as in Equation (5).

$$P_x = 1 - \frac{2m}{H}. \tag{5}$$

P_x includes a portion of cells, where only one or none element does not meet T, which must be excluded out because they would not be represented by U. Therefore, we derive the probability of each element that could actually turn to U as in Equation (6).

$$P_y = \sum_{i=\lceil T \cdot h \rceil}^{h} \frac{1}{\binom{h}{i}\binom{a}{1}} = \sum_{i=\lceil T \cdot h \rceil}^{h} \frac{i!(h-i)!}{a \cdot h!}, \tag{6}$$

where P_y is the probability that the result of an element's hash hits meets T.

Let j be the hit number for each cell, and the probability of U could be designated as in Equation (7).

$$P_u = P_x \cdot (1 - \sum_{j=2}^{H} P_y^j). \tag{7}$$

In Equation (7), the second term in the parentheses represents the probability of not getting pass as a member when the number of hit elements is j.

Using Equation (7), when $h = 6$, $a = 100$, $n = 1,000$, $m = 40,000$, $T = 0.5$, P_x would be $2/15 = 0.1333$ according to Equation (5), and P_y would be approximately 0.1283 according to Equation (6). So the probability of U for each cell P_u is approximately 0.13.

3 Experimental Results

Normally, the performance metrics for the original Bloom filter or its variants include querying time, memory space and false-positive rate. For the proposed approach, we replace the bit array with a multibit one associated with proposed procedures for providing simultaneous attribute information when the membership is true Therefore, the time metric would not be a concern because the number of memory accesses is the same with the original Bloom filter, depending on the hash number. Additional operations such as counting the attribute codes and comparing with the threshold rate only accounts for a small fraction of the computational time. Under such a circumstance, we consider the memory space utilized for the array and the threshold rate as indicators for evaluating the false-positive rate in this section. The data element of the experiments was email addresses, which represent text strings with varied lengths. The dataset included 300,000 elements, while 100,000 others (outside the dataset) were used for evaluating the false-positive rates. Each of the set elements was assigned to a random attribute code between 1 and 14 since we used 4 bits (code values $0 \sim 15$) for each cell of the array. The value 0 designated the unused cells and 15 was the universal code that could represent all the attribute codes 1 14.

The memory space was designated by the number of bits utilized for the array. For example, the array would contain 750,000 cells if 3,000 K bits of memory space was utilized, because 3,000Kb divided by 4 bits for each cell equals to 750,000 cells. The hash number h was set to 5 because we investigated the memory space from 1,000 to 5,000K bits with an increment of 500K, whose m/n ratio was about 0.83 (1,000Kb) to 4.17 (5,000Kb), where the

highest ratio was close to h=5. Besides, the setting the threshold rates was from 20% to 100% with an interval of 20%, so there was a distinct count for each rate using h=5. For example, if the threshold rate is 60%, at least 3 out of the 5 hashing cells of the array must be the identical attribute code or the universal one to be recognized as a member with that attribute code.

With the settings of the memory space and the threshold rates, we conducted three experiments, each of which would include serval runs according to these settings. The first one was to investigate the array composition after the programming process, where the false-positive rates of different array sizes were also presented. The array composition could include zeros, attribute codes and the universals. The results are depicted in Figure 2.

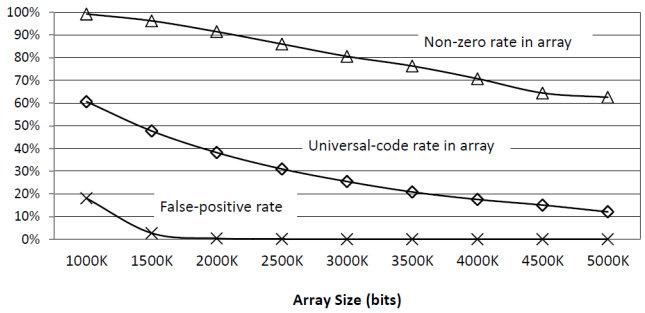


Figure 2: Array composition with varied array sizes after the programming phase

As shown in Figure 2, the non-zero rate started to stabilize when the memory space reached 4500K, while the universal-code rate continued to decrease. The situation was conceivable because the hashing process would address a similar portion of the array cells when memory space was sufficient. Each addressed cell would then be addressed by less set elements, which led to a less chance for issuing the universal code to that cell. As we can see in Figure 2, when the universal-code rate was under 40%, the false-positive rate became very close to zero. It was actually no false-positive errors at all when the rate of the universal codes was under 30% or the memory space was above 3,000K.

Since the array composition is important for an application to be successful as far as the performance is concerned, it was suggested that a healthy composition should include the portion of zeros near 50% of the array [3]. Therefore, in the second experiment, we further investigated both the non-zero and universal ratios in regard with the threshold rates. The non-zero codes certainly include the universal ones, however, we show separate results in Figures 3 and 4 in order to demonstrate the behavior of the universal-code rate because not only it is an important factor for our proposed approach to ensure the set elements to be successfully coded in the programming phase, but also represent a beacon for the FP error because the more number of universal-coded cells, the more chance of the FP errors.

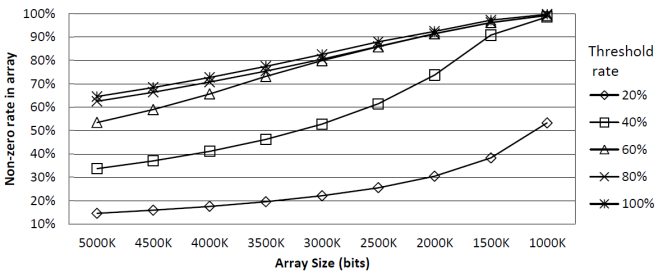


Figure 3: The array utilization rates of varied sizes with different threshold rates

As shown in Figure 3, the threshold rates above 40% (at least 2 out of 5 in the hashing addressed cells) reached a nearly full array utilization situation when the memory space is small. However, as we described that near 50% non-zero rate could be considered as a healthy composition of such a data structure and according to previous experiment that 3,000K of memory space was a suitable for a reasonably low FP error, we can see that at the threshold rate of 60% with 5,000K or 40% with 3,000K were suitable as the appropriate combinations.

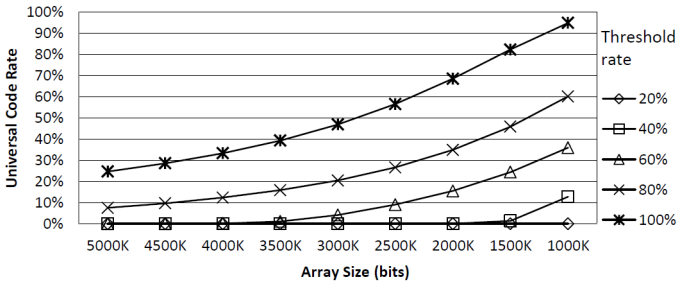


Figure 4: The universal-code rates of varied array sizes with different threshold rates

In Figure 4, it is reasonable that the universal-code rates decreased as the array size increased. For example, when the array size was 3,000Kb or above, the universal rates were all less than 50%, which were even smaller when the threshold rates decreased, *e.g.*, 20%. A small threshold rate was prone to associate with more FP errors because the low standard would easily claim membership for non-member elements. Therefore, it is important to determine suitable array size with the threshold rate for assuring the FP rate an acceptable level. According to the above-mentioned analyses, we recommend the 3,000Kb memory space for the array with the threshold rate of 40% in our specific case, whose m/n ratio is around 2.5. We also recommended the array size of 5,000K with 4.17 m/n ratio and 60% of threshold rate because it could achieve an even lower FP error. In the third experiment, nevertheless, we went through all combinations of these two factors, i.e., the array size and the threshold rate, for evaluating the FP rate as shown in Figure 5.

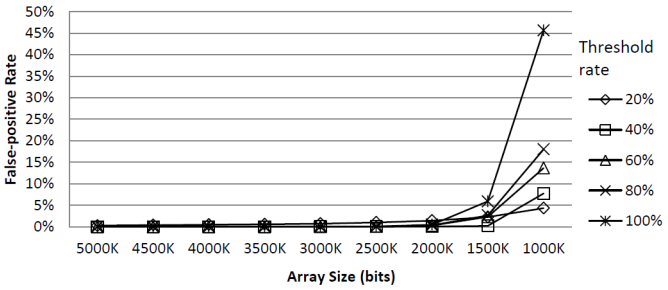


Figure 5: False-positive rates of threshold rate and array size combinations

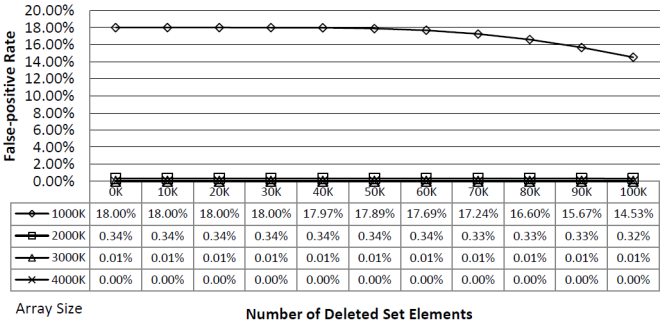


Figure 6: The false-positive rates of varied array sizes and the numbers of deletion

In Figure 5, the FP rates for all of the threshold rates were so close to 0% at the 4,000K memory space level, but the 100% and 80% of the threshold rates started to pick up as the memory space dropped to 3500K or less, which even went up to above 15% as that reduced to 1,000K. In the end, we still consider the memory space of 3,000K $\sim$ 4,000K of memory space, whose m/n ratio was 2.5 3.33 with threshold rates of 40 or 60% could be the best options for the designated case using the proposed approach.

Besides the three experiments, we conducted an extra one regarding dynamic element insertion and deletion of the dataset, which is important for certain applications requiring immediate adjustments of the set members. Therefore, the dynamic element insertion and deletion must include some mechanisms for encoding newly added or deleted members in real time, because the dataset shouldn't be reprogrammed upon slight changes of the dataset. The dynamic insertion for the proposed approach is to encode a new pair of member data (ex, ax) to the array, where ex is the new element and ax is the associated attribute code. The insertion could only be successful when the hash-addressed cells of ex in the array include ax. Once there is at least one ax, the insertion is guaranteed to be successful because the proposed approach uses the universal code U to accommodate all the attribute codes. As far as the dynamic deletion is concerned, we adopted an addition bit as the "sign" bit for each cell, whose values were initially zeros. When a member is determined to be disqualified from its membership, the sign bits of all the hash-addressed cells were turned to 1. With the proposed mechanism, some non-members would be judged as deleted members when all the sign bits of the hash-addressed are one, especially when the array size is insufficiently small, *e.g.*, 1,000Kb. We depicted this situation in Figure 6, where only the FP rate of the 1,000K memory space was shown to decrease as the number of the number of the dynamic deletion elements increased, while others remained low FP rates.

As to other members, only a fraction of cells or none whatsoever whose addressed sign bits are 1 and the attribute code is still in effect without considering the sign bit. However, if a member that was not dynamically deleted but addressed all cells with the sign bits of 1, it would be considered as a deleted member. Such a situation is referred to as a true-negative error, which is far serious than the one presented above because some members would be denied due to the sign-bit mechanism. We show the experimental results in Figure 7, where only an insufficient array size of 1,000K was significant on this situation. The true-negative errors could be a serious matter for some applications, because they represent a security hole where non-members are considered as true ones. Fortunately, the odds of such a situation are relatively slim as far as the array size is sufficient. Figure 7 shows the experimental results on this issue, where the true-negative errors only occurred when the array size was 1,000Kb. We show the details on the bottom of the figure.

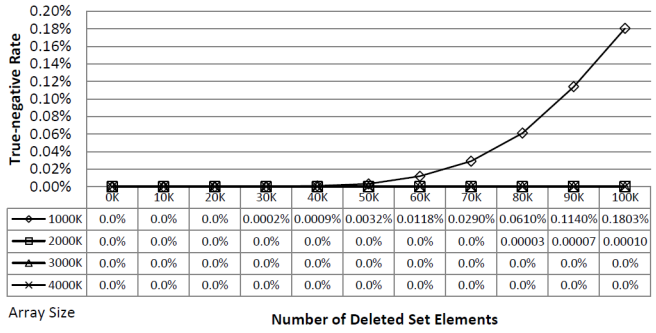


Figure 7: The true-negative rates of threshold rate and array size combinations

4 Conclusion

This study proposes a multibit representation for the original Bloom filter, whose purpose is to simultaneously provide the membership and the associated attribute information. We suggested a programming process that incorporates counting the attribute code and setting a threshold rate for the count percentage to exceed as a member. We also incorporate the universal code that accommodate all the attribute codes for the sake of expediting the programming process. Furthermore, we relieved the all-one

policy of the original Bloom filter by establishing multiple scales of threshold rate as the hurdle for determining a membership of the set elements. As the experimental results showed, we could select proper settings of these factors mentioned above after a pilot run was taken place, and we could then proceed the programming process accordingly until all the set elements are coded in the multi-bit array. Such a process also considered the FP rates as well as the array sizes with certain threshold rates at an acceptable level. The proposed approach would not elevate the computational overhead, neither the FP errors.

We also carried out an additional experiment concerning the dynamic insertion and deletion of elements of the dataset. The element insertion would be successful when the hash process addressed at least a cell whose attribute code was the same as the one of the inserting element because of the universal code; however, it would be fail when the stated-above condition does not stand. We are currently elaborating work around to establish ground work for such a matter. As far as the dynamic deletion is concerned, we strongly suggested establishing a bit array for recording the status of a deleted member, which worked well in our experiments, but required a minor addition of memory space, i.e., one bit for a cell in the array. The performance, however, was extraordinarily well because all the error rates would stay low as the memory space was relatively sufficient.

References

[1] M. H. Aghdam and P. Kabiri, "Feature selection for intrusion detection system using ant colony optimization," *International Journal of Network Security*, vol. 18, no. 3, pp. 420-432, 2016.

[2] B. H. Bloom, "Space/time trade-offs in hash coding with allowable errors," *Communications of the ACM*, vol. 13, no. 7, pp. 422–426, 1970.

[3] K. Christensen, A. Roginsky and M. Jimeno, "A new analysis of the false positive rate of a Bloom filter," *Information Processing Letters*, vol. 110, no. 21, pp. 944–949, 2010.

[4] H. Dai, Y. Zhong, A. X. Liu, W. Wang and M. Li, "Noisy Bloom Filters for multi-set membership testing," in *Proceedings of ACM Sigmetrics*, pp. 139-151, 2016.

[5] D. Ellison, "On the convergence of the multidimensional Albus perceptron," *The International Journal of Robotics Research*, vol. 10, pp. 338-357, 1991.

[6] M. Goodrich and M. Mitzenmacher, "Invertible bloom lookup tables," in *Allerton Conference on Communication, Control, and Computing*, pp. 792–799, 2011.

[7] F. Hao, M. Kodialam and T. V. Lakshman and H. Song, "Fast dynamic multiple-set membership testing using combinatorial bloom filters," *IEEE/ACM Transactions on Networking*, vol. 20, no. 1, pp. 295-304, 2012.

[8] Y. Hua, B. Xiao, B. Veeravalli and D. Feng, "Locality-sensitive bloom filter for approximate membership query," *IEEE Transactions on Computers*, vol. 61, no. 6, pp. 817-830, 2012.

[9] C. W. Liu, W. F. Hsien, C. C. Yang and M. S. Hwang, "A survey of attribute-based access control with user revocation in cloud data storage," *International Journal of Network Security*, vol. 18, no. 5, pp. 900-916, 2016.

[10] P. Liu, H. Wang, S. Gao, T. Yang, L. Zou, L. Uden and X. Li, "ID bloom filter: Achieving faster multi-set membership query in network applications," in *Proceedings of IEEE ICC*, pp. 1-6, 2018.

[11] H. Ma, Y. C. Tseng and L. I. Chen, "A CMAC-based scheme for determining membership with classification of text strings," *Neural Computing with Applications*, vol. 27, pp. 1959-1967, 2016.

[12] Q. S. Qassim, A. M. Zin, and M. J. Ab Aziz, "Anomalies classification approach for network-based intrusion detection system," *International Journal of Network Security*, vol. 18, no. 63, pp. 1159-1972, 2017.

[13] Y. Qiao, S. Chen, Z. Mo and M. Yoon, "When bloom filters are no longer compact: Multi-set membership lookup for network applications," *IEEE/ACM Transactions on Networking*, vol. 24, no. 6, pp. 3326-3339, 2016.

[14] D. Thiyagarajan and R. Ganesan, "Cryptographically imposed model for efficient multiple keyword-based search over encrypted data in cloud by secure index using bloom

filter and false random bit generator," *International Journal of Network Security*, vol. 19, no. 3, pp. 413-420, 2017.

[15] B. Xiao, Y. Hua, "Using parallel bloom filters for multiattribute representation on network services," *IEEE Transactions on Parallel and Distributed Systems*, vol. 21, no. 1, pp. 20-32, 2010.

[16] T. Yang T, A. X. Liu, M. Shahzad, D. Yang, Q. Fu, G. Xie and X. Li, "A shifting framework for set queries," *IEEE/ACM Transactions on Networking*, vol. 25, no. 5, pp. 3116-3131, 2016.

Biography

Ying-Chih Tseng is currently the deputy superintendent of Hsinchu Cathay General Hospital, Taiwan. He received his Ph.D. degree from the Ph.D. Program of Technology Management, Chung-Hua University, Taiwan. He is also an assistant professor in the Department of Information Management, Yuanpei University of Medical Technology, Taiwan. His research interests include medical image recognition and neural networks.

Heng Ma is currently a professor and the department chair of the Department of Industrial Management, Chung-Hua University, Taiwan. He received his Ph.D. degree from the Department of Industrial & Manufacturing

Engineering, The Pennsylvania State University in 1996, MS from the Department of Industrial & System Engineering, Ohio University in 1992, and BS from the Department of Industrial Engineering, National Tsing-Hua University, Taiwan, in 1988. His research interests include robotics, neural networks, image recognition, and network security.

Research on Security of Mobile Communication Information Transmission Based on Heterogeneous Network

Jing Chen, Feng Zhao, and Haiyan Xing

(Corresponding author: Jing Chen)

Department of Information Engineering and Art Design, Shandong Labor Vocational and Technical College
No. 23266, Jingshi Road, Jinan, Shandong 250022, China
(Email: jingc_cj@yeah.net)

(Received Feb. 2, 2019; revised and accepted Dec. 2, 2019)

Abstract

The popularity of the heterogeneous networks has greatly improved the performance of mobile communication technologies, but at the same time, the openness of mobile communication made information security threatened. In order to improve the security of mobile communication, this paper briefly introduced heterogeneous networks and analyzed the physical layer security performance of heterogeneous network through the security rate. Then the Monte Carlo method was used to simulate the average security rate of the two-layer heterogeneous network. The results showed that the increase of the distribution density of the micro-base station in heterogeneous network reduced the average safety rate; the increase of the transmitting power of micro-base station and the signal to interference plus noise ratio (SINR) received by legitimate users increased the average security rate until it was stable; the increase in the distribution density of eavesdropping users in heterogeneous network reduced the average security rate, but when the transmitting power of the micro-base station exceeded 0.4W and the receiving SINR of legitimate users exceeds 30 dB, the average security rate was not affected.

Keywords: Heterogeneous Network; Mobile Communication; Physical Layer Security; Security Rate

1 Introduction

China's upcoming 5G technology will further improve the performance of wireless communication system, and bring more convenience and new related industries [**?**]. At the same time, high-performance mobile communication technology not only brings convenience, but also brings demanding security performance issues [**?**]. Compared with traditional wired communication technology, the wireless communication technology has higher openness [**?**]. If the energy attenuation of wireless signal is not considered, the eavesdropper almost has the same reception condition as legitimate users, that is, eavesdroppers is completely possible to intercept signals containing information in the communication process. The traditional security assurance in mobile communication is accomplished by encrypting the transmitted information, but this method does not fundamentally solve the possibility of information interception. Physical layer security can be achieved between different base stations and users in a heterogeneous network.

The principle of physical layer security of heterogeneous networks is mainly as follows: different wireless communication channels have random characteristics, and random channels generated by different base stations in the network will interfere with other channels to some extent, so the physical layer security of mobile communication can be achieved by rational use of the above characteristics. Relevant studies include: Wang *et al.* [**?**] comprehensively studied the physical layer security of multi-layer heterogeneous cellular networks with random distribution of base stations, authorized users and eavesdroppers. The simulation results showed that the introduction of appropriate access threshold could significantly improve the security throughput performance of heterogeneous computer network (HCN). Wei *et al.* [**?**] obtained the signal-to-noise ratio (SNR) of users through the vertical height and downtilt angle of the base station antenna in three-dimensional heterogeneous network. Then, based on the distribution of the base station and users, the expressions of the cumulative SNR distribution function and the average security rate of users were obtained.

The simulation results verified that the expression was correct and the physical layer could be improved by adjusting the downtilt angle. Qi *et al.* [**?**] studied the user's connection interruption probability, confidentiality interruption and transmission interruption of users in two-tier heterogeneous cellular networks under the confidentiality

protection scheme and threshold-based scheme. The simulation results showed that the antenna system, eavesdropper density, predetermined access thresholds and detection area radius all had an impact on the security performance of heterogeneous networks. This paper briefly introduced heterogeneous networks and analyzed the physical layer security performance of heterogeneous network through the security rate. Then the Monte Carlo method was used to simulate the average security rate of the two-layer heterogeneous network.

2 Heterogeneous Network

Heterogeneous network [?] is a mobile wireless network that integrates multiple types of networks with overlapping working areas through intelligent access and provides services for users. The schematic diagram of the model is shown in Figure 1. In real life, whether it is a macro-base station or a micro-base station, its working power is limited. The power of the macro-base station [?] is relatively larger, and the effective working range that can be covered is relatively larger. However, the farther away from the base station, the weaker the signal will be. In addition, different base stations are provided with different services by different operators. If the traditional service mode is adopted, it is very likely that bad signal will occur. The micro-base station in the sub-layer of heterogeneous network can solve this problem. The micro-base station receives the signal from macro-base station and forwards it to users, which is equivalent to expanding and increasing mobile communication information. At the same time, users can still receive the signal directly from the macro-base station in this process, and they can choose to receive the stronger base station signal according to the signal strength of the macro-base station and the micro-base station.

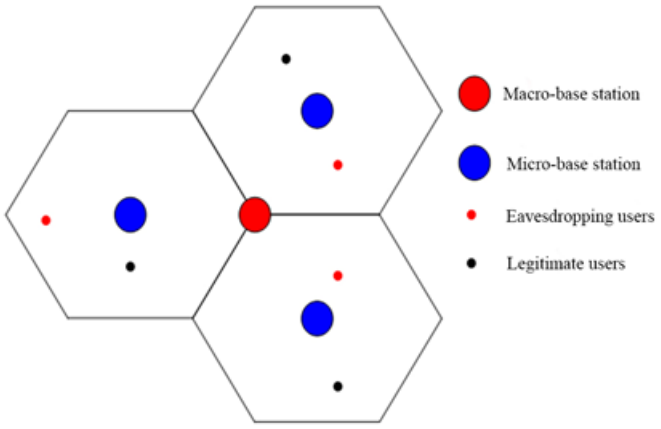


Figure 1: Schematic diagram of heterogeneous network model

According to the heterogeneous network model shown above, the expression of signals that legitimate users and eavesdropping users can receive [?] is:

$$\begin{cases} y_s &= h^H x + n_s \\ y_e &= g^H x + n_e \end{cases} \tag{1}$$

where y_s, y_e respectively stand for legitimate and eavesdropping signals received by user; h, h^H are channel vectors of legitimate users and their corresponding transpose matrices respectively; g, g^H are channel vector of eavesdropping users and their corresponding transposed matrix; x is the message signal of mobile communication; n_s, n_e are noise signals received by the legitimate and eavesdropping users respectively. Gaussian noise that obeys independent distribution is adopted in this model.

3 Physical Layer Security

The physical security model of mobile communication in a heterogeneous network [?] is shown in Figure 2. The base station first transmits a signal, then the legitimate user receives the signal transmitted by the base station through the legitimate channel, and the eavesdropping user receives the signal transmitted by the base station through the eavesdropping channel. The indexes to measure the physical security performance of heterogeneous networks include: signal to interference plus noise ratio (SINR), security rate and security interruption probability [?].

The signal to interference plus noise ratio refers to the ratio of the effective signal power and the interference signal power in the signal received by users in heterogeneous networks, and the higher the ratio is, the higher the signal quality is; at a certain transmission rate, the eavesdropping user in heterogeneous network cannot receive information through eavesdropping channel, while the legitimate user can receive information through the legitimate channel with almost no errors. Then the transmission speed is the security rate, and the maximum security rate is the security capacity of heterogeneous networks. When the security rate of legitimate users receiving information in the network is lower than the set threshold, the channel is judged to be eavesdropped and the communication is interrupted. The probability of the security interruption is the probability of occurrence of the aforementioned event.

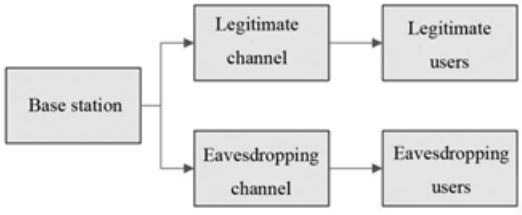


Figure 2: Physical security model of mobile communication

This paper mainly studied the mobile communication security of heterogeneous networks through the security rate. The expression of the security rate of legitimate users in heterogeneous networks [**?**] is:

$$C = (C_s - C_e)^+ = \max\{C_s - C_e, 0\}, \tag{2}$$

where C_s is the capacity of legal channel between base station and legitimate user; C_e is the capacity of eavesdropping channel between base station and eavesdropping user. The equation for calculating the average safety rate is:

$$\overline{C} = \frac{\int_0^\infty \frac{F_s(x)}{1+x}(1 - F_e(x))dx}{\ln 2} \tag{3}$$

where $F_s(x)$ and $F_e(x)$ are respectively the cumulative distribution function of SINR of legitimate and eavesdrop users. The first step is to get the SINR received by legitimate users and eavesdropping users:

$$\begin{cases} \gamma_s = \frac{P_t|h_{0,0}|^2 K r^{-(\alpha+1)}}{\sum_{i\in\phi_p\setminus\{0\}} P_t|h_{j,0}|^2 K d_i^{-(\alpha+1)} + \sum_{k\in\phi_m} P_m|g_{j,0}|^2 K l_j^{-(\alpha+1)} + \delta^2} \\ \gamma_e = \max\left\{ \frac{P_t|h_{0,e}|^2 K r_e^{-(\alpha+1)}}{\sum_{i\in\phi_p\setminus\{0\}} P_t|h_{j,e}|^2 K d_i^{-(\alpha+1)} + \sum_{k\in\phi_m} P_m|g_{j,e}|^2 K l_j^{-(\alpha+1)} + \delta^2} \right\} \end{cases} \tag{4}$$

where γ_s and γ_e are SINR received by legitimate users and eavesdropping users under the nearest base station service, respectively; $h_{j,0}$ and $h_{j,e}$ are small-scale fading coefficients between micro-base station and legitimate and eavesdropping users, respectively; $g_{j,0}$ and $g_{j,e}$ are small-scale fading coefficients between macro-base station and legitimate and eavesdropping users, respectively, both of which are Rayleigh Fading [**?**]; P_t and P_m are the transmission power of micro-base station and macro-base station respectively; K is the signal attenuation factor caused by the path; r is the distance between the user and the base station providing the service; d_i and l_j are the distance from other micro-base stations and macro-base stations to users respectively; α is the path loss index; ϕ_p and ϕ_m respectively represent that the micro-base station and macro-base station obey the Poisson distribution in heterogeneous network; δ^2 is noise power.

In heterogeneous networks, the distribution between micro-base stations and macro-base stations is independent of each other. Within the signal coverage range of the base station, the probability density function of no other base station within the distance between the legitimate user and base station is:

$$f(r) = 4\pi\lambda r^2 \exp(-\frac{4\pi\lambda r^3}{3}), \tag{5}$$

where λ is the distribution density of users; r is the distance between legitimate users and base station. Then, by combining Equation (4) and Laplace transform, the cumulative distribution function of legitimate user's SINR is deduced as follows:

$$\begin{aligned} F_s(x) = {} & 1 - \int_{r\geq 0} 4\pi r^2 \exp(-\frac{4\pi\lambda r^3}{3}) \\ & \exp(\gamma_s P_t^{-1} K^{-1} r^{\alpha+1}\delta^2) \\ & \exp(-4\pi[\lambda_p \int_r^\infty \frac{\lambda_e\gamma_s P_t^{-1}\chi^2}{\lambda_e\gamma_s P_t^{-1} + (\chi/r)^{\alpha+1}} dx \\ & + \lambda_m \int_r^\infty \frac{\lambda_e\gamma_s P_t^{-1}\chi^2}{\lambda_e\gamma_s P_t^{-1} + (\chi/r)^{\alpha+1}} dx]), \end{aligned} \tag{6}$$

where λ_p and λ_m are the distribution density of micro-base station and macro-base station, respectively. Similarly, the cumulative distribution function of SINR of eavesdropping users can be deduced, which has the same form as legitimate users. The average security rate of legitimate users can be obtained by combining Equation (3) and SINR cumulative distribution function of legitimate users and eavesdropping users:

$$\begin{aligned} \overline{C} = {} & \frac{\pi(\lambda_p + \lambda_m)}{\ln 2} \\ & \int_0^\infty \frac{\exp(-\pi\lambda_e/(\lambda_s\gamma_s P_t^{-1} r^{\alpha+1}) x^{4/(\alpha+1)})}{(1+x)(\lambda_s\gamma_s P_t^{-1} r^{\alpha+1}) x^{4/(\alpha+1)} + \pi(\lambda_p + \lambda_m)} dx \end{aligned} \tag{7}$$

where λ_s and λ_e are the distribution density of legitimate users and eavesdropping users, respectively.

From the deduced the legitimate user of average security rate, Equation (7), it can be seen that the average security rate of legitimate users is related to multiple factors in the heterogeneous network, including: the transmission power of micro-base stations, the distribution density of micro-base stations and macro-base stations, the distribution density of legitimate and eavesdropping users, the path loss index, the distance between users and base-stations, *etc.*

4 Simulation Experiment

4.1 Simulation Environment

In this paper, the Monte Carlo method [**?**] was used to conduct simulation analysis on the heterogeneous network. The simulation experiment was carried out in the laboratory server. The server configuration: Windows7 system, I7 processor, 16G memory.

4.2 Simulation Parameters

For the convenience of calculation, the simulation model established in this paper was a two-layer heterogeneous network, and the relevant initial parameters were: the effective working range of macro-base station was 5 km, and the working power was 30 W; the effective working range of the micro-base station was 100 m and the working power was 0.3 W; the antenna array of base station was

125 antennas/row and 60 antennas/column, and the frequency of working radio wave was 800 MHz; the density of legitimate user nodes was 0.01; the density of eavesdropping user nodes was 0.001; the path loss index was 4.0.

4.3 Simulation Project

1) The distribution density of the micro-base station was set between 10-4 and 10-1, and the average security rate of the legitimate users with eavesdropping node density of 0.0005, 0.001, and 0.0015 was simulated separately. Other heterogeneous network parameters were shown as the initial parameters above.

2) The working power of the micro-base station was set between 0.1 W and 0.5 W, and the average security rate of the legitimate users with eavesdropping node density of 0.0005, 0.001, and 0.0015 was simulated separately. Other heterogeneous network parameters were shown as the initial parameters above.

3) The SINR received by legitimate users is set between 20 dB and 35 dB, and the average security rate of the legitimate users with eavesdropping node density of 0.0005, 0.001, and 0.0015 was simulated separately. Other heterogeneous network parameters were shown as the initial parameters above.

4.4 Simulation Results

As shown in Figure 3, in terms of horizontal comparison, the average security rate of legitimate users decreased with the increase of the density of micro-base stations. When the density of micro-base station was between 0.0001 and 0.01, with the increase of the density of the micro-base station, the decrease amplitude of the average security rate was relatively small. When the density of the micro-base station exceeded 0.01, the descending amplitude of the average security rate of legitimate users increased. In terms of longitudinal comparison, under the same distribution density of micro-base stations, the higher the distribution density of eavesdropping users in heterogeneous networks, the lower the average security rate of legitimate users. The above simulation results showed that the increase of distribution density of micro-base stations in heterogeneous networks and the increase of eavesdropping users increased the probability of leakage in the process of information transmission, thereby reducing the average security rate of legitimate users in heterogeneous networks.

As shown in Figure 4, in terms of horizontal comparison, under the same distribution density of eavesdropping users, the average security rate of legitimate users increased with the increase of the power of the micro-base station, and the rising amplitude gradually decreased; when the power of the micro-base station increased to about 0.4 W, the average safety rate reached the maximum and remained stable. In terms of longitudinal comparison, before the power of micro-base station increased to 0.4 W and under the same power, the higher the distribution density of eavesdropping users in heterogeneous networks, the lower the average security rate of legitimate users. But after the power of micro-base station exceeded 0.4 W, regardless of the distribution density of eavesdroppers, the average security rate of legitimate users reached the same fixed value. The above simulation results showed that increasing the transmitting power of micro-base stations in heterogeneous networks could effectively improve the average security rate of legitimate users, and reduce the influence of the density of eavesdropping user distribution on the average security rate after increasing to a certain extent. At the same time, the simulation results also showed that the increase of the power of the micro-base station had a limit to the increase of the average safety rate. Considering the cost, it was not necessary to increase the power of the micro-base station as much as possible.

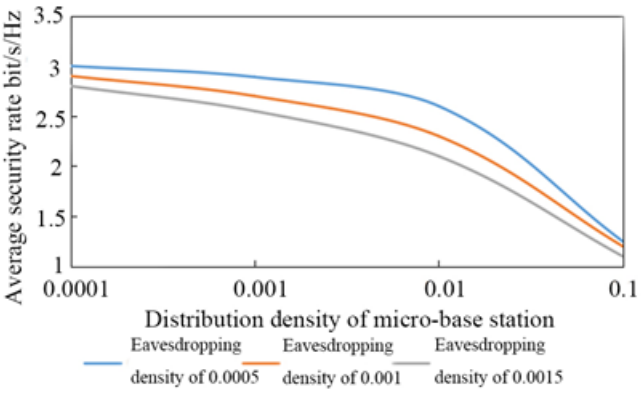


Figure 3: The effect of distribution density of micro-base station on average security rate

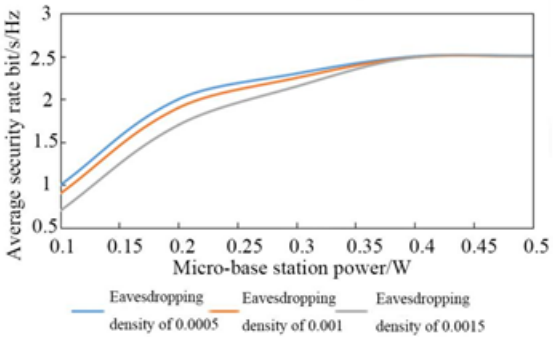


Figure 4: The effect of micro-base station transmitting power on average security rate

As shown in Figure 5, SINR received by legitimate users is adjusted by adjusting the distance between users and the base station and the path loss index α. By horizontal comparison, the average safety rate of SINR be-

tween 20 dB and 30 dB increased with the increase of SINR received by legitimate users, and the rising amplitude decreased gradually; after exceeding 30 dB, the average security rate tended to be stable. In the longitudinal comparison, when SINR was between 20 dB and 30 dB, under the same SINR, the higher the distribution density of eavesdropping users, the lower the average security rate of legitimate users. When the SINR received by legitimate users exceeded 30 dB, the average security rate of legitimate users reached the same fixed value regardless of the distribution density of eavesdropping users. The above simulation results showed that improving the SINR of the signal received by the legitimate user could effectively weaken the eavesdropping effect of the eavesdropping users and improve the average security rate.

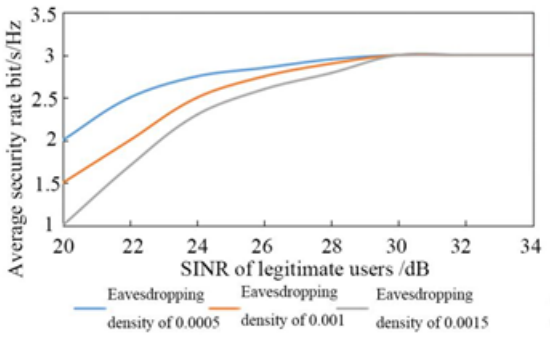


Figure 5: The effect of SINR received by legitimate users on average security rate

5 Conclusion

This paper briefly introduced heterogeneous networks and analyzed the physical layer security performance of heterogeneous network through the security rate. Then the Monte Carlo method was used to simulate the average security rate of the two-layer heterogeneous network. The results are as follows:

1) In heterogeneous network, the increase of the distribution density of micro-base stations reduced the average security rate of legitimate users; the greater the distribution density, the larger the reduction range; at the same time, the increase in the distribution density of eavesdropping users reduced the average security rate;
2) When the micro-base station transmitting power was between 0.1 W and 0.4 W, the increase of transmitting power of micro-base station increased the average safety rate; at the same time, the increase in the distribution density of eavesdropping users reduced the average security rate; after exceeding 0.4 W, the average security rate remained stable and unaffected by the distribution density of eavesdropping users;
3) When SINR received by legitimate users was between 20 dB and 30 dB, the increase of SINR improved the average security rate, and the increase of the distribution density of eavesdropping users reduced the average security rate; after exceeding 30 dB, the average security rate remained stable and was not affected by the distribution density of eavesdropping users.

Acknowledgments

This research was supported by Shandong vocational education teacher studio funding project: Lu teacher's letter [2017] No. 30.

Biography

Jing Chen now works in Shandong Vocational and Technical College of labor. She is a professor. She is interested in computer communication and mobile data transmission technology.

Feng Zhao, born in Jinan, Shandong Province, holds a master's degree. He is now working in Shandong Vocational and Technical College of labor. He is a professor. She is interested in computer network and data communication technology.

Haiyan Xing, born in Dezhou famale, from Jinan, Shandong, China, has gained the master's degree. She is now working in Shandong labor vocational and technical college. She is a lecture. she is interested in big data technology and AI technology.

Linear Complexity of Two Classes of Binary Interleaved Sequences with Low Autocorrelation

Shidong Zhang[1,2], Tongjiang Yan[1,2], Yuhua Sun[1,3], Lianhai Wang[3]
(Corresponding author: Tongjiang Yan)
College of Science, China University of Petroleum, Qingdao, Shandong 266580, China[1]
Key Laboratory of Applied Mathematics, Fujian Province University (Putian University), Fujian 350117, China[2]
Qilu University of Technology (Shandong Academy of Sciences), Shandong Computer Science Center
(National Supercomputer Center in Jinan), Shandong Provincial Key Laboratory of Computer Networks[3]
(Email: yantoji@163.com)
(Received Sep. 28, 2018; Revised and Accepted Dec. 7, 2018; First Online Jan. 11, 2019)

Abstract

The linear complexity of a key stream sequence in a stream cipher is an important cryptographic property. In this paper, we discuss the linear complexity of two classes of binary interleaved sequences of period $4N$ with low autocorrelation. Results show that the linear complexity of these two classes of sequences is large enough to resist the Berlekamp-Massey algorithm.

Keywords: Interleaved Sequence; Linear Complexity; Minimal Polynomial; Stream Cipher

1 Introduction

Sequences with good autocorrelation and large linear complexity have many applications in cryptography and communication systems [3, 10, 14].

Given two binary sequences $a = (a_t)_{t=0}^{\infty}$ and $b = (b_t)_{t=0}^{\infty}$ of period n defined on the Galois field $GF(2)$, the periodic correlation between them is defined by

$$R_{a,b}(\tau) = \sum_{t=0}^{n-1} (-1)^{a(t)+b(t+\tau)}, 0 \le \tau < n,$$

where the addition $t+\tau$ is performed modulo n. If $a = b$, $R_{a,b}(\tau)$ is called the (periodic) autocorrelation function of a, denoted by $R_a(\tau)$, otherwise, $R_{a,b}(\tau)$ is called the (periodic) cross-correlation function of a and b [13].

Binary sequences with optimal autocorrelation values can be classified into four types as follows according to the remainders of n modulo 4: (1) $R_a(\tau) = -1$ if $n \equiv 3 \bmod 4$; (2) $R_a(\tau) \in \{-2, 2\}$ if $n \equiv 2 \bmod 4$; (3) $R_a(\tau) \in \{1, -3\}$ if $n \equiv 1 \bmod 4$; (4) $R_a(\tau) \in \{0, -4\}$ or $\{0, 4\}$ if $n \equiv 0 \bmod 4$, where $0 < \tau < n$ [7]. In the first case, $R_a(\tau)$ is often called ideal autocorrelation. For the last type, if $R_a(\tau) \in \{0, \pm 4\}$, $R_a(\tau)$ is called almost optimal autocorrelation. For more details about optimal autocorrelation, the reader is referred to [1,10,12]. However, in applications, sequences with low autocorrelation values rather than optimal autocorrelation values also play important roles.

The linear complexity of a sequence is often described in terms of the shortest linear feedback shift register (LFSR) that generates the sequence. Generally speaking, for a sequence with the linear complexity is $LC(s)$, if $2LC(s)$ consecutive elements of the sequence are known, then we can find the linear recurrence relation of the sequence by solving homogeneous linear equations or B-M algorithm. Thus the whole sequence can be recovered easily [6, 15]. So the linear complexity of a key sequence must be large enough to oppugn the known-plaintext attack [2, 5].

In [9], we have proposed two new constructions of binary interleaved sequences of period $4N$ as the following:

$$a = \mathbf{I}(s^1, L^d(\overline{s^1}), s^2, L^d(\overline{s^2})). \quad (1)$$

$$a = \mathbf{I}(s^1, L^d(\overline{s^1}), \overline{s^2}, L^d(s^2)). \quad (2)$$

where s^1 is the even decimated sequence of a binary ideal autocorrelation sequence s of period N, s^2 is the odd decimated sequence of the sequence s, $\overline{s^1}$ and $\overline{s^2}$ are the complement sequences of s^1 and s^2 respectively, and d is an arbitrary integer. We have proved that both these two interleaved sequences have low autocorrelation, especially, when $d = \frac{N+1}{4}$, the sequence a in Equation (2) is a binary sequence with almost optimal autocorrelation. Ideally, a key stream sequence need to combine the low autocorrelation property with large linear complexity. So we continue to discuss the linear complexity of these two classes of sequences in this paper.

The remainder of this paper is organized as follows. Section 2 introduces some related definitions and lemmas which would be used later. In Section 3, we give both the minimal polynomials and linear complexity of these two sequences defined by Equations (1) and (2). Conclusions and remarks are given in Section 4.

2 Preliminaries

Definition 1. *[8] Let $\{a_0, a_1, \cdots, a_{T-1}\}$ be a set of T sequences of period N. An $N \times T$ matrix U is formed by placing the sequence a_i on the ith column, where $0 \le i \le T-1$. Then one can obtain an interleaved sequence u of period NT by concatenating the successive rows of the matrix U. For simplicity, the interleaved sequence u can be written as*

$$u = \mathbf{I}(a_0, a_1, \cdots, a_{T-1}),$$

where $\mathbf{I}$ denotes the interleaved operator.

Definition 2. *[8] Let $s = (s_i)_{i=0}^{\infty}$ be a sequence over a Galois field $GF(2)$. A polynomial of the form*

$$f(x) = 1 + c_1 x + c_2 x^2 + \cdots + c_r x^r \in GF[x]$$

is called the characteristic polynomial of the sequence s if

$$s_i = c_1 s_{i-1} + c_2 s_{i-2} + \cdots + c_r s_{i-r}, \forall i \ge r.$$

Among all the characteristic polynomials of s, the monic polynomial $m_s(x)$ with the lowest degree is called its minimal polynomial. The linear complexity of s is defined as the degree of $m_s(x)$, which is described as $\mathbf{LC}(s)$.

Definition 3. *[8] Let $s = (s_i)_{i=0}^{\infty}$ be a binary sequence of period N and define the sequence polynomial*

$$s(x) = s_0 + s_1 x + \cdots + s_{N-1} x^{N-1}. \tag{3}$$

Then, its minimal polynomial and linear complexity can be determined by Lemma 1.

Lemma 1. *[14] Assume s is a sequence of period N with the sequence polynomial $s(x)$ defined by Equation (3). Then the minimal polynomial is*

$$m_s(x) = \frac{x^N - 1}{\gcd(x^N - 1, s(x))};$$

the linear complexity is

$$\mathrm{LC}(s) = N - \deg(\gcd(x^N - 1, s(x))),$$

where $\gcd(x^N - 1, s(x))$ denotes the greatest common divisor of $x^N - 1$ and $s(x)$.

For the sequence polynomial, we have the following results.

Lemma 2. *[11] Let a be a binary sequence of period N, and $s_a(x)$ be its sequence polynomial. Then*

1) $s_b(x) = x^{N-\tau} s_a(x)$, *if* $b = L^{\tau}(a)$;
2) $s_b(x) = s_a(x) + \dfrac{x^N - 1}{x - 1}$, *if b is the complement sequence of a;*
3) $s_u(x) = s_a(x^4) + x s_b(x^4) + x^2 s_c(x^4) + x^3 s_d(x^4)$, *if* $u = \mathbf{I}(a, b, c, d)$.

Lemma 3. *Let N be an odd integer. The even decimated sequence and odd decimated sequence of a binary sequence of period N $s = (s_i)_{i=0}^{\infty}$ is denoted by $s^1 = (s_{2t})_{t=0}^{\infty}$ and $s^2 = (s_{2t+1})_{t=0}^{\infty}$, where $2t$ and $2t+1$ are performed modulo N. Let $s_{s^1}(x)$, $s_{s^2}(x)$ denote the sequence polynomials of s^1, s^2 respectively. Then we have*

$$s_{s^1}(x^4) + x^2 s_{s^2}(x^4) = (1 + x^{2N}) s(x^2). \tag{4}$$

Proof By Equation (3), $s_{s^1}(x)$, $s_{s^2}(x)$ can be represented as the following

$$\begin{aligned} s_{s^1}(x) &= s_0 + s_2 x + s_4 x^2 + \cdots + s_{(2(N-1))} x^{N-1} \\ &= \sum_{t=0}^{N-1} s_{2t} x^t, \\ s_{s^2}(x) &= s_1 + s_3 x + \cdots + s_{(2(N-1)+1)} x^{N-1} \\ &= \sum_{t=0}^{N-1} s_{2t+1} x^t. \end{aligned}$$

So we have

$$\begin{aligned} & s_{s^1}(x^4) + x^2 s_{s^2}(x^4) \\ = \; & s_0 + s_1 x^2 + \cdots + s_{N-2} x^{2(N-2)} \\ & + s_{N-1} x^{2(N-1)} + s_0 x^{2N} \\ & + s_1 x^{2(N+1)} + \cdots + s_{N-1} x^{2(2N-1)} \\ = \; & (1 + x^{2N}) s(x^2). \end{aligned}$$

It should be noted that we take the Legendre sequence with period of $N \equiv 3 \bmod 8$ as the base sequence of interleaved structures in Equations (1) and (2). So we have to introduce some preliminaries about Legendre sequences.

Definition 4. *[4] Let $\mathbf{Q}$ and $\mathbf{NQ}$ denote all the quadratic residues and quadratic nonresidues in Z_N respectively, where N is a prime. The Legendre sequence $l = (l_i)_{i=0}^{\infty}$ of period N is defined as*

$$l(i) = \begin{cases} 0 \text{ or } 1, & \text{if } i = 0; \\ 1, & \text{if } i \in \mathbf{Q}; \\ 0, & \text{if } i \in \mathbf{NQ}. \end{cases}$$

Specifically, l is called the first type Legendre sequence if $l(0) = 1$ otherwise the second type Legendre sequence. For simplicity, we employ l and l' to describe the first and second type of Legendre sequences respectively.

Let s be the second type Legendre sequence of period N. Then by Equation (3), we have $s(x) = \sum_{i \in \mathbf{Q}} x^i$.

Lemma 4. *[4] Let β be a primitive Nth root of unity over the field $GF(2^m)$ that is the splitting field of $x^N - 1$. Then we obtain the following basic facts:*

1) $(\mathbf{Q}, \cdot)$ *is a group with* $|\mathbf{Q}| = (N-1)/2$ *and* $q \cdot \mathbf{NQ} = \mathbf{NQ}$ *for any* $q \in \mathbf{Q}$, *where $\cdot$ denotes integer multiplication modulo N.*
2) $s(\beta^q) = s(\beta)$ *for any* $q \in \mathbf{Q}$, *and* $s(\beta^n) = 1 + s(\beta)$ *for any* $n \in \mathbf{NQ}$.

3) $s(\beta) \in \{0,1\}$ *if and only if* $2 \in \mathbf{Q}$.

4) $2 \in \mathbf{Q}$ *if and only if* $N = 8t+1$ *for some* t.

Let $q(x) = \prod_{q \in \mathbf{Q}} (x - \beta^q)$ and $n(x) = \prod_{n \in \mathbf{NQ}} (x - \beta^n)$. Then

$$x^N - 1 = (x-1)q(x)n(x).$$

3 Minimal Polynomial and Linear Complexity

3.1 The Linear Complexity of the First Class Interleaved Sequences

Theorem 1. *Let* $a = \mathbf{I}(s^1, L^d(\overline{s^1}), s^2, L^d(\overline{s^2}))$ *be a binary interleaved sequence of period* $4N$ *defined by Equation (1), where the base sequence* s *is a Legendre sequence of period* $N \equiv 3 \bmod 8$, $d \neq \frac{N+1}{4}$. *Then the minimal polynomial is* $m_a(x) = x^{2N}+1$, *and the linear complexity is* $\mathbf{LC}(a) = 2N$.

Proof By Lemmas 2 and 3, $s_a(x)$ can be written as

$$\begin{aligned}
& s_a(x) \\
= \; & s_{s^1}(x^4) + x s_{L^d(\overline{s^1})}(x^4) + x^2 s_{s^2}(x^4) + x^3 s_{L^d(\overline{s^2})}(x^4) \\
= \; & s_{s^1}(x^4) + x^{4(N-d)+1}(s_{s^1}(x^4) + \frac{x^{4N}-1}{x^4-1}) \\
& + x^2 s_{s^2}(x^4) + x^{4(N-d)+3}(s_{s^2}(x^4) + \frac{x^{4N}-1}{x^4-1}) \\
= \; & (x^{4N-4d+1}+1)s_{s^1}(x^4) + (x^{4N-4d+3}+x^2)s_{s^2}(x^4) \\
& + \frac{x^{4N-1}}{x^4-1}(x^{4N-4d+1} + x^{4N-4d+3}) \\
= \; & (x^{4N-4d+1}+1)(s_{s^1}(x^4) + x^2 s_{s^2}(x^4)) \\
& + x^{4N-4d+1}(1+x^2)\frac{x^{4N}-1}{x^4-1} \\
= \; & (x^{4N-4d+1}+1)(x^{2N}+1)s(x^2) \\
& + x^{4N-4d+1}(1+x^2)\frac{x^{4N}-1}{x^4-1}.
\end{aligned}$$

Since the finite field $GF(2^m)$ with characteristic 2 is the splitting field of $x^N - 1$, we have $x^{4N} - 1 = (x^N - 1)^4$. Then by Lemma 1

$$\begin{aligned}
& \gcd(x^{4N}-1, s_a(x)) \qquad (5) \\
= \; & (x^2-1)\gcd\Big(\frac{x^{4N}-1}{x^2-1}, x^{4N-4d+1}\frac{x^{4N}-1}{x^4-1} \\
& + (x^{4N-4d+1}+1)s(x^2)\frac{x^{2N}-1}{x^2-1}\Big) \\
= \; & (x^2-1)\gcd\Big(\frac{x^{4N}-1}{x^4-1}, \\
& (x^{4N-4d+1}+1)s(x^2)\frac{x^{2N}-1}{x^2-1}\Big) \\
= \; & (x^2-1)\frac{x^{2N}-1}{x^2-1}\gcd\Big(\frac{x^{2N}-1}{x^2-1}, \\
& (x^{4N-4d+1}+1)s(x^2)\Big).
\end{aligned}$$

Next, we analyse the above Equation (5). By Lemma 4, we have

$$\frac{x^{2N}-1}{x^2-1} = q^2(x)n^2(x) = \prod_{q \in \mathbf{Q}} (x-\beta^q)^2 \prod_{n \in \mathbf{NQ}} (x-\beta^n)^2.$$

So we only need consider whether $x - \beta^j$ is a divisor of $(x^{4N-4d+1}+1)s(x^2)$, where $1 \leq j < N$. Since the base sequence s is a Legendre sequence of period $N \equiv 3 \bmod 8$, by 2), 3) and 4) in Lemma 4, we have $s(\beta) \notin \{0,1\}$, $s(\beta^q) = s(\beta)$ for any $q \in Q$, and $s(\beta^n) = 1 + s(\beta)$ for $n \in N$. So $s(\beta^j) \neq 0$ for any $1 \leq j < N$. $s(x) \in GF(2)[x]$. Thus

$$s(x^2) = s(x)^2.$$

Then $x - \beta^j$ is not a divisor of $s(x^2)$, where $1 \leq j < N$.

Besides, since $d \neq \frac{N+1}{4}$, $4N - 4d + 1 \not\equiv 0 \bmod N$ and $1 + (\beta^j)^{4N-4d+1} \neq 0, 1 \leq j < N$. Hence $x - \beta^j$ is not the divisor of $1 + x^{4N-4d+1}$, where $1 \leq j < N$. Then

$$\gcd\Big(\frac{x^{2N}-1}{x^2-1}, (x^{4N-4d+1}+1)s(x^2)\Big) = 1.$$

According to the above discussion, it follows that $\gcd\Big(x^{4N}-1, s_a(x)\Big) = x^{2N}-1$.

Then by Lemma 1, the minimal polynomial of the sequence a defined in Theorem 1 is $m_a(x) = x^{2N}-1$, and the linear complexity is $\mathbf{LC}(a) = 2N$.

Hence, we complete the proof of Theorem 1.

3.2 The Linear Complexity of the Second Class Interleaved Sequences

Theorem 2. *Let* $a = \mathbf{I}(s^1, L^d(\overline{s^1}), \overline{s^2}, L^d(s^2))$ *be a binary interleaved sequence of period* $4N$ *defined by Equation (2), where the base sequence* s *is a Legendre sequence of period* $N \equiv 3 \bmod 8$, $d \neq \frac{N \pm 1}{4}$. *Then the minimal polynomial is* $m_a(x) = (x-1)(x^{2N}-1)$, *and the linear complexity is* $\mathbf{LC}(a) = 2N+1$.

Proof By Lemmas 1 and 2, $s_a(x)$ can be written as

$$\begin{aligned}
& s_a(x) \\
= \; & s_{s^1}(x^4) + x s_{L^d(\overline{s^1})}(x^4) + x^2 s_{\overline{s^2}}(x^4) + x^3 s_{L^d(s^2)}(x^4) \\
= \; & s_{s^1}(x^4) + xL^d(s_{s^1}(x^4) + \frac{x^{4N}-1}{x^4-1}) \\
& + x^2(s_{s^2}(x^4) + \frac{x^{4N}-1}{x^4-1}) + x^{4N-4d+3} s_{s^2}(x^4) \\
= \; & (x^{4N-4d+1}+1) s_{s^1}(x^4) + (x^{4N-4d+3}+x^2) s_{s^2}(x^4) \\
& + \frac{x^{4N}-1}{x^4-1}(x^{4N-4d+1}+x^2) \\
= \; & (x^{4N-4d+1}+1)(s_{s^1}(x^4) + x^2 s_{s^2}(x^4)) \\
& + (x^{4N-4d+1}+x^2)\frac{x^{4N}-1}{x^4-1} \\
= \; & (x^{4N-4d+1}+1)(x^{2N}+1) s(x^2) \\
& + x^2(x^{4N-4d-1}+1)\frac{x^{4N}-1}{x^4-1}.
\end{aligned}$$

Next, we consider $\gcd(x^{4N}-1, s_a(x))$. By Lemma 4, we have

$$\begin{aligned}
x^{4N}-1 &= (x-1)^4 q^4(x) n^4(x) \\
&= (x-1)^4 \prod_{q \in \mathbf{Q}} (x-\beta^q)^4 \prod_{n \in \mathbf{NQ}} (x-\beta^n)^4.
\end{aligned}$$

So we only need to consider whether $x-\beta^j, j \in Z_N$, is a divisor of $s_a(x)$. Since the base sequence s is the Legendre sequence of period $N \equiv 3 \bmod 8$, by 2), 3) and 4) in Lemma 4, we have $s(\beta^j) \neq 0$ for any $1 \leq j < N$. Then by 1) in Lemma 4, we have

$$s(1) \equiv \frac{N-1}{2} \bmod 2 = 1 \neq 0.$$

So we can obtain $s(\beta^j) \neq 0$ for any $j \in Z_N$. Additionally, since $d \neq \frac{N \pm 1}{4}$, we have $4N-4d+1 \not\equiv 0 \bmod N$ and $4N-4d-1 \not\equiv 0 \bmod N$. Thus

$$1 + (\beta^j)^{4N \pm 4d+1} \neq 0, 1 \leq j < N.$$

Then $x-\beta^j, 1 \leq j < N$, is not a divisor of $1+x^{4N-4d+1}$ and $1+x^{4N-4d-1}$. Moreover, since both $4N-4d+1$ and $4N-4d-1$ are odd, $x-1$ is the only nontrivial common divisor of $1+x^{4N-4d+1}$, $1+x^{4N-4d-1}$ and $x^{4N}-1$. Combining the above analysis, we have

$$\begin{aligned}
& \gcd(x^{4N}-1, s_a(x)) \\
= \; & (x-1)\gcd\left(\frac{x^{4N}-1}{x^4-1}, (1+x^{2N}) + \frac{x^{4N}-1}{x^4-1}\right) \\
= \; & (x-1)\frac{x^{2N}-1}{x^2-1} \\
= \; & \frac{x^{2N}-1}{x-1}.
\end{aligned}$$

Then by Lemma 1, the minimal polynomial of the sequence a is

$$m_a(x) = (x-1)(x^{2N}-1),$$

and the linear complexity is $\mathbf{LC}(a) = 2N+1$.

Hence, the proof of Theorem 2 is completed.

Example 1. *Let* $s = (0,1,0,1,1,1,0,0,0,1,0)$ *be a Legendre sequence of period* $N = 11$, $d = 1$. *Then the new binary interleaved sequence* $a = \mathbf{I}(s^1, L^d(\overline{s^1}), s^2, L^d(\overline{s^2}))$ *of period* $4N = 44$ *defined in Theorem 1 is*

$$\begin{aligned}
a \; = \; & (0,1,1,0,0,0,1,0,1,1,1,1,0,1,0,0,0,1,1,1,0,0, \\
& 0,1,1,0,0,0,1,0,1,1,1,1,0,1,0,0,0,1,1,1,0,0).
\end{aligned}$$

By Magma program, the minimal polynomial of a *is* $m_a(x) = x^{22}-1$ *and the linear complexity of* a *is* $\mathbf{LC}(a) = 22$, *which are compatible with the results given by Theorem 1.*

Example 2. *Let* $s = (0,1,0,1,1,1,0,0,0,1,0)$ *be a Legendre sequence of period* $N = 11$, $d = 2$. *Then the new binary interleaved sequence* $a = \mathbf{I}(s^1, L^d(\overline{s^1}), \overline{s^2}, L^d(s^2))$ *of period* $4N = 44$ *defined in Theorem 2 is*

$$\begin{aligned}
a \; = \; & (0,0,0,1,0,1,0,0,1,1,0,1,0,1,1,0,0,0,0,0,0,0, \\
& 1,1,1,0,1,0,1,1,0,0,1,0,1,0,0,0,1,1,1,1,1,1,1).
\end{aligned}$$

By Magma program, the minimal polynomial of a *is* $m_a(x) = (x-1)(x^{22}-1)$ *and the linear complexity of* a *is* $\mathbf{LC}(a) = 23$, *which are compatible with the results given by Theorem 2.*

4 Conclusion

In this paper, based on the discussion of roots of the sequence polynomials in the splitting field of x^N-1, we determine both minimal polynomials and linear complexity of two classes of binary interleaved sequences of period $4N$ with low autocorrelation value/magnitude constructed in [9]. Results show that when the base sequence s is a Legendre sequence of period $N \equiv 3 \bmod 8$, and $d \neq \frac{N \pm 1}{4}$, the linear complexity of these two classes of sequences is enough to resist the Berlekamp-Massey algorithm. Especially, the linear complexity of the first class sequence a is just right one half of its period, which can be applied in the construction of cyclic codes with proper dimension.

Furthermore, apart from autocorrelation property and linear complexity, the 2-adic complexity of these two classes of sequences remains to be solved.

Acknowledgments

This work was supported by Shandong Provincial Natural Science Foundation of China (No. ZR2017MA001, No. ZR2016FL01), the Open Research Fund from Shandong provincial Key Laboratory of Computer Networks, Grant No. SDKLCN-2017-03, Qingdao application research on special independent innovation plan project (No.16-5-1-5-jch), the Open Research Fund from Key Laboratory

of Applied Mathematics of Fujian Province University (Putian University) (No.SX201702, No.SX201806), and the Fundamental Research Funds for the Central Universities (No.17CX02030A).

References

[1] K. T. Arasu, C. Ding, T. Helleseth, P. V. Kumar, and H. M. Martinsen, "Almost difference sets and their sequences with optimal autocorrelation," *IEEE Transactions on Information Theory*, vol. 47, no. 7, pp. 2934–2943, 2001.

[2] Z. Chen and V. Edemskiy, "Linear Complexity of Quaternary Sequences Over Z4 Derived From Generalized Cyclotomic Classes Modulo 2p," *International Journal of Network Security*, vol. 19, no. 4, pp. 613–622, 2017.

[3] T. W. Cusick, C. Ding, and Ari Renvall, *Stream Ciphers and Number Theory*, Amsterdam: Elsevier, 2004.

[4] C. Ding, T. Helleseth, and W. Shan, "On the linear complexity of Legendre sequences," *IEEE Transactions on Information Theory*, vol. 44, no. 3, pp. 1276–1278, 1998.

[5] V. Edemskiy, C. Li, X. Zeng, and T. Helleseth, "The linear complexity of generalized cyclotomic binary sequences of period p^n," *Designs, Codes and Cryptography*, vol. 30, pp. 1–15, 2018.

[6] C. Fan, "The linear complexity of a class of binary sequences with optimal autocorrelation," *Designs, Codes and Cryptography*, vol. 86, no. 10, pp. 2441–2450, 2018.

[7] N. Li and X. Tang, "On the linear complexity of binary sequences of period 4N with optimal autocorrelation value/magnitude," *IEEE Transactions on Information Theory*, vol. 57, no. 11, pp. 7597–7604, 2011.

[8] X. Ma, T. Yan, D. Zhang, and Y. Liu, "Linear complexity of some binary interleaved sequences of period 4N," *International Journal of Network Security*, vol. 18, no. 2, pp. 244–249, 2016.

[9] R. Meng and T. Yan, "New Constructions of two binary interleaved sequences with low autocorrelation," *International Journal of Network Security*, vol. 19, no. 4, pp. 546–560, 2017.

[10] W. Su, Y. Yang, and C. Fan, "New optimal binary sequences with period 4p via interleaving Ding-Helleseth-Lam sequences," *Designs, Codes and Cryptography*, vol. 86, no. 6, pp. 1329–1338, 2018.

[11] Q. Wang and X. Du, "The linear complexity of binary sequences with optimal autocorrelation," *IEEE Transactions on Information Theory*, vol. 56, no. 12, pp. 6388–6397, 2010.

[12] T. Yan, "New binary sequences of period pq with low values of correlation and large linear complexity," *International Journal of Network Security*, vol. 10, no. 3, pp. 185–189, 2010.

[13] T. Yan, Z. Chen, and B. Li, "A general construction of binary sequences with optimal autocorrelation," *Information Sciences*, vol. 287, pp. 26–31, 2014.

[14] T. Yan, X. Du, and S. Li, "Trace representations and multi-rate constructions of two classes of generalized cyclotomic sequences," *International Journal of Network Security*, vol. 7, no. 2, pp. 269–272, 2008.

[15] J. Zhou and W. Xiong, "An algorithm for computing m-tight error linear complexity of sequences over $GF(p^m)$ with period pm," *International Journal of Network Security*, vol. 15, no. 1, pp. 59–63, 2013.

Biography

Shidong Zhang was born in 1992 in Shandong Province of China. He was graduated from Jining University . He will study for a postgraduate degree at China University of Petroleum in 2016. And his tutor is Tongjiang Yan. Email: zhangshdo1992@163.com

Tongjiang Yan was born in 1973 in Shandong Province of China. He was graduated from the Department of Mathematics, Huaibei Coal-industry Teachers College, China, in 1996. In 1999, he received the M.S. degree in mathematics from the Northeast Normal University, Lanzhou, China. In 2007, he received the Ph.D. degree in Xidian University. He is now a professor of China University of Petroleum. His research interests include cryptography and algebra. Email: yantoji@163.com

Yuhua Sun was born in 1979. She was graduated from Shandong Normal University, China, in 2001. In 2004, she received the M.S. degree in mathematics from the Tongji University, Shanghai and a Ph.D. in Cryptography from the Xidian University. She is currently a lecturer of China University of Petroleum. Her research interests include cryptography, coding and information theory. Email: sunyuha_1@163.com

Lianhai Wang was born in 1969 in Shandong Province of China. He was graduated from the Department of Mathematics, Shandong University, China, in 1992. In 2003, he received the M.S. degree in computer science from the Shandong University, China, China. In 2014, he received the Ph.D. degree in Shandong University. He is now a professor of Shandong Computer Science Center (National Supercomputer Center in Jinan). His research interests include digital forensics,network security and blockchain. Email: Wanglh@sdas.org

An Improved Image Encryption Algorithm Based on Chaotic Mapping and Discrete Wavelet Transform Domain

Lei Meng, Shoulin Yin, Chu Zhao, Hang Li, and Yang Sun
(Corresponding author: Shoulin Yin and Chu Zhao)

Software College, Shenyang Normal University
Shenyang 110034, China
(Email: ysl352720214@163.com)

(Received Aug. 28, 2018; Revised and Accepted Nov. 22, 2018; First Online Mar. 9, 2019)

Abstract

The traditional chaotic image encryption algorithm has some problems, such as low security, image scrambling and diffusion (cannot resist the Chosen-plaintext attack), low efficiency and low key sensitivity and high correlation. Therefore, an improved image encryption algorithm based on chaotic mapping and discrete wavelet transform domain is proposed in this paper. First of all, the image scrambling only changes the position of each pixel, but cannot change the pixel value. Its statistical features do not change. Therefore, image encryption algorithm widely adopts the combination of scrambling and grayscale diffusion, through multiple rounds of encryption, to enhance the image confusion and diffusion characteristics. Then, the image is processed by wavelet transform. Finally, the image is mapped by chaos. Experimental results show that the algorithm has certain advantages in statistical performance, robust performance and key sensitivity and also meets the requirements of image security and real-time performance.

Keywords: Chaotic Mapping; Discrete Wavelet Transform; Grayscale Diffusion; Image Encryption

1 Introduction

Image encryption is widely used in military reconnaissance and public security inspection. The security and efficiency of image encryption have been studied widely. In many special application scenarios such as confidential video meetings and military, the image's security requirements are very high. So it is very important for the image encryption [3, 10]. However, how to balance the scrambling performance, security performance, robustness performance and the quality maintenance of decryption image is always difficult. In recent years, many researchers had applied chaotic systems to cryptosystems which has obtained the better encryption effect. The chaos system is very sensitive to the initial conditions. As long as the initial conditions are slightly changed, the results will be greatly different, which is very suitable for plaintext scrambling. The chaotic system is characterized by randomness, sensitivity to initial values and broadband power spectral density of similar noise. The traditional encryption algorithm (such as AES, DES, *etc.*) is inefficient and difficult to meet the real-time requirements [4,9,20]. The chaotic system has the characteristics of ergodicity, pseudo-randomness and initial value sensitivity and the chaotic cipher has natural advantages in the large data volume processing. Therefore, the chaotic cipher is adopted, which becomes the hot spots for fast image encryption algorithm.

Image encryption technology currently has the following three types:

1) Based on image pixel scrambling [13, 15]. The represented approaches are Arnold transform and the magic square transform. These encryption algorithms directly act on the pixels of the image. According to some linear transformation, it changes the position of the pixel to achieve the purpose of image encryption.

2) Based on modern cryptography [18, 21]. Both commercially and militarily widely use the modern cryptography. In technically, image information as a data format is fully capable of being encrypted by modern cryptography including symmetric cryptography and asymmetric cryptography. In practical applications, symmetric cryptography is mainly used to encrypt commercial or military information, it is often used to encrypt short messages.

3) Based on chaotic technique [5, 12]. due to the development of the chaotic dynamics in recent years, people gradually realize that the chaos can be used as a new password system, which can be used to encrypt text voice and image data. Chaos is used as a new

cryptosystem which is determined by the properties of chaotic system itself.

For image encryption, there are some discoveries. Mccarthy [11] discussed that an identity-based encryption scheme enables the efficient distribution of keys in a multi-user system. Such schemes are particularly attractive in resource constrained environments where critical resources such as processing power, memory and bandwidth are severely limited. This research examines the first pragmatic lattice-based IBE scheme and brings it into the realm of practicality for use on small devices. Assad [2] proposed a new fast, simple and robust chaos-based cryptosystem structure and analyzed its performances. The cryptosystem used a diffusion layer followed by a bit-permutation layer, instead of byte-permutation, to shuffle the positions of the image pixels. Moreover, the permutation layer was achieved by a new proposed formulation of the 2D cat map that allowed an efficient implementation, measured by the time complexity, in terms of arithmetic and logic operations and also, in terms of clock cycles, of the key-dependent permutation process in comparison with the standard one. Hariyanto [7] presented arnold's cat map algorithm in digital image encryption. Su [14] proposed an image encryption scheme based on chaos system combining with DNA coding and information entropy, in which chaos system and DNA operation were used to perform substitution and entropy driven chaos system was used to perform permutation. However, two vulnerabilities were found and presented in this paper, which made the encryption fail under chosen-plaintext attack. A complete chosen-plaintext attack algorithm was given to rebuild chaos systems' outputs and recover plain image and its efficiency was demonstrated by analysis and experiments. Ye [19] proposed an efficient symmetric image encryption algorithm based on an intertwining logistic map.

So this paper propose an improved image encryption algorithm based on chaotic mapping and discrete wavelet transform domain. The rest of the paper is organized as follows. Section 2 introduces the improved elliptic curve cryptography. New medical image encryption is illustrated in Section 3. Section 4 outlines the experiments. Section 5 finally concludes the paper.

2 Chaotic Encryption Algorithm

Baker mapping [6, 8, 16] is one of chaotic mapping processing two-dimensional mapping for unit plane. When returning the spatial domain, the Baker mapping can obtain a large degree of randomness. Baker mapping is defined as follows.

$$Baker(x,y) = \begin{cases} (2x, y/2) & 0 \le x \le 0.5 \\ (2x-1, y/2+0.5) & 0.5 \le x \le 10. \end{cases}$$

Discrete Baker mapping can be denoted as $Baker(n_1, n_2, \cdots, n_k)$. Where $(n_1, n_2, \cdots, n_k)$ is integer sequence. Each integer divides N, $N_i = n_1 + \cdots + n_i$. For the pixel in position of (r, s), $N_i \le r \le N_i + n_i$, $0 \le s < N$, so the Baker mapping of this position is:

$$\begin{aligned} Baker_{(n_1,n_2,\cdots,n_k)}(r,s) = & [\frac{N}{n_i}(r - N_i) + s \bmod (\frac{N}{n_i}) \\ & + \frac{n_i}{N}(s - s \bmod (\frac{N}{n_i})) + N_i]. \end{aligned}$$

First, one square matrix $N \times N$ is divided into k vertical rectangle (height is N, width is n_i). Then each vertical rectangle is segmented as n_i sub-patches with N points. Map each sub-block to a row of pixels through the column-column.

3 Proposed Image Encryption

Digital image can be described in the spatial domain by pixel location and grey value information. The digital image encryptions are based on the combination of the two factors. Conventional image encryption algorithm is divided into pixel scrambling and gray diffusion. Image scrambling only changes the position of each pixel, cannot change the pixel values, its statistical characteristics do not change. If only using gray diffusion, the tiny change of ciphertext pixel is difficult to influence all ciphertext pixels, therefore, we combine scrambling and gray diffusion in image encryption algorithm, through several rounds of encryption, it enhances the image confusion and diffusion properties.

3.1 Image Pixel Scrambling

The scrambling of digital images is a commonly used algorithm for dealing with image security problems. Image scrambling is to change the order of pixels of the original image, so that the third party cannot distinguish image information. Using chaotic system to achieve image scrambling, the methods can be divided into two categories: a) Using chaotic transformation as a scrambling transformation matrix, this method is simple and fast, but the scrambled image has strong texture features; b) Using the chaotic system to generate sequences row-by-row of the images. The texture features are not obvious and the randomness is good. However, the multi-round iteration of chaotic systems is expensive and slow. Since Arnold is the most widely used chaotic transform, this paper uses Logistic map to generate image scrambling method by combining control parameters and Arnold mapping, which improves the scrambling algorithm.

The Arnold mapping expression is shown in following equation.

$$[x_{i+1}\ y_{i+1}]^T = [(1,q)(p,pq+1)]^T [x_i\ y_i]^T (\mathrm{mod} N).$$

Where p and q are the control parameters of the chaotic equation. (x, y) is the image pixel position; N is the side length of the image. The control parameters in the equation are generated by the Logistic map and the Logistic

map is as shown:

$$x_1(n+1) = \lambda x_1(n)[1 - x_1(n)].$$

To achieve chaotic state, let $3 \leq \lambda \leq 3.2$ (the encrypter can select floating-point type data within this range). The specific process of generating control parameter is as follows:

- Iterate the Logistic map 200 times to eliminate the impact of the initial value.
- Generate a scrambling control parameter by above equation, where $\lfloor x \rfloor$ represents the largest integer not greater than x.

$$p = \lfloor x_1(k_1) \times 2^{12} \rfloor \bmod N$$
$$q = \lfloor x_1(k_1) \times 2^{10} \rfloor \bmod N.$$

  Where $x_1(k_1)$ is the state value after the equation iterates k_1 times and N is the edge length of the image.

The position of the $(0, 0)$ pixel in the image remains unchanged regardless of the number of rounds of scrambling. In order to prevent the cracker from analyzing the ciphertext with $(0, 0)$ as the breakthrough point, it chooses to exchange $(0, 0)$ with the scrambled (m, n) point to reduce the risk of ciphertext being cracked.

3.2 Image Gray Diffusion

The using of pixel scrambling alone cannot prevent the decipher from analyzing through plaintext attacks. Decipherers often choose specific points and study their position changes during the scrambling process to find the law of transformation. The security is not high only using scrambling method. The gray-scale diffusion algorithm changes the gray value of each pixel, which avoids the above-mentioned plaintext attack and further improves the confidentiality.

The diffusion algorithm is generally performed by using a modulo operation and an addition operation. The modulo algorithm can make the calculation result within a normal value interval and the addition operation can correlate the gray values of different pixels with each other to increase the mutual influence between the pixels. On the other hand, the distribution of the gray value of each pixel is made more uniform and the texture features of the scrambled image are eliminated. In order to enhance the diffusion effect, referring to the pseudo-randomness and ergodicity of chaotic phenomena, this paper introduces the Kent chaotic factor into the algorithm. The following equation is a diffusion formula based on modulo operations, addition operations and chaotic sequences.

$$c(k) = S(k) \oplus |[P(k) + S(k)] \bmod M| \oplus C(k-1).$$

Where $P(k)$ and $C(k)$ are the current plaintext value and ciphertext value, respectively. $C(k-1)$ is the previous ciphertext value, where $C(0)$ is defined as a constant (here set to 100). $M = 256$ is the gray level. $S(k)$ is the control parameter. It is different from the control parameter generation method of pixel scrambling, $S(k)$ is obtained by Kent mapping, which is also a commonly used chaotic map. The Kent mapping definition is as shown in:

$$x_2(n+1) = \begin{cases} x_2(n)/\mu & \text{if } 0 < x \leq \mu \\ (1 - x_2(n))/(1 - \mu) & \text{if } \mu < x \leq 1. \end{cases}$$

In the above equation, taking $0 < \mu < 1$ (the encrypter can select the floating point type data within this range). The specific process of controlling parameter generation is as follows:

- Iterate the Kent mapping 200 times to eliminate the impact of the initial value.
- The scramble control parameter is generated by above equation.

$$S(k) = \lfloor x_2(k_2) 2^{16} \rfloor \bmod M.$$

Where $x_2(k_2)$ is the state value after iterating k_2 in Kent mapping.

3.3 Discrete Wavelet Transform

For a $N \times N$ image, we define the discrete wavelet transform(DCT) as:

$$C(u,v) = \frac{2}{N}\alpha(u)\alpha(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y) \cos[u\pi\frac{x}{N} + \frac{u\pi}{2N}]\cos[v\pi\frac{y}{N} + \frac{v\pi}{2N}].$$

In here, $f(x, y)$ is the pixel density of position (x, y), $C(u, v)$ denotes the DCT coefficient. In this paper, we define $\alpha(u) = \alpha(v) = 1$.

The DCT decomposes the original signal into a set of integral coefficients. The lifting method is an effective method for DCT operation and the lifting scheme generally uses roundoff function, defined as rof.

$$d_i^{l+1} = s_{2i+1}^l - rof(0.5625(s_{2i}^l + s_{2i+2}^l) - 0.0625(s_{2i-2}^l + s_{2i+4}^l)).$$
$$s_i^{l+1} = s_{2i}^l + rof(0.25(d_{i-1}^{l+1} + d_i^{l+1})).$$

Decomposed signal in $l + 1$th is s_i^l. When the time is i, the input is d_i^{l+1} along with high frequency output. First, the image is processed by DWT. After wavelet decomposition, image is divided into four blocks, that is, a low-frequency block and three high-frequency blocks, they are encrypted by four different keys to improve security than other methods.

4 Experiment Results and Analysis

In order to verify the effectiveness of proposed image encryption, we select two images (512×512 size) as input

Table 1: PSNR comparison with different methods

Image	DCC	DNAE	C-ECE	New
Lena	51.28	55.37	55.86	56.97
Pepper	51.18	52.17	53.84	57.41

image conducted on MATLAB. Figures 1 and 2 are the original images. We also make comparison with DCC [1], DNAE [22] and C-ECE (Chaotic Systems and Elliptic Curve ElGamal Scheme) [17].


Figure 1: Original images

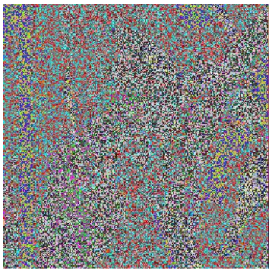
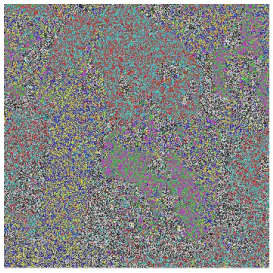



Figure 2: Encryption result with proposed method

4.1 Robustness Analysis

Since noise is inevitably introduced in the encryption process, the robustness of the algorithm in this paper is tested and PSNR value is used to judge the quality of the encrypted image as defined below:

$$PSNR = 10log\frac{WH255^2}{\sum_{i=0}^{H-1}\sum_{j=0}^{W-1}(f_1(i,j)-f_2(i,j))^2}$$

Where $f_1(i,j)$ is the pixel value of the original image pixel (i,j) and $f_2(i,j)$ represents the pixel value of the decryption terminal pixel (i,j). Obviously, the higher the PSNR value is, the better the performance of the encryption algorithm is. Table 1 is the PSNR value of Lena and Pepper. Obviously, the chaotic encryption algorithm in the transform domain has good robustness.

4.2 Differential Attack

Modifying the original plaintext image, high sensitivity is an important attribute in the image encryption algorithm. General experimental method is that it only modifies one pixel in the original image and then observe the change of image to get quantitative relationship between ciphertext image and original image, if the original image has small changes that can cause larger cipher text image change, it argues that the encryption algorithm has good robustness for differential attack.

In order to test the effect of a pixel change on the entire ciphertext image, two famous measurement methods are adopted: UACI and NPCR. Setting two encrypted images, there is only one different pixel in the two images as I_1 and I_2, the corresponding gray values are $I_1(i,j)$ and $I_2(i,j)$. Define a bipolar array B, I_1 and I_2 have the same image size. $B(i,j)$ is determined by $I_1(i,j)$ and $I_2(i,j)$. If $I_1(i,j) = I_2(i,j)$, then $B(i,j) = 1$. Otherwise, $B(i,j) = 0$. So

$$NPCR = \frac{\sum_{i,j} B(i,j)}{W \times H} \times 100\%.$$

Where W and H represent the width and height of the encrypted image and NPCR measures the ratio of the number of pixels with different pixel values between the two images to the total pixel values.

$$UACI = \frac{1}{W \times H}[\sum_{ij}\frac{I_1(i,j) - I_2(i,j)}{255}] \times 100\%.$$

UACI measures the average strength of the two images and tests the the Lena, Pepper image by modifying one pixel. The results are shown in Table 2. From the UACI and NPCR values in the table, it can be seen that the encryption algorithm in this paper has a great sensitivity to the small difference of the original image.

Table 2: NPCR, UACI comparison with different methods

Image	DCC	DNAE	C-ECE	New
Lena(NPCR)	93.2	95.4	96.7	98.1
Pepper(NPCR)	95.8	96.1	97.1	97.9
Lena(UACI)	32.5	31.6	30.9	28.7
Pepper(UACI)	31.9	30.7	30.1	29.2

4.3 Performance of Proposed Scheme

The algorithm has strong ability of resisting differential attack, encrypting 512×512 image only needs 0.041s for Lena. And we make comparison with other two methods obtaining Table 3. The comparison result shows that the method introduced in this paper guarantees the security of encryption, while the encryption speed is fast and the real-time performance is strong.

4.4 Information Entropy

Information entropy denotes the degree of uncertainty system and it is used to describe the uncertainty of image information. The information entropy can be used to

Table 3: Performance comparison with different methods

Image	DCC	DNAE	C-ECE	New
Lena	0.081	0.078	0.069	0.041
Pepper	0.084	0.067	0.062	0.052

Table 4: Information entropy comparison

Method	Lena	Pepper
DCC	0.675	0.712
DNAE	0.708	0.721
C-ECE	0.735	0.784
New	0.957	0.926

analyze the distribution of gray value in the image. Let $P(m_i)$ be proportion of pixel with gray value m_i in image and $\sum_{i=0}^{255} P(m_i) = 1$. The information entropy of the pixel is defined as:

$$H(m) = -\sum_{i=0}^{255} P(m_i \log_2 P(m_i)).$$

The comparison results are as shown in Table 4.

4.5 Key Sensitivity Analysis

Since the precision of floating point data can reach 10^{15}, the key space of the algorithm is $2 \times N^2 \times 10^{59}$, so it can be seen that the algorithm has sufficient key space. In the decryption part, we change the key, the decryption of image is fail. So the experiment shows that the algorithm has good confidentiality and the slight difference of the key will lead to the failure of image decryption as shown figure 3 in terms of Lena.

Figure 3: Left: Correct decryption result; Right: Wrong decryption result

Figure 4 is the histogram of original Lena and encrypted Lena. Figure 5 is the histogram of original Pepper and encrypted Pepper. It can be seen from the comparison of the two gray histogram images that the encrypted image pixels are uniformly distributed in the gray range of 0 255, which well covers the statistical properties of the original image and meets the expected requirements.

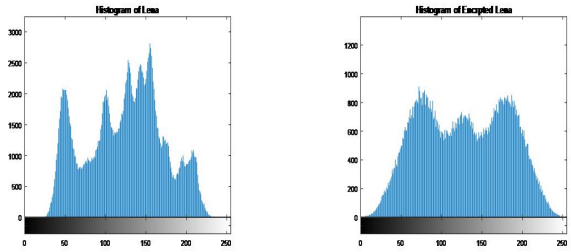


Figure 4: Left: Histogram of original Lena; Right: Histogram of encrypted Lena

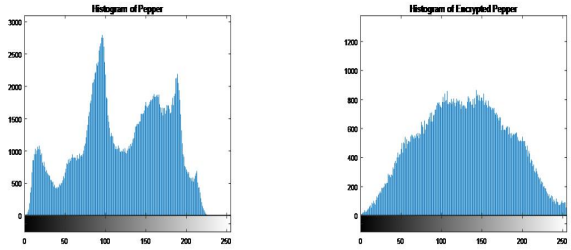


Figure 5: Left: Histogram of original Pepper; Right: Histogram of encrypted Pepper

5 Conclusion

In this paper, aiming to solve the weak performance of chaotic encryption algorithm in space domain, we propose an improved image encryption algorithm based on chaotic mapping and discrete wavelet transform domain. First, image encryption algorithm widely adopts the combination of scrambling and grayscale diffusion, through multiple rounds of encryption, to enhance the image confusion and diffusion characteristics. Then, the image is processed by wavelet transform. Finally, the image is mapped by chaos. The experimental analysis shows that the algorithm has sufficient key space and strong key sensitivity and can effectively resist the exhaustive analysis attack. After encryption, the pixel gray distribution is uniform and the correlation between adjacent pixel points is weak, which can well resist the difference attack; In the current size of the image, encryption time is short, real-time performance is strong, which can meet the need of real-time encryption and decryption.

6 Acknowledgments

This study was supported by the Natural Science Fund Project Guidance Plan in Liaoning Province of China (No. 20180520024).

References

[1] S. E. Assad, M. Farajallah, "A new chaos-based image encryption system," *Signal Processing Image Communication*, vol. 41, pp. 144-157, 2016.

[2] S. E. Assad, M. Farajallah, "A new chaos-based image encryption system," *Signal Processing Image Communication*, vol. 41, pp. 144-157, 2016.
[3] C. C. Chang, M. S. Hwang, T. S. Chen, "A new encryption algorithm for image cryptosystems", *Journal of Systems and Software*, vol. 58, no. 2, pp. 83–91, Sep. 2001.
[4] S. Dey and R. Ghosh, "A review of cryptographic properties of S-Boxes with generation and analysis of Crypto secure S-Boxes," *International Journal of Electronics and Information Engineering*, vol. 8, no. 1, pp. 49-73, 2018.
[5] S. Farwa, T. Shah, N. Muhammad, *et al.*, "An image encryption technique based on chaotic s-box and arnold transform," *International Journal of Advanced Computer Science & Applications*, vol. 8, no. 6, 2017.
[6] C. Fu, Z. K. Wen, Z. L. Zhu, *et al.*, "A security improved image encryption scheme based on chaotic Baker map and hyperchaotic Lorenz system," *International Journal of Computational Science & Engineering*, vol. 12, no. 2, 2016.
[7] E. Hariyanto, R. Rahim, "Arnold's cat map algorithm in digital image encryption," *International Journal of Science & Research*, vol. 5, no. 10, pp. 6-391, 2016.
[8] L. C. Huang, M. S. Hwang, and L. Y. Tseng, "Reversible data hiding for medical images in cloud computing environments based on chaotic Henon map," *Journal of Electronic Science and Technology*, vol. 11, no. 2, pp. 230–236, 2013.
[9] J. Liu, S. L. Yin, H. Li and L. Teng, "A density-based clustering method for k-anonymity privacy protection," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 8, no. 1, pp. 12-18, Jan. 2017.
[10] L. Liu, Z. Cao, "Analysis of two confidentiality-preserving image search schemes based on additive homomorphic encryption," *International Journal of Electronics and Information Engineering*, vol. 5, no. 1, pp. 1–5, 2016.
[11] S. Mccarthy, N. Smyth, E. O. Sullivan, "A practical implementation of identity-based encryption over NTRU lattices," in *IMA International Conference on Cryptography and Coding*, pp. 227-246, 2017.
[12] S. Rajendran, M. Doraipandian, "Chaotic map based random image steganography using LSB technique," *International Journal of Network Security*, vol. 19, no. 4, pp. 593-598, 2017.
[13] H. Sharma, N. Khatri, "An image encryption scheme using chaotic sequence for pixel scrambling and DFrFT," in *Proceedings of First International Conference on Smart System, Innovations and Computing*, pp. 487-493, 2018.
[14] X. Su, W. Li, H. Hu, "Cryptanalysis of a chaos-based image encryption scheme combining DNA coding and entropy," *Multimedia Tools & Applications*, vol. 76, no. 12, pp. 1-13, 2016.
[15] L. Teng, H. Li, S. Yin, "A multi-keyword search algorithm based on polynomial function and safety inner-product method in secure cloud environment," *International Journal of Network Security*, vol. 8, no. 2, pp. 413-422, 2017.
[16] L. Teng, H. Li, J. Liu and S. Yin, "An efficient and secure cipher-text retrieval scheme based on mixed homomorphic encryption and multi-attribute sorting method," *International Journal of Network Security*, vol. 20, no. 5, pp. 872-878, 2018.
[17] J. Wu, X. Liao, B. Yang, "Color image encryption based on chaotic systems and elliptic curve ElGamal scheme," *Signal Processing*, vol. 141, 2017.
[18] S. Yin, L. Teng, J. Liu, "Distributed searchable asymmetric encryption," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 4, no. 3, pp. 684-694, 2016.
[19] G. Ye, X. Huang, "An efficient symmetric image encryption algorithm based on an intertwining logistic map," *Neurocomputing*, vol. 251, pp. 45-53, 2017.
[20] S. L. Yin and J. Liu, "A k-means approach for map-reduce model and social network privacy protection," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 7, no. 6, pp. 1215-1221, Nov. 2016.
[21] Q. Zhang, L. T. Yang, X. Liu, Z. Chen and P. Li, "A tucker deep computation model for mobile multimedia feature learning," *ACM Transactions on Multimedia Computing, Communications and Applications*, vol. 13, no. 3, pp. 1-39:18, 2017.
[22] P. Zhen, G. Zhao, L. Min, *et al.*, "Chaos-based image encryption scheme combining DNA coding and entropy," *Multimedia Tools & Applications*, vol. 75, no. 11, pp. 6303-6319, 2016.

Biography

Lei Meng biography. He is a full associate professor of the Kexin software college at Shenyang Normal University. He has research interests in wireless networks, cloud computing and network security. Email:8871346@qq.com

Shoulin Yin biography. He received the B.Eng. and M.Eng. degree from Shenyang Normal University, Shenyang, Liaoning province, China in 2016 and 2013 respectively. Now, he is a doctor in Harbin Institute of Technology. His research interests include multimedia security, network security and image processing.

Chu Zhao biography. She received the M.Eng. degree from Shenyang Normal University, Shenyang , Liaoning province, China in 2011. Her research interests include Network Security and Data Mining. Email:910675024@qq.com.

Attribute Based Encryption with Efficient Revocation from Lattices

Kang Yang[1], Guohua Wu[2], Chengcheng Dong[1], Xingbing Fu[2,3], Fagen Li[4], Ting Wu[2]
(Corresponding author: Guohua Wu)

School of Computer Science and Technology, Hangzhou Dianzi University[1]
Hangzhou, Zhejiang Province, China
School of Cyberspace, Hangzhou Dianzi University[2]
Hangzhou, Zhejiang Province, China
Lab of Security Insurance of Cyberspace[3]
Sichuan Province
School of Computer Science and Engineering, University of Electronic Science and Technology of China[4]
Chengdu, China
(Email: drwuguohua@163.com)
(Received Sept. 7, 2018; Revised and Accepted Jan. 7, 2019; First Online Jan. 24, 2019)

Abstract

Attribute-based encryption (ABE) can be used in many cloud storage and computing applications, and it is an attractive alternative to identity-based encryption. The feature of the ABE is that it provides a flexible mechanism to achieve fine-grained access control. The revocable ABE (RABE) is an extension of the ABE. The attribute revocation is essential because of the factors such as changes of user's attributes, key exposures and key loss. In this paper, we propose a revocable ciphertext policy ABE (CP-RABE) scheme from lattices, which supports flexible access control and efficient revocation. In our scheme, a binary tree with an attribute revocation list is used to revoke attributes, and key update is logarithmically related to the number of each user attribute. Finally, the security of the scheme is proved to be selective-attribute secure in the standard model and security can be reduced to hardness of learning with error assumption.

Keywords: Attribute Based Encryption; Attribute Revocation; Binary Tree; Lattice Based Cryptography

1 Introduction

Attribute based encryption first proposed by Sahai and Waters [25] is a type of public key encryption [9, 28], and it provides a flexible mechanism with fine-grained access control. In attribute based encryption schemes [19], both the ciphertext and the key are associated with a set of attributes. According to the contents to be encrypted or the receiver's attributes, a sender can specify the access control policy such that only users whose attributes satisfy the access control policy can decrypt the encrypted ciphertext. Attribute based encryption is classified as key policy attribute based encryption (KP-ABE) [21] and ciphertext policy attribute based encryption (CP-ABE) [10,20]. In a KP-ABE scheme, the private key is associated with an access policy, and the ciphertext is associated with a set of attributes. On the contrary, in a CP-ABE scheme, the ciphertext is associated with an access policy, and the private key is associated with a set of attributes. In general, a CP-ABE scheme is more flexible than a KP-ABE scheme, since the data sender can specify the access policy when encrypting the message, instead of the key authority setting policy when user's key is extracted.

In recent years, researchers have proposed various ABE schemes [11, 27, 30]. Meanwhile, the attribute based encryption schemes from lattices [3,6,7,12,13,17] have got a great deal of attention from the cryptographic researcher. The constructions of lattice based encryption schemes are highly efficient, and its operation is fast and secure. Moreover, the lattice based encryption schemes are considered to be resistant to quantum attacks since there is no known algorithm which can break the lattice based encryption schemes.

When ABE schemes are used in practical scenarios, due to factors such as changes of user permissions and key exposures [26], it is inevitable to consider the issue of attribute revocation. The revocable attribute based encryption can be used for fine-grained access control of encrypted data in cloud computing [18] or Internet of Things. The ABE revocation scheme was first proposed in [23] where a key authority establishes an attribute revocation list and sets a valid period for user's attributes. The attribute revocation list is periodically updated ac-

cording to the expiration date, and the scheme obtains the revocation of attributes by updating the latest version of attributes. According to the scope of attribute revocation, ABE revocation schemes mainly include three types: users' revocation, revocation of users' part attributes and revocation of system attributes. With no affect to other users, the users' revocation is the revocation on all attributes contained in the attribute set of the given user, while the revocation of users' part attributes is that the user's part attributes are revoked, while the remaining attributes are not revoked. The certain user who is performed the revocation operation loses the permissions corresponding to the attributes which are revoked, but remaining users still hold the permissions of these attributes. However, when the revocation operation is about the system attributes, all users will lose the permissions of revoked attributes. According to different revocation performers, the current ABE revocation schemes are divided into two types: direct revocation and indirect revocation. The direct revocation is performed by the sender, who directly adds the revocation list of the user when encrypting message, which obtains the revocation of attributes. However, the indirect revocation is performed by the key authority, who periodically updates the unrevoked user's key. Only can the unrevoked user's key be updated, and the unrevoked user decrypt the ciphertext with the new key, while the revoked user will not be able to receive the updates, which will result in the invalidation of his key.

Inspired by the revocable identity based encryption scheme [8], this paper achieves an attribute based encryption scheme from lattices that supports user's attribute revocation. The scheme is an indirect revocation scheme and has been proved to be selective-attribute secure in the standard model.

Our contributions. We propose a revocable attribute based encryption scheme, which supports flexible threshold access control [29]. The following building blocks are used in our scheme: (1) Based on Chen et al.'s scheme [8], we propose an lattice based attribute based encryption scheme, and the scheme empolys a binary tree to support attribute revocation. It achieves flexible threshold access control and increases expressiveness of the scheme. (2) Using the Shamir secret sharing scheme [4] to recover key, our scheme chooses a random polynomial, and associates each attribute with a component of the key. (3) Our scheme proposes tuples $(key, value)$ associated with all nodes of a binary tree [5], and achieves the user's attribute revocation by updating key. The binary tree improves the efficiency of key update and makes the workload of key update logarithmically related with the maximal number of each user's attributes.

From four aspects, we have compared our scheme and other schemes in **Table 1**.

Our Techniques. In our construction, each user is associated with a binary tree and the user's attributes are associated with the leaf nodes of the binary tree, numbering all nodes of the binary tree from 1 to ξ as shown in **Figure** 1. We use tuples $(key, value)$ to store some specific information for each node of the binary tree. The key is the number of the node and $value$ is the set of attribute's leaf nodes owned by the user j when we consider the current node as the root node. When the attribute i of the user j is revoked, all nodes on the path from the root node to the leaf node are added to the revocation list RL_j. Traversing the path, adding the revoked node to the set S_1 and the unrevoked children of the revoked node to the set S_2. When obtaining the decryption key, we need to determine whether the elements' number of intersection of the set of all $value$ in the set S_2 and the set of attributes in the ciphertext policy W is equal or greater than the system threshold k. If so, the user j can obtain the decryption key. If not, return $\perp$. The time complexity of key update can be reduced to a logarithmic relation with the maximal number of each user's attributes.

Table 1: Feature comparisons

Scheme	Attribute based	Quantum security	Revocable	Standard model
Agrawal et al. [1]	no	yes	no	no
Agrawal et al. [2]	no	yes	no	yes
Zhang et al. [29]	yes	yes	no	yes
Chen et al. [8]	no	yes	yes	yes
Sahai et al. [24]	yes	no	yes	yes
Hur et al. [16]	yes	no	yes	yes
Our scheme	yes	yes	yes	yes

We define a system attribute set $\mathcal{Q} = \{1, 2, \ldots, f\}$ and a default attribute set $\mathcal{M} = \{f+1, f+2, \ldots, f+l\}$, let $\mathcal{Q}' = \mathcal{Q} \cup \mathcal{M}$. When a user with an attribute set $\mathcal{G}$ is added to the system, where $\mathcal{G} \subset \mathcal{Q}$, the scheme at random chooses a vector $\mathbf{u} = (u_1, \ldots, u_n) \in \mathbb{Z}_q^n$, and let $\mathcal{G}' = \mathcal{G} \cup \mathcal{M}$. Applying Shamir's secret sharing scheme, the vector $\mathbf{u}$ is divided into i parts, where $i \in \mathcal{G}'$, and each vector $\hat{\mathbf{u}}_i$ is at random divided into two vector $\hat{\mathbf{u}}_{i,1}, \hat{\mathbf{u}}_{i,2}$ that are associated with attributes and time, respectively. The sender sets an access structure W and a threshold k to encrypt a message, and let $W' = W \cup \{f+1, f+2, \ldots, f+l+1-k\}$. When decrypting, if there are unrevoked attributes in the user's attribute set $\mathcal{G}$ and $\mid \mathcal{G} \cap W \mid < k$, the user can't obtain the decryption key. If $\mid \mathcal{G} \cap W \mid \geq k$, the user can obtain the decryption key to decrypt the ciphertext. Then $\mid \mathcal{G}' \cap W' \mid \geq l+1$, choose a subset P such that $\mid P \mid = l+1$. Finally, we show that our scheme is secure in the standard model.

Related work. There are many attribute based encryption schemes that support attribute revocation [14–16,21, 24], most of which are indirect revocation schemes. We introduce three typical works as follows.

- Goyal et al. [14] limited the validity of the key by adding an extra expiration attribute to each user and achieved the attribute revocation by updating

the key of the expiration attribute. However, the key authority needs to regularly distribute keys to users who have not been revoked permissions. The workload of its key authority is linear in the number of users in the system, and it also requires a secure channel between the key authority and each user.

- Hur et al. [16] proposed an attribute based encryption scheme that supports immediate revocation in the context of outsourcing ciphertext. In their scheme, the sender sends the ciphertext to data outsourcing server, and data outsourcing server re-encrypts the ciphertext. Only can users whose attributes have not been revoked obtain the updated key and decrypt the new ciphertext. However, the scheme has expensive cost on key maintenances and cannot resist quantum attacks.

- Using a binary tree, Sahai et al. [24] set each user to be associated with leaf nodes such that the complexity of key update is logarithmical in the number of users in the system. Combining the nature of "ciphertext delegation", an efficient encryption scheme with attribute revocation is proposed. The key authority only needs to periodically send update key to the receiver to obtain attribute revocation, which reduces the workload of key update.

However, these schemes are all built on the traditional bilinear pairing. Bilinear pairing has its own fatal flaw, and if quantum computers are invented, cryptographic schemes based on bilinear pairing will no longer be secure. Chen et al. [8] applied the work of Sahai et al. [24] to lattices and proposed a revocable identity based encryption scheme.

2 Preliminaries

2.1 Notation

We use lowercase boldface alphabet for vectors such as $\mathbf{e}$; uppercase boldface alphabet for matrices such as $\mathbf{A}$; lowercase regular alphabet for scalars such as l. q represents a prime number, $\mathbb{R}$ represents a real number set and $\mathbb{Z}$ represents an integer set. For the positive integer f, $[f]$ denotes $(1, \ldots, f)$ and the system security parameter is n. The length of a matrix is the length of its longest vector norm: $\|\mathbf{X}\| = max\|\mathbf{x}_i\|$. ϵ: $\mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ is a negligible function if $\epsilon(\lambda)$ is smaller than all polynomial fractions for sufficiently large λ, we call an event non-negligible if its probability is $1 - \epsilon(\lambda)$.

2.2 Syntax of CP-RABE

Definition 1. *A revocable ciphertext policy lattice based attribute based encryption scheme* $\mathbf{CP-RABE}$ = { **Setup**, **PriKeyGen**, **KeyUpd**, **DecKeyGen**, **Enc**, **Dec**, **AttRev** } *consists of the following seven probabilistic polynomial time(PPT) algorithms:*

RABE.Setup$(1^\lambda, \mathcal{Q}, N) \rightarrow (pp, msk, RL_j)$. *The algorithm takes a security parameter* λ*, an attribute set* $\mathcal{Q}$ *and a maximum number of users* N *as input, and returns a public parameters* pp*, a master key* msk *and revocation lists* RL_j, $j \in N$.

RABE.PriKeyGen$(\mathrm{pp}, \mathrm{msk}, \mathcal{G}) \rightarrow \mathrm{SK}_\mathcal{G}$. *The algorithm takes the master key* msk*, the public parameters* pp *and an attribute set* $\mathcal{G} \in \mathcal{Q}$ *as input, and returns a private key* $\mathrm{SK}_\mathcal{G}$ *associated with the attribute set* $\mathcal{G}$.

RABE.KeyUpd$(\mathrm{pp}, \mathrm{msk}, t, RL_j) \rightarrow \mathrm{KU}_t$. *The algorithm takes the public parameters* pp*, the master key* msk*, an update time* $t \in \mathcal{T}$ *and revocation lists* RL_j *as input, and returns a key update* KU_t.

RABE.DecKeyGen$(\mathrm{SK}_\mathcal{G}, \mathrm{KU}_t, (W, k)) \rightarrow \mathrm{DK}_{\mathcal{G},t}$. *The algorithm takes a private key* $\mathrm{SK}_\mathcal{G}$*, a key update* KU_t*, an access structure W, and a system threshold k as input, and returns a decription key* $\mathrm{DK}_{\mathcal{G},t}$ *indicating the user has enough attributes to decrypt or a special symbol* $\perp$ *meaning that some attributes of the user are revoked.*

RABE.Enc$(\mathrm{pp}, (W, k), t, M) \rightarrow \mathrm{CT}_{W,t}$. *The algorithm takes an access structure W, a threshold k, the public parameters* pp*, a message* $M \in M_0$ *and an encryption time* $t \in \mathcal{T}$ *as input, and returns a ciphertext* $\mathrm{CT}_{W,t}$.

RABE.Dec$(\mathrm{DK}_{\mathcal{G},t}, \mathrm{CT}_{W,t}) \rightarrow M$. *The algorithm takes the decryption key* $\mathrm{DK}_{\mathcal{G},t}$ *and ciphertext* $\mathrm{CT}_{W,t}$ *as input, and returns the decryption message* M.

RABE.RevListUpd$(\mathcal{G}, t, RL_j) \rightarrow \widetilde{RL}_j$. *The algorithm takes an attribute set* $\mathcal{G}$*, a revocation time* $t \in \mathcal{T}$ *and revocation lists* RL_j *as input, and returns updated revocation lists* $\widetilde{RL}_j$.

In order to ensure the validity of the time t, the message M, and the attribute set $\mathcal{G}$, all $t \in \mathcal{T}$, $M \in M_0$, and $\mathcal{G} \in \mathcal{Q}$. The algorithms **Setup**, **PriKeyGen**, **KeyUpd**, and **RevListUpd** are run by the key authority, the algorithm **Enc** is run by the sender, and the algorithms **DecKeyGen** and **Dec** are run by the receiver.

2.3 Security Model of CP-RABE

The security model of the CP-RABE scheme under the selective-attribute and chosen plaintext attack(IND-$sAtt$-CPA) will be given below. In the model, the adversary needs to provide a challenge access structure before the system is set up. For example, the adversary chooses an access structure (W^*, k^*) before obtaining the private keys in *Phase* 1, then the attribute set $\mathcal{G}$ choosed by the adversary must satisfy $\mathcal{G} \subsetneq W^*$. The security game between an adversary $\mathcal{A}$ and a challenger $\mathcal{C}$ is as follows:

Init: The adversary $\mathcal{A}$ announces to a challenger $\mathcal{C}$ a revocation list RL_j on time period t^* and the challenge access structure (W^*, k^*).

Setup: The challenger $\mathcal{C}$ uses the **Setup** algorithm to generate the public parameters pp and the master key msk, sends the public parameters pp to the adversary $\mathcal{A}$, and holds the master key msk by himself.

Phase 1: The adversary $\mathcal{A}$ arbitrarily chooses the attribute set $\mathcal{G} = \{a_i | a_i \notin W^*\}$ and initiates a request to the challenger $\mathcal{C}$ to get the private key. The challenger $\mathcal{C}$ runs the **KeyGen** algorithm to answer the adversary's request. The adversary $\mathcal{A}$ initiates a request for updating the private key to the challenger $\mathcal{C}$ according to the revocation list RL_j, and the challenger $\mathcal{C}$ runs the **KeyUpd** algorithm to answer the adversary's request. The adversary $\mathcal{A}$ is allowed to query only during time periods are increased.

Challenge: The adversary $\mathcal{A}$ sends two message bits m_0 and m_1 to the challenger $\mathcal{C}$. The challenger $\mathcal{C}$ performs a fair coin-toss and chooses $b \in \{0,1\}$ to run the **Enc** algorithm, and sends the challenge ciphertext to the adversary $\mathcal{A}$.

Phase 2: Similar as *Phase* 1, the adversary $\mathcal{A}$ continues to make a request to the challenger $\mathcal{C}$.

Guess: The adversary $\mathcal{A}$ guesses $b' \in \{0,1\}$. If $b' = b$, then the adversary $\mathcal{A}$ succeeds in attacks.

The advantage of the adversary's success in the game is defined as

$$Adv_{\mathcal{CP}-\mathcal{RABE},\mathcal{A}}^{IND-sAtt-CPA}(\nu) = |Pr[b = b'] - \frac{1}{2}|,$$

where the probability depends on the probability distribution of random parameters and internal random coin tosses.

Definition 2. *A* CP − RABE *scheme is said to be secure against IND-sAtt-CPA secure if the advantage* $\mathrm{Adv}_{\mathcal{CP}-\mathcal{RABE},\mathcal{A}}^{\mathrm{IND-sAtt-CPA}}(\nu)$ *is a negligible function in* ν *for all polynomial time adversary* $\mathcal{A}$.

3 Background

We describe the required background knowledge as follows.

3.1 Integer Lattices

Definition 3. ([1], Definition 2). *Given any* m *linearly independent vectors* $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_m \in \mathbb{Z}^m$, *we call linear combinations of their integral coefficients as* $\mathcal{L}(\mathbf{A})$, *where* $\mathbf{A} = \{\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_m\}$, *then:*

$$\Lambda := \mathcal{L}(\mathbf{A}) := \{\mathbf{y} \in \mathbb{R}^m \ s.t. \ \exists \ \mathbf{c} = (c_1, \ldots, c_n) \in \mathbb{Z}^n, \ \mathbf{y} = \mathbf{As} = \Sigma_{i=1}^n c_i a_i\}.$$

Definition 4. *For a matrix* $\mathbf{B} \in \mathbb{Z}_q^{n \times m}$, *a vector* $\mathbf{u} \in \mathbb{Z}_q^n$, *and a prime number* q, *then:*

$$\Lambda_q^{\perp}(\mathbf{B}) = \{\mathbf{s} \in \mathbb{Z}^m \quad s.t. \quad \mathbf{Bs} = 0 \ (mod \ q)\}$$
$$\Lambda_q^{\mathbf{u}}(\mathbf{B}) = \{\mathbf{s} \in \mathbb{Z}^m \quad s.t. \quad \mathbf{Bs} = \mathbf{u} \ (mod \ q)\}.$$

3.2 Trapdoors for Lattices

Theorem 1. [22]. *Let a prime* $q \geq 2$, $m > 5n \log_2 q$, *and a positive integer* n, *there is a PPT algorithm* **TrapdoorGen**(q, n), *output a matrix* $\mathbf{B} \in \mathbb{Z}_q^{n \times m}$ *and* $T_{\mathbf{B}} \in \mathbb{Z}_q^{m \times m}$, *where* $\mathbf{B}$ *is statistically uniform on* $\mathbb{Z}_q^{n \times m}$, $T_{\mathbf{B}}$ *is the base of the lattice* $\Lambda_q^{\perp}(\mathbf{B})$ *and* $\| \widetilde{T_{\mathbf{B}}} \| \leq O(\sqrt{n \log_2 q})$.

3.3 Discrete Gaussians

Definition 5. [1]. *For any real number* $\mathbf{r} > 0$, *Gaussian function with* r *as the parameter and* $\mathbf{c}$ *as the center on* $\mathbb{R}^n$, *defined as follows:*

$$\forall x \in \mathbb{R}^n, \ \rho_{r,\mathbf{c}}(\mathbf{x}) = exp(-\pi \frac{\|x-\mathbf{c}\|^2}{r^2}).$$

When $\mathbf{c}$ *is the origin or* $r = 1$, *the subscript can be omitted.*

For any $\mathbf{c} \in \mathbb{R}^n$, *the real* $r > 0$ *and* n*-dimensional lattice* $\mathcal{L}$, *the discrete Gaussian distribution on lattices is defined as:*

$$\forall \mathbf{y} \in \mathcal{L} \ , \ \mathcal{D}_{\mathcal{L},r,\mathbf{c}}(\mathbf{x}) = \frac{\rho_{r,\mathbf{c}}(\mathbf{x})}{\rho_{r,\mathbf{c}}(\mathcal{L})}$$

For any countable set $\mathbf{B}$, $\rho_{r,\mathbf{c}}(\mathbf{B}) = \Sigma_{\mathbf{x} \in \mathbf{B}} \rho_{r,\mathbf{c}}(\mathbf{x})$.

3.4 Sampling Algorithms

SampleLeft algorithm [29]. $SampleLeft(\mathbf{A}, \mathbf{B}, \mathbf{T_A}, \mathbf{u}, s) \mapsto \mathbf{e}$. *Given a full rank matrix* $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$, *a matrix* $\mathbf{B} \in \mathbb{Z}_q^{n \times m_1}$, *a basis* $\mathbf{T_A}$ *for* $\Lambda_q^{\perp}(\mathbf{A})$, *a vector* $\mathbf{u} \in \mathbb{Z}_q^n$, *and a Gaussian parameter* $s > \|\widetilde{\mathbf{T_A}}\| \cdot \omega(\sqrt{\log(m + m_1)})$, *outputs a vector* $\mathbf{e} \in \mathbb{Z}^{m+m_1}$ *sampled from a distribution statistically close to* $D_{\Lambda_q^{\mathbf{u}}([\mathbf{A}\|\mathbf{B}]),s}$.

SampleRight algorithm [29].$SampleRight$ $(\mathbf{A}, \mathbf{B}, \mathbf{R}, \mathbf{T_G}, \mathbf{u}, s) \mapsto \mathbf{e}$. *Given a full rank matrix* $\mathbf{A}, \mathbf{B} \in \mathbb{Z}_q^{n \times m}$, *a matrix* $\mathbf{R} \in \mathbb{Z}^{m \times m}$, *a vector* $\mathbf{u} \in \mathbb{Z}_q^n$, *a basis* $\mathbf{T_B}$ *for* $\Lambda_q^{\perp}(\mathbf{B})$, *and a Gaussian parameter* $s > \|\widetilde{\mathbf{T_B}}\| \cdot \|\mathbf{R}\| \cdot \omega(\sqrt{\log m})$ *outputs a vector* $\mathbf{e} \in \mathbb{Z}^{2m}$ *sampled from a distribution statistically close to* $D_{\Lambda_q^{\mathbf{u}}([\mathbf{A}\|\mathbf{AR}+\mathbf{B}]),\sigma}$.

3.5 The LWE Hardness Assumption

Definition 6. [1]. *Decisional Learning With Errors (DLWE). Let* q *be a prime number and* n *be a positive integer. For any* $a > 0$, *define 0 is the center of* Ψ_a, *and the normal distribution on* $[0, 1)$ *with variance* $a/\sqrt{2\pi}$, *the discrete distribution on the corresponding* $\mathbb{Z}_q$ *is* $\overline{\Psi}_a$. *Suppose the learning with error* χ *on* $\mathbb{Z}_q$, *define the distribution* $\mathbf{A}_{s,\chi}$ *on* $(\mathbf{u}_i, v_i) = (\mathbf{u}_i, \mathbf{u}_i^T s + x_i) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$, *where* $\mathbf{u}_i \in \mathbb{Z}_q$ *is a randomly selected vector,* $x_i \in \mathbb{Z}_q$ *is independently selected according to the distribution* χ. *The decision* $(\mathbb{Z}_q, n, \chi) - LWE$ *is to distinguish between the pseudo-random distribution and the true random distribution on* $\mathbf{A}_{s,\chi}$ *and* $\mathbb{Z}_q^n \times \mathbb{Z}_q$.

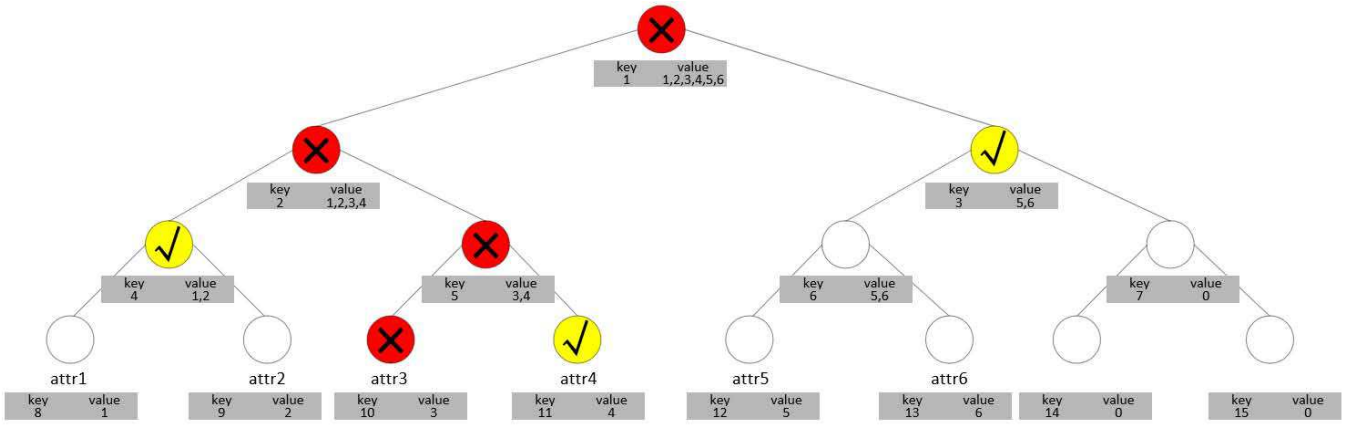


Figure 1: Description of the KUNodes algorithm on binary tree

3.6 Encoding Attributes and Time as Matrices

Definition 7. **[8].** *Let q be a prime number, m be a positive integer. A full rank difference (FRD) map function $\mathbf{H}: \mathbb{Z}_q^m \longrightarrow \mathbb{Z}_q^{m\times m}$, It has two properties. One is that for all different $i, t \in \mathbb{Z}_q^m$, the matrix $\mathbf{H}(i) - \mathbf{H}(t) \in \mathbb{Z}_q^{m\times m}$ is full rank; the other is that H is computable in the polynomial time of $O(m \log q)$.*

3.7 The Binary Tree Data Structure

Our construction takes advantage of binary tree to support attribute revocation, as with [5, 8]. In our scheme, the user j is associated with a binary tree BT_j. Each attribute i of the user j is associated with a leaf node, the path $Path(i)$ denotes a set of all nodes from the leaf node i to the root node. All nodes are associated with tuples $(key, value)$, the key is set to the number of the node, and the $value$ is the set of attribute's leaf nodes owned by the user j when we consider the current node as the root node. The $value$ is 0 when the current node doesn't have any attribute leaf nodes. If ξ is an intermediate node, ξ_l and ξ_r represent the left and right child node of the node ξ, respectively. t_i is the revocation time of the attribute i. S_1 is the set of all nodes in $Path(i)$ whose attribute i was revoked after time t, and S_2 is the set of non-revoked child nodes whose attribute i was revoked after time t.

For leaf node $(i, t_i) \in RL_j$ of the user j, if all nodes $\xi \in RL_j$ in $Path(i)$, then add $Path(i)$ to the set S_1. For all nodes $\xi \in S_1$, if $\xi_l \notin S_{1,j}$, then add $value_{\xi_l}$ to the set S_2. If $\xi_r \notin S_{1,j}$, then add $value_{\xi_r}$ to the set S_2. If the set S_2 is empty, then add the root node to the set S_2. By running the **KUNodes** algorithm, all parent nodes of the revoked node are revoked. The algorithm outputs all non-revoked child nodes of the revoked node, indicating that the user's attributes were not revoked at the time t. The **KUNodes** algorithm that obtains the attribute revocation is as follows:

$\mathbf{KUNodes}(BT_j, RL_j, t)$

$S_1, S_2 \leftarrow \emptyset$
$\forall (i, t_i) \in RL_j$
  if $t_i \leq t$ then add $Path(i)$ to S_1
$\forall \xi \in S_1$
  if $\xi_l \notin S_1$ then add $value_{\xi_l}$ to S_2
  if $\xi_r \notin S_1$ then add $value_{\xi_r}$ to S_2
if $\mid S_2 \mid = 0$ then add root to S_2
Return S_2

We give an example to illustrate our attribute revocation method as follows.

As shown in **Figure** 1, the nodes 2 to 15 are associated with a tuple $(key, value)$, respectively. The $value$ of the node 2 is the set of the attributes $attr1, attr2, attr3$ and $attr4$ because these attribute nodes are the leaf nodes of the node 2, the $value$ of other nodes is calculated in the same way. Assume the user 1 owns the attributes $attr1, attr2, attr3, attr4, attr5, attr6$. When $attr3$ is revoked, the nodes 1, 2, 5 and 10 are added to the set X, and the $value = 1, 2$, $value = 4$, $value = 5, 6$ are added to the set Y, so $\mathbf{KUNodes}(BT_j, RL_j, t) \rightarrow Y = \{1, 2, 4, 5, 6\}$. Assume the access structure $W = \{1, 2, 3\}$ and the system threshold $k = 3$, then

$$IN = W \cap \mathbf{KUNodes}(BT_j, RL_j, t) = \{1,2\}$$

Because of $\mid IN \mid < k$, it means that the user 1 has not decryption permissions.

4 A New CP-RABE Scheme from Lattices

In this section, we propose a revocable ciphertext policy lattice based attribute based encryption scheme. Unlike previous bilinear pairing based cryptographic schemes, the scheme is built on the mathematical structure of lattices. For convenience, it is assumed that there are f attributes in the system, and $\mathcal{Q} = \{1, 2, ..., f\}$, representing a set of all attributes. The ciphertext CT is associated with an access policy (W, k), where $W \subset \mathcal{Q}$ is an

attribute set, an integer k represents a threshold which is up bounded by a system parameter l, and $D = ((f+l)!)^2$. The access policy (W, k) indicates that the scheme can be successfully decrypted when the set of attributes associated with the key intersects W by more than or equal to k.

RABE.Setup$(1^\lambda, \mathcal{Q}, N)$. On input a security parameter λ, a system attribute set $\mathcal{Q} = \{1, 2, ..., f\}$ and a maximum number of users N in the system, as described in the construction framework of Section 2.2. The system sets the parameters $q, m, n, l, \sigma, \alpha$. Do:

1) Select a default set of attributes $\mathcal{M} = \{f+1, f+2, ..., f+l\}$, let $\mathcal{Q}^{'} = \mathcal{Q} \cup \mathcal{M}$.

2) Call the **TrapdoorGen(q,n)** algorithm to generate a uniformly random matrix $\mathbf{A}_0 \in \mathbb{Z}_q^{n\times m}$ with a basis $\mathbf{T}_{\mathbf{A}_0} \in \mathbb{Z}_q^{m\times m}$ for a lattice $\mathbf{\Lambda}_q^{\perp}(\mathbf{A}_0)$ such that $\| \widetilde{\mathbf{T}_{\mathbf{A}_0}} \| \leq \mathcal{O}(n\sqrt{\log q})$.

3) Select uniformly random matrices $\mathbf{B_1}, \mathbf{B_2}, \mathbf{C_1}, \mathbf{C_2} \in \mathbb{Z}_q^{n\times m}$ and a vector $\mathbf{u} = (u_1, \ldots, u_n) \in \mathbb{Z}_q^n$. For each $i \in \mathcal{Q}^{'}$, $\mathbf{a}_i \in \mathbb{Z}_q^n$ is randomly selected. Select a FRD map H.

4) Let RL_j be initially an empty list, where $j \in N$, and BT_j be a binary tree.

5) Output revocation lists RL_j, the public parameters pp, and the master key msk,

$$\mathrm{pp} = (\mathbf{A_0}, \mathbf{B_1}, \mathbf{B_2}, \mathbf{C_1}, \mathbf{C_2}, \{\mathbf{a}_i\}_{i\in\mathcal{R}'}, \mathbf{u}, \mathrm{H}),$$
$$\mathrm{msk} = \{\mathbf{T}_{\mathbf{A}_0}\}.$$

RABE.PriKeyGen$(\mathrm{pp}, \mathrm{msk}, \mathcal{G})$. On input the public parameters pp, the master key msk and a user attribute set $\mathcal{G} \subset \mathcal{Q}$, and let $\mathcal{G}^{'} = \mathcal{G} \cup \mathcal{M}$. Do:

1) For $i = 1, 2, \ldots, n$, randomly choose degree d polynomial $p_i(x) \in \mathbb{Z}_q[x]$ such that $p_i(0) = u_i$. For each attribute in $i \in \mathcal{G}^{'}$, let $\hat{\mathbf{u}}_i = (p_1(i), \ldots, p_n(i))^T \in \mathbb{Z}_q^n$. Note that, for any subset $P \subseteq \mathcal{G}^{'}$ *with* $|P| = l+1$, we have $\mathbf{u} = \Sigma_{i\in P} L_i \cdot \hat{\mathbf{u}}_i$, where the Lagrangian coefficient $L_i = \frac{\Pi_{j\in J, j\neq i} -j}{\Pi_{j\in J, j\neq i} i-j}$.

2) All nodes of the binary tree are sequentially numbered from the root node, the number of the root node is 1 and the number of the ξ-th node is ξ. For any node ξ in $Path(i)$ of the current user, $\hat{\mathbf{u}}_{i,\xi,1} \in \mathbb{Z}_q^n$ are randomly choosed, and let $\hat{\mathbf{u}}_{i,\xi,2} = \hat{\mathbf{u}}_i - \hat{\mathbf{u}}_{i,\xi,1}$, it is stored in the node ξ.

3) Calculate **SampleLeft**$(\mathbf{A}_0, \mathrm{H}(\mathbf{a}_i)\ \mathbf{C_1} + \mathbf{B}_1, \mathbf{T}_{\mathbf{A}_0}, \sigma, \hat{\mathbf{u}}_{i,\xi,1}) \rightarrow \mathbf{e}_{i,\xi,1}$, output node private key $\mathrm{SK}_{\mathcal{G}}$,

$$\mathrm{SK}_{\mathcal{G}} = (\xi, \mathbf{e}_{i,\xi,1}, value_\xi)_{\xi\in Path(i)}$$

RABE.KeyUpd$(\mathrm{pp}, \mathrm{msk}, t, RL_j)$. On input the public parameters pp, the master key msk, a key update time t and revocation lists RL_j, we think t as a vector $\mathbf{t} \in \mathbb{Z}_q^n$. Do:

1) For any node $\xi \in$ **KUNodes** (BT_j, RL_j, t), if $\hat{\mathbf{u}}_{i,\xi,1}$,$\hat{\mathbf{u}}_{i,\xi,2}$ are not defined, then generate $\hat{\mathbf{u}}_{i,\xi,1}$, $\hat{\mathbf{u}}_{i,\xi,2}$ according to the **PriKeyGen**$(\mathrm{pk}, \mathrm{msk}, \mathcal{G})$ algorithm.

2) Calculate **SampleLeft**$(\mathbf{A}_0, \mathrm{H}(\mathbf{t})\mathbf{C_2} + \mathbf{B}_2, \mathbf{T}_{\mathbf{A}_0}, \sigma, \hat{\mathbf{u}}_{i,\xi,2}) \rightarrow \mathbf{e}_{i,\xi,2}$, where H is a function that maps time t to an n×m matrix. Output an updated key:

$$\mathrm{KU}_t = (\xi, \mathbf{e}_{i,\xi,2}, value_\xi)_{\xi\in\mathbf{KUNodes}(BT, RL_j, t)}.$$

RABE.DecKeyGen$(\mathrm{SK}_{\mathcal{G}}, \mathrm{KU_t}, (W, k))$. On input two sets $\mathrm{SK}_{\mathcal{G}} = \{(x, \mathbf{e}_{i,x,1}, value_x)\}_{x\in S_1}$ and $\mathrm{KU_t} = \{(y, \mathbf{e}_{i,y,2}, value_y)\}_{y\in S_2}$, where S_1 represents nodes contained in path $Path(i)$ of S_1, $i \in \mathcal{G}$, $j \in N$, and S_2 represents unrevoked children of revoked nodes. Do:

1) If the elements' number of intersection of the set of all *value* in the set S_2 and the set of attributes in the ciphertext policy W is equal or greater than the system threshold k, it means that the user has decryption permissions, then let $\mathrm{DK}_{\mathcal{G},\mathrm{t}} = (\mathbf{e}_{i,x_j,1}, \mathbf{e}_{i,y_j,2})$. If the intersection is less than the system threshold k, it means that the user has not decryption permissions, then output $\mathrm{DK}_{\mathcal{G},\mathrm{t}} = \perp$.

2) For the user $j \in N$ with decryption permissions and the attribute $i \in \mathcal{G}$, there are some of the same nodes between each $Path(i)$ of S_1 and the set S_2, i.e., for the user j, the algorithm finds components of $\mathrm{SK}_{\mathcal{G}}$ and $\mathrm{KU_t}$ such that $\mathbf{F}_i\mathbf{e}_{i,1} + \mathbf{F}_t\mathbf{e}_{i,2} = \hat{\mathbf{u}}_i$ since they are in the same node(*The matrix* $\mathbf{F}_i$ *and* $\mathbf{F}_t$ *will be introduced in the encryption phase*). Because of $x_j = y_j$ in the previous step, we can omit x_j, y_j, then $\mathrm{DK}_{\mathcal{G},\mathrm{t}} = (\mathbf{e}_{i,1}, \mathbf{e}_{i,2})$.

RABE.Enc$(\mathrm{pp}, (W, k), t, M)$. On input the public key $\mathrm{pp} = (\mathbf{A}_0, \mathbf{B}_1, \mathbf{B}_2, \mathbf{C_1}, \mathbf{C_2}, \{\mathbf{a}_i\}_{i\in\mathcal{R}'}, \mathbf{u}, \mathrm{H})$, an access structure W, a threshold k, satisfy $1 \leq k \leq min(|\ W\ |, l)$, a message bit m and a time $\mathbf{t} \in \mathbb{Z}_q^n$. Let $W^{'} = W \cup \{f+1, f+2, \ldots, f+l+1-k\}$ and $D = ((f+l)!)^2$. Do:

1) Construct

$$\mathbf{F}_i = (\mathbf{A}_0 \mid \mathrm{H}(\mathbf{a}_i)\mathbf{C_1} + \mathbf{B}_1) \in \mathbb{Z}_q^{n\times 2m},$$
$$\mathbf{F}_t = (\mathbf{A}_0 \mid \mathrm{H}(\mathbf{t})\mathbf{C_2} + \mathbf{B}_2) \in \mathbb{Z}_q^{n\times 2m}$$
$$\mathbf{F}_{i,t} = (\mathbf{A}_0 \mid \mathrm{H}(\mathbf{a}_i)\mathbf{C_1} + \mathbf{B}_1 \mid \mathrm{H}(\mathbf{t})\mathbf{C_2} + \mathbf{B}_2) \in \mathbb{Z}_q^{n\times 3m}$$

2) Choose a uniformly random $\mathbf{s} \in \mathbb{Z}_q^n$.

3) Choose a noise $x \xleftarrow{\overline{\Psi}_\alpha} \mathbb{Z}_q$, a noise vectors $\mathbf{y} \xleftarrow{\overline{\Psi}_\alpha^m} \mathbb{Z}_q^m$.

4) For each attribute $i \in W^{'}$, randomly choose two matrices $\mathbf{R}_{i,1}, \mathbf{R}_{i,2} \in \{-1, 1\}^{m\times m}$, calculate $\mathbf{r}_{i,1} \leftarrow \mathbf{R}_{i,1}^T\mathbf{y} \in \mathbb{Z}_q^m, \mathbf{r}_{i,2} \leftarrow \mathbf{R}_{i,2}^T\mathbf{y} \in \mathbb{Z}_q^m$.

5) Output ciphertext $\mathrm{CT}_{W,t} = (c_0, \mathbf{c}_i) \in \mathbb{Z}_q \times \mathbb{Z}_q^{3m}$, where

$$c_0 \leftarrow \mathbf{u}^T\mathbf{s} + Dx + M\lfloor\tfrac{q}{2}\rfloor \in \mathbb{Z}_q,$$
$$\mathbf{c}_i \leftarrow \mathbf{F}_{i,t}^T\mathbf{s} + D(\mathbf{y}, \mathbf{r}_{i,1}, \mathbf{r}_{i,2})^T \in \mathbb{Z}_q^{3m}$$

RABE.Dec($\mathrm{DK}_{\mathcal{G},t}, \mathrm{CT}_{W,t}$). On input a decryption key $\mathrm{DK}_{\mathcal{G},t}$ and a ciphertext $\mathrm{CT}_{W,t}$. The user's attribute set $\mathcal{G}$ is associated with $\mathrm{DK}_{\mathcal{G},t}$, and the access structure W is associated with $\mathrm{CT}_{W,t}$. If $\mid \mathcal{G} \cap W \mid < k$, then return $\perp$; otherwise, let $\mathcal{G}^{'} = \mathcal{G} \cup \mathcal{M}$, $W^{'} = W \cup \{f+1, f+2, \ldots, f+l+1-k\}$. Since $\mid \mathcal{G} \cap W \mid \geq k$, there is $\mid \mathcal{G}^{'} \cap W^{'} \mid \geq l+1$. Choose a subset P of $\mid \mathcal{G}^{'} \cap W^{'} \mid$ such that $\mid P \mid = l+1$. Do:

1) Parse c_i as

$$\begin{bmatrix} \mathbf{c}_{i,0} \\ \mathbf{c}_{i,1} \\ \mathbf{c}_{i,2} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_0^T \mathbf{s} \\ (\mathrm{H}(\mathbf{a}_i)\mathbf{C_1} + \mathbf{B}_1)^T \mathbf{s} \\ (\mathrm{H}(\mathbf{t})\mathbf{C_2} + \mathbf{B}_2)^T \mathbf{s} \end{bmatrix} + D \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,1} \\ \mathbf{r}_{i,2} \end{bmatrix},$$

where $\mathbf{c}_{i,0}, \mathbf{c}_{i,1}, \mathbf{c}_{i,2} \in \mathbb{Z}_q^m$.

2) Compute

$$\begin{aligned} \mathbf{c}_i^{'} &= \mathbf{e}_{i,1}^T \begin{bmatrix} \mathbf{c}_{i,0} \\ \mathbf{c}_{i,1} \end{bmatrix} + \mathbf{e}_{i,2}^T \begin{bmatrix} \mathbf{c}_{i,0} \\ \mathbf{c}_{i,2} \end{bmatrix} \\ &= \hat{\mathbf{u}}_i^T \mathbf{s} + D(\mathbf{e}_{i,1}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,1} \end{bmatrix} + \mathbf{e}_{i,2}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,2} \end{bmatrix}) \in \mathbb{Z}_q^n. \end{aligned}$$

3) According to the Lagrangian interpolation formula in Shamir's secret-sharing scheme $\mathbf{u} = \Sigma_{i \in P} L_i \cdot \hat{\mathbf{u}}_i$ to recover

$$\mathbf{c}^{''} = \mathbf{u}^T \mathbf{s} + \sum_{i \in P} DL_i (\mathbf{e}_{i,1}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,1} \end{bmatrix} + \mathbf{e}_{i,2}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,2} \end{bmatrix}),$$

where the Lagrangian coefficient $L_i = \frac{\Pi_{j \in J, j \neq i} -j}{\Pi_{j \in J, j \neq i} i-j}$.

4) Compute $\mathbf{c}^{''}$, if $\mid c_0 - \mathbf{c}^{''} - \lfloor \frac{q}{2} \rfloor \mid < \frac{q}{4}$, return $M = 1$, otherwise return 0.

RABE.RevListUpd($\mathcal{G}, t, RL_j$).On input an attribute set $\mathcal{G}$, a time t and a revocation list $RL_j, j \in N$, the algorithm adds attribute set $\mathcal{G}$ and time t of all nodes associated with attribute i to the revocation list RL_j, and returns the revocation list $\widetilde{RL}_j$.

4.1 Correctness and Parameters

When the user's attributes satisfy the threshold access control policy W, that is, $\mid \mathcal{G} \cap W \mid \geq k$, we have $\mid \mathcal{G}^{'} \cap W^{'} \mid \geq l+1$. Choose a set of attributes with $l+1$ legal attributes. For each attribute i, we have:

$$\begin{aligned} \mathbf{c}_i^{'} &= \mathbf{e}_{i,1}^T \begin{bmatrix} \mathbf{c}_{i,0} \\ \mathbf{c}_{i,1} \end{bmatrix} + \mathbf{e}_{i,2}^T \begin{bmatrix} \mathbf{c}_{i,0} \\ \mathbf{c}_{i,2} \end{bmatrix} \\ &= \mathbf{e}_{i,1}^T \begin{bmatrix} \mathbf{A}_0^T \mathbf{s} + D\mathbf{y} \\ (\mathrm{H}(\mathbf{a}_i)\mathbf{C_1} + \mathbf{B}_1)^T \mathbf{s} + D\mathbf{r}_{i,1} \end{bmatrix} \\ &\quad + \mathbf{e}_{i,2}^T \begin{bmatrix} \mathbf{A}_0^T \mathbf{s} + D\mathbf{y} \\ (\mathrm{H}(\mathbf{t})\mathbf{C_2} + \mathbf{B}_2)^T \mathbf{s} + D\mathbf{r}_{i,2} \end{bmatrix} \\ &= \mathbf{e}_{i,1}^T \begin{bmatrix} \mathbf{A}_0^T \\ (\mathrm{H}(\mathbf{a}_i)\mathbf{C_1} + \mathbf{B}_1)^T \end{bmatrix} \mathbf{s} \\ &\quad + \mathbf{e}_{i,2}^T \begin{bmatrix} \mathbf{A}_0^T \\ (\mathrm{H}(\mathbf{t})\mathbf{C_2} + \mathbf{B}_2)^T \end{bmatrix} \mathbf{s} \\ &\quad + D(\mathbf{e}_{i,1}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,1} \end{bmatrix} + \mathbf{e}_{i,2}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,2} \end{bmatrix}) \\ &= \mathbf{e}_{i,1}^T \mathbf{F}_i^T \mathbf{s} + \mathbf{e}_{i,2}^T \mathbf{F}_t^T \mathbf{s} + D(\mathbf{e}_{i,1}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,1} \end{bmatrix} + \mathbf{e}_{i,2}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,2} \end{bmatrix}) \\ &= (\mathbf{F}_i e_{i,1} + \mathbf{F}_t e_{i,2})^T \mathbf{s} + D(\mathbf{e}_{i,1}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,1} \end{bmatrix} + \mathbf{e}_{i,2}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,2} \end{bmatrix}) \\ &= \hat{\mathbf{u}}_i^T \mathbf{s} + D(\mathbf{e}_{i,1}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,1} \end{bmatrix} + \mathbf{e}_{i,2}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,2} \end{bmatrix}) \in \mathbb{Z}_q^n. \end{aligned}$$

According to the Lagrangian interpolation formula in Shamir secret sharing scheme $\mathbf{u} = \Sigma_{i \in P} L_i \cdot \hat{\mathbf{u}}_i$ to recover

$$\begin{aligned} \mathbf{c}^{''} &= \sum_{i \in P} L_i \mathbf{c}_i^{'} \\ &= \sum_{i \in P} L_i (\hat{\mathbf{u}}_i^T \mathbf{s} + D(\mathbf{e}_{i,1}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,1} \end{bmatrix} + \mathbf{e}_{i,2}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,2} \end{bmatrix})) \\ &= \mathbf{u}^T \mathbf{s} + \sum_{i \in P} DL_i (\mathbf{e}_{i,1}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,1} \end{bmatrix} + \mathbf{e}_{i,2}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,2} \end{bmatrix}) \in \mathbb{Z}_q^n \end{aligned}$$

then

$$\begin{aligned} w &= c_0 - \mathbf{c}^{''} \\ &= \mathbf{u}^T \mathbf{s} + Dx + M \lfloor \frac{q}{2} \rfloor \\ &\quad - \mathbf{u}^T \mathbf{s} - \sum_{i \in J} DL_i (\mathbf{e}_{i,1}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,1} \end{bmatrix} + \mathbf{e}_{i,2}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,2} \end{bmatrix}) \\ &= M \lfloor \frac{q}{2} \rfloor + Dx - \sum_{i \in J} DL_i (\mathbf{e}_{i,1}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,1} \end{bmatrix} + \mathbf{e}_{i,2}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,2} \end{bmatrix}) \\ &= M \lfloor \frac{q}{2} \rfloor + error \in \mathbb{Z}_q. \end{aligned}$$

As in [29], we need to set the parameters m, q, α, σ to ensure that the error term < q/5:

$$\begin{aligned} & q = \alpha m(\|R\|+1)(f+1)((f+l)!)^4 \cdot \omega(\sqrt{2m}) \\ & m = 6n^{1+\delta}, \qquad \sigma = m \cdot \omega(\sqrt{\log 2n}) \\ & \alpha = (\sigma\sqrt{m}(\|R\|+1)(f+1)((f+l)!)^4 \cdot \omega\sqrt{m})^{-1} \end{aligned}$$

and round up q to the nearest larger prime number, and m to the nearest larger integer. Here we assume that δ is such that $n^{1+\delta} > \lceil (n+1)\log q + \omega(\log n) \rceil$.

4.2 Security Analysis

Under the LWE assumption in the standard model, we prove that our construction is secure, and the specific process is as follows.

Theorem 2. *If there is a PPT adversary $\mathcal{A}$ with advantage $\epsilon > 0$ against the selective security game for the RABE scheme described above, then there is a PPT algorithm $\mathcal{B}$, which decides the LWE problem with advantage $\epsilon/2$.*

Proof. Suppose that the adversary $\mathcal{A}$ has a probability polynomial time algorithm that can selectively attack the scheme, the adversary breaks through the above scheme with advantage ε, then we construct an algorithm $\mathcal{B}$ that can distinguish the decision $(\mathbb{Z}_q, n, \chi) - LWE$ problem

Table 2: Efficiency comparison

	Chen's scheme [8]	Zhang's scheme [29]	Our scheme
Private Key Size	$O(\log N)\cdot\widetilde{O}(n^{\varepsilon+1})$	$\widetilde{O}(n)$	$O(\log^2 M)\cdot\widetilde{O}(n^{\eta+1})$
Key Update Size	$r_u \log\frac{N}{r_u}\cdot\widetilde{O}(n^{\delta+1})$	–	$r_a \log\frac{M}{r_a}\cdot\widetilde{O}(n^{\delta+1})$
Public Key Size	$\widetilde{O}(n^{\varepsilon+2})$	$\widetilde{O}(n^{\delta+2})$	$\widetilde{O}(n^{\delta+\eta+2})$
Cipertext Size	$\widetilde{O}(n^{\varepsilon+1})$	$\widetilde{O}(n^{\delta+1})$	$\widetilde{O}(n^{\delta+\eta+1})$

with advantage ε. Recall *Definition* 6 provides an instance of the LWE problem as a sample oracle $\mathcal{O}$, for some secret key $s\in\mathbb{Z}_q^n$, which can be either truly random $\mathcal{O}_\$$ or noisy pseudorandom $\mathcal{O}_s$. The simulator $\mathcal{B}$ uses the adversary $\mathcal{A}$ to distinguish between the two, and proceeds as follows:

Instance. $\mathcal{B}$ requests from $\mathcal{O}$ and receives a fresh pair $(\mathbf{u}_i, v_i)\in\mathbb{Z}_q^n\times\mathbb{Z}_q$, for each $i=0,\ldots,m$.

Init. The adversary $\mathcal{A}$ announces to $\mathcal{B}$ the challenge access structure (W^*,k^*) and a revocation list RL_j on time period t^*.

Setup. The simulator $\mathcal{B}$ sets the public parameters as follows:

1) Let an attribute set $\mathcal{Q}=\{1,2,\ldots,f\}$, a default attribute set $\mathcal{M}=\{f+1,f+2,\ldots,f+l\}$ and $\mathcal{Q}'=\mathcal{Q}\cup\mathcal{M}$.
2) The adversary submits an access control policy $(W^*,\ k^*)$ to $\mathcal{B}$, where $1\le k^*\le min(|W^*|,l)$. Let $W'=W^*\cup\{f+1,f+2,\ldots,f+l+1-k^*\}$.
3) After the simulator $\mathcal{B}$ receives (W^*,k^*), the $f+l$ uniformly random matrixs $\mathbf{a}_i^*$ are chosen. Using $\mathbf{TrapdoorGen}(q,n)$ to generate $(\mathbf{C}_1,\mathbf{T}_{\mathbf{C}_1})$, $(\mathbf{C}_2,\ \mathbf{T}_{\mathbf{C}_2})$, For $i\in W'$, the simulator $\mathcal{B}$ randomly choose $\mathbf{R}_{i,1}^*,\ \mathbf{R}_{i,2}^*\in\{-1,1\}^{m\times m}$, calculate $\mathbf{B}_1=\mathbf{A}_0\mathbf{R}_{i,1}^*-\mathrm{H}(\mathbf{a}_i^*)\mathbf{C}_1$, $\mathbf{B}_2=\mathbf{A}_0\mathbf{R}_{i,2}^*-\mathrm{H}(\mathbf{t}^*)\mathbf{C}_2$ and give the public parameters $pp=(\mathbf{A_0},\mathbf{B_1},\mathbf{B_2},\mathbf{C_1},\mathbf{C_2},\{\mathbf{a}_i^*\}_{i\in\mathcal{Q}'},\mathbf{u},\mathrm{H})$ to $\mathcal{A}$.

Phase 1. The simulator $\mathcal{B}$ can use the trapdoor $\mathbf{T}_{\mathbf{C}_1}$,$\mathbf{T}_{\mathbf{C}_2}$ to respond to private key queries:

1) When the adversary's query attributes $\mathcal{G}\in\mathcal{Q}$ satisfies the access control policy (W^*,k^*), $\mathcal{B}$ returns $\perp$.
2) When the adversary's query attributes $\mathcal{G}\in\mathcal{Q}$ doesn't satisfy the access control policy (W^*,k^*), i.e., $|\mathcal{G}\cap W^*|\le k^*-1$, let $\mathcal{G}'=\mathcal{G}\cup\{f+1,f+2,\ldots,f+l\}$. $|\mathcal{G}'\cap W'|\le d$ because of $W'=W\cup\{f+1,f+2,\ldots f+l+1-k\}$. Choose a subset $\hat{\mathcal{G}}$, satisfy $(\mathcal{G}'\cap W')\subseteq\hat{\mathcal{G}}\subset\mathcal{G}'$, $|\ \hat{\mathcal{G}}\ |=l$.
3) For $i\in\hat{\mathcal{G}}$, define $\mathbf{F}_i=(\mathbf{A}_0\mid\mathrm{H}(\mathbf{a}_i)\mathbf{C_1}+\mathbf{B}_1)$, $\mathbf{F}_t=(\mathbf{A}_0\mid\mathrm{H}(\mathbf{t})\mathbf{C_2}+\mathbf{B}_2)$. Choose $\mathbf{e}_{i,1},\mathbf{e}_{i,2}\leftarrow\mathcal{D}_{\mathbb{Z}_q^{2m},\sigma}$, Calculate $\hat{\mathbf{u}}_{i,1}=\mathbf{F}_i\mathbf{e}_{i,1}$, $\hat{\mathbf{u}}_{i,2}=\mathbf{F}_t\mathbf{e}_{i,2}$, $\hat{\mathbf{u}}_i=\hat{\mathbf{u}}_{i,1}+\hat{\mathbf{u}}_{i,2}$.
4) Choose n polynomials of degree d, that is, $p_1(x),\ldots,p_n(x)\in\mathbb{Z}_q[x]$ such that $\mathbf{u}=(p_1(0),\ldots,p_n(0))^T$. For every $i\in\hat{\mathcal{G}}$, $\hat{\mathbf{u}}_i=(p_1(x),\ldots,p_n(x))^T$, we can recover the polynomial $p_1(x),\ldots,p_n(x)\in\mathbb{Z}_q[x]$ by using the Lagrange interpolation formula.
5) If $i\in\mathcal{G}'/\hat{\mathcal{G}}$, that is $i\notin W'$, then

$$\begin{aligned}\mathbf{F}_i &= (\mathbf{A}_0\mid\mathrm{H}(\mathbf{a}_i)\mathbf{C_1}+\mathbf{B}_1)\\ &= (\mathbf{A}_0\mid\mathbf{A}_0\mathbf{R}_{i,1}^*+(\mathrm{H}(\mathbf{a}_i)-\mathrm{H}(\mathbf{a}_i^*))\mathbf{C_1}),\\ \mathbf{F}_t &= (\mathbf{A}_0\mid\mathrm{H}(\mathbf{t})\mathbf{C_2}+\mathbf{B}_2)\\ &= (\mathbf{A}_0\mid\mathbf{A}_0\mathbf{R}_{i,2}^*+(\mathrm{H}(\mathbf{t})-\mathrm{H}(\mathbf{t}^*))\mathbf{C_2}).\end{aligned}$$

There is the FRD's definition in Section 3.6, $(\mathrm{H}(\mathbf{a}_i)-\mathrm{H}(\mathbf{a}_i^*))$ and $(\mathrm{H}(\mathbf{t})-\mathrm{H}(\mathbf{t}^*))$ are all full rank matrix. Therefore, $\mathbf{T}_{\mathbf{C}_1}$ and $\mathbf{T}_{\mathbf{C}_2}$ are also trapdoors for $\Lambda_q^\perp(\mathbf{C}'_1)$ and $\Lambda_q^\perp(\mathbf{C}'_2)$ respectively, where $\mathbf{C}'_1=(\mathrm{H}(\mathbf{a}_i)-\mathrm{H}(\mathbf{a}_i^*))\mathbf{C}_1$ and $\mathbf{C}'_2=(\mathrm{H}(\mathbf{t})-\mathrm{H}(\mathbf{t}^*))\mathbf{C}_2$. Run the SampleRight algorithm:

$$\begin{aligned}\mathbf{e}_{i,1} &\leftarrow \mathrm{SampleRight}(\mathbf{A}_0,\mathbf{C}'_1,\mathbf{R}_{i,1}^*,\mathbf{T}_{\mathbf{C}_1},\hat{\mathbf{u}}_{i,1},\sigma),\\ \mathbf{e}_{i,2} &\leftarrow \mathrm{SampleRight}(\mathbf{A}_0,\mathbf{C}'_2,\mathbf{R}_{i,2}^*,\mathbf{T}_{\mathbf{C}_2},\hat{\mathbf{u}}_{i,2},\sigma),\end{aligned}$$

return private key $(\mathbf{e}_{i,1},\mathbf{e}_{i,2})$.

Challenge. The adversary sends two message bits $M_0,M_1\in\{0,1\}$ and an access structure W^* to the simulator $\mathcal{B}$ and $\mathcal{B}$ randomly choose $b\in\{0,1\}$, calculate $c_0=Dv_0+M_b\lfloor q/2\rfloor\in\mathbb{Z}_q$, $\mathbf{v}_i=(v_1,v_2,\ldots,v_m)^T\in\mathbb{Z}_q^m$. For $i\in W'$, calculate $\mathbf{c}_{i,1}=D(\mathbf{R}_{i,1}^*)^T\mathbf{v}_i$, $\mathbf{c}_{i,2}=D(\mathbf{R}_{i,2}^*)^T\mathbf{v}_i$. Return challenge ciphertext

$$c^*=(c_0,\{\mathbf{c}_{i,1}\}_{i\in W'},\{\mathbf{c}_{i,2}\}_{i\in W'},W^*,\mathbf{t}^*).$$

Phase 2. Similar as Phase 1, the adversary $\mathcal{A}$ continues to initiate a request to $\mathcal{B}$.

Guess. The adversary $\mathcal{A}$ outputs a guess b'. The simulator $\mathcal{B}$ uses the guess to determine an answer on the LWE oracle: Output yes if b' = b, else output no.

For each $i \in W'$, we have

$$\begin{aligned}
\mathbf{c}_{i,1} &= D(\mathbf{R}_{i,1}^*)^T \mathbf{v}_i \\
&= D(\mathbf{R}_{i,1}^*)^T (\mathbf{A}_0^T \mathbf{s} + \mathbf{y}) \\
&= (\mathbf{A}_0 \mathbf{R}_{i,1}^*)^T (D\mathbf{s}) + D(\mathbf{R}_{i,1}^*)^T \mathbf{y} \\
&= (\mathrm{H}(\mathbf{a}_i^*)\mathbf{C}_1 + \mathbf{B}_1)^T (D\mathbf{s}) + D(\mathbf{R}_{i,1}^*)^T \mathbf{y} \\
\mathbf{c}_{i,2} &= D(\mathbf{R}_{i,2}^*)^T \mathbf{v}_i \\
&= D(\mathbf{R}_{i,2}^*)^T (\mathbf{A}_0^T \mathbf{s} + \mathbf{y}) \\
&= (\mathbf{A}_0 \mathbf{R}_{i,2}^*)^T (D\mathbf{s}) + D(\mathbf{R}_{i,2}^*)^T \mathbf{y} \\
&= (\mathrm{H}(\mathbf{t}^*)\mathbf{C}_2 + \mathbf{B}_2)^T (D\mathbf{s}) + D(\mathbf{R}_{i,2}^*)^T \mathbf{y}
\end{aligned}$$

Because the adversary could not obtain the $\{\mathbf{R}_{i,1}^*, \mathbf{R}_{i,2}^*\}_{i \in \mathcal{Q}'}$ from the public key, the adversary cannot distinguish actual ciphertext distribution from O_s or $O_\$$. If the adversary can have a non-negligible probability to guess the value of b, then there is an algorithm to solve the **LWE** problem. □

5 Performance Evaluation

We give an efficiency comparison with other schemes in **Table** 2. Here, $\mathcal{M}$ is the number of attributes, N is the number of users, r_u denotes the number of revoked users, r_a denotes the number of revoked attributes, δ is a small constant such that $\delta < 1/2$, ε is a small constant and $n^\varepsilon > O(\log N)$, η is a small constant and $n^\eta > O(\log \mathcal{M})$. Compared with Chen's scheme that can only achieve one-to-one communication, our scheme can achieve one-to-many communication. Compared with Zhang's scheme, our scheme supports attribute revocation.

6 Conclusions

In this paper, we propose a ciphertext policy attribute-based encryption scheme from lattices with efficient attribute revocation, which resists quantum attacks. The scheme builds a binary tree structure to update the legitimate user's key, and obtains the attribute revocation. We prove that our construction is secure against selective-attribute attacks in the standard model and security can be reduced to hardness of learning with error assumption. Although our scheme achieves a flexible threshold access control, how to construct a more complex access structure (such as access tree structure, circuit structure, etc.) is the work that will be carried out in the next step. In addition, how to design a scheme against adaptive attacks is also our future work.

Acknowledgments

This work was supported by Department of Education of Zhejiang Province of China (No.Y201636547), higher education research of Hangzhou Dianzi University in 2017, Key Research Project of Zhejiang Province (No.2017C01062), Guangxi Key Laboratory of Cryptography and Information Security (No.GCIS201718), the Opening Project of Guangdong Provincial Key Laboratory of Information Security Technology(No.2017B030314131-05), the Fund of Lab of Security Insurance of Cyberspace, Sichuan Province (No.szjj2017-055), research on the innovation of training mode of the first level subject of Cyberspace Security (No.YB201767), and cyberspace subject of Hangzhou Dianzi University (No.GK168800225075).

References

[1] S. Agrawal, D. Boneh, and X. Boyen, "Lattice basis delegation in fixed dimension and shorter-ciphertext hierarchical IBE," in *Advances in Cryptology (CRYPTO'10), 30th Annual Cryptology Conference*, pp. 98–115, Aug. 2010.

[2] S. Agrawal, X. Boyen, V. Vaikuntanathan, P. Voulgaris, and H. Wee, *Functional encryption for threshold functions (or fuzzy IBE) from lattices*. Darmstadt, Germany: Springer Berlin Heidelberg, 2012.

[3] D. Apon, X. Fan, and F. H. Liu, "Deniable attribute based encryption for branching programs from LWE," in *Theory of Cryptography - 14th International Conference (TCC'16)*, pp. 299–329, 2016.

[4] R. Bendlin and I. Damgård, "Threshold decryption and zero-knowledge proofs for lattice-based cryptosystems," in *Theory of Cryptography, 7th Theory of Cryptography Conference (TCC'10)*, pp. 201–218, 2010.

[5] A. Boldyreva, V. Goyal, and V. Kumar, "Identity-based encryption with efficient revocation," in *Proceedings of the 2008 ACM Conference on Computer and Communications Security (CCS'08)*, pp. 417–426, 2008.

[6] X. Boyen, *Attribute-Based Functional Encryption on Lattices*, Tokyo, Japan: Springer Berlin Heidelberg, 2013.

[7] X. Boyen and Q. Y. Li, *Attribute-Based Encryption for Finite Automata from LWE*, Kanazawa, Japan: Springer International Publishing, 2015.

[8] J. Chen, H. W. Lim, S. Ling, H. X. Wang, and K. Nguyen, "Revocable identity-based encryption from lattices," in *17th Australasian Conference on Information Security and Privacy (ACISP'12)*, pp. 390–403, 2012.

[9] J. S. Chen, C. Y. Yang, and M. S. Hwang, "The capacity analysis in the secure cooperative communication system," *International Journal of Network Security*, vol. 19, no. 6, pp. 863–869, 2017.

[10] P. S. Chung, C. W. Liu, and M. S. Hwang, "A study of attribute-based proxy re-encryption scheme in cloud environments," *International Journal of Network Security*, vol. 16, no. 1, pp. 1–13, 2014.

[11] X. B. Fu, X. Y. Nie, T. Wu, and F. G. Li, "Large universe attribute based access control with efficient decryption in cloud storage system," *Journal of Systems and Software*, vol. 135, pp. 157–164, 2018.

[12] S. Gorbunov, V. Vaikuntanathan, and H. Wee, "Attribute-based encryption for circuits," in *Symposium on Theory of Computing Conference (STOC'13)*, pp. 545–554, 2013.

[13] S. Gorbunov and D. Vinayagamurthy, *Riding on Asymmetry: Efficient ABE for Branching Programs*, New Zealand: Springer Berlin Heidelberg, 2015.

[14] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proceedings of the 13th ACM Conference on Computer and Communications Security (CCS'06)*, pp. 89–98, Oct. 2006.

[15] M. A. Hamza, J. F. Sun, X. Y. Nie, Z. Q. Qin, and H. Xiong, "Revocable abe with bounded ciphertext in cloud computing," *International Journal of Network Security*, vol. 19, no. 6, pp. 973–983, 2017.

[16] J. Hur and K. N. Dong, "Attribute-based access control with efficient revocation in data outsourcing systems," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 7, pp. 1214–1221, 2011.

[17] V. Kuchta and O. Markowitch, "Multi-authority distributed attribute-based encryption with application to searchable encryption on lattices," in *Paradigms in Cryptology (Mycrypt'16). Malicious and Exploratory Cryptology - Second International Conference*, pp. 409–435, 2016.

[18] C. Mao L. Liu, Z. Cao, "A note on one outsourcing scheme for big data access control in cloud," *International Journal of Electronics and Information Engineering*, vol. 9, no. 1, pp. 29–35, 2018.

[19] C. C. Lee, P. S. Chung, and M. S. Hwang, "A survey on attribute-based encryption schemes of access control in cloud environments," *International Journal Network Security*, vol. 15, no. 4, pp. 231–240, 2013.

[20] M. Aref M. Bayat, "An attribute based key agreement protocol resilient to kci attack," *International Journal of Electronics and Information Engineering*, vol. 2, no. 1, pp. 10–20, 2015.

[21] H. Ma, T. Peng, and Z. H. Liu, "Directly revocable and verifiable key-policy attribute-based encryption for large universe," *International Journal of Network Security*, vol. 19, no. 2, pp. 272–284, 2017.

[22] D. Micciancio and C. Peikert, *Trapdoors for Lattices: Simpler, Tighter, Faster, Smaller*, Cambridge, UK: Springer Berlin Heidelberg, 2012.

[23] M. Pirretti, P. Traynor, P. D. McDaniel, and B. Waters, "Secure attribute-based systems," in *Proceedings of the 13th ACM Conference on Computer and Communications Security (CCS'06)*, pp. 99–112, 2006.

[24] A. Sahai, H. Seyalioglu, and B. Waters, "Dynamic credentials and ciphertext delegation for attribute-based encryption," *IACR Cryptology ePrint Archive*, vol. 2012, pp. 437, 2012.

[25] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in *Advances in Cryptology (EUROCRYPT'05), 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 457–473, Denmark, May 2005.

[26] J. Singh, "Cyber-attacks in cloud computing: A case study," *International Journal of Electronics and Information Engineering*, vol. 1, no. 2, pp. 78–87, 2014.

[27] Y. H. Wang, Y. Q. Liu, and K. Wang, "A secure and efficient ciphertext encryption scheme based on attribute and support strategy dynamic update via hybrid encryption method," *International Journal of Network Security*, vol. 20, no. 5, pp. 907–913, 2018.

[28] N. I. Wu and M. S. Hwang, "Development of a data hiding scheme based on combination theory for lowering the visual noise in binary images," *Displays*, vol. 49, pp. 116–123, 2017.

[29] J. Zhang, Z. F. Zhang, and A. J. Ge, "Ciphertext policy attribute-based encryption from lattices," in *Symposium on Information, Compuer and Communications Security (ASIACCS'12)*, pp. 16–17, 2012.

[30] L. Y. Zhang and H. J. Yin, "Recipient anonymous ciphertext-policy attribute-based broadcast encryption," *International Journal of Network Security*, vol. 20, no. 1, pp. 168–176, 2018.

Biography

Kang Yang is currently pursuing his master's degreee in the computer science and technology, Hangzhou Dianzi University. His research interests include cloud computing security and cryptography.

Guohua Wu is a professor at Hangzhou Dianzi University, and he received the Ph.D. degree from Zhejiang University in 1998. His research interests include cryptography and information hiding.

Chengcheng Dong is currently pursuing his master's degreee in the computer science and technology, Hangzhou Dianzi University. His research interests include cloud computing security and cryptography.

Xingbing Fu is a lecturer, and he received the Ph.D. degree from University of Electronic Science and Technology of China (UESTC) in 2016. His research interests include cloud computing and cryptography.

Fagen Li is a professor, and he received the Ph.D. degree in Cryptography from Xidian University, Xi'an, P.R. China in 2007. His research interests include cryptography and network security.

Ting Wu received the Ph.D. degree in Shandong University in 2002. He is a Professor at Hangzhou Dianzi University. His research interests include cryptography and information security.

Correlation Functions of $m-$Sequences of Different Lengths

Zepeng Zhuo[1], Jinfeng Chong[1,2], and Lei Yu[3]
(Corresponding author: Jinfeng Chong)
School of Mathematical Sciences, Huaibei Normal University[1]
Information College, Huaibei Normal Universtiy[2]
School of Computer Science and Technology, Huaibei Normal Universtiy[3]
Huaibei, Anhui 235000, China
(Email: cjf791009@sohu.com)
(Received Oct. 16, 2018; Revised and Accepted Jan. 23, 2019; First Online Mar. 4, 2019)

Abstract

The sequences over a finite field F_p with good correlation properties are important applications in coding, communication, and cryptography. The maximal period sequences($m-$ sequences) and their decimations are widely used to design sequence families with low correlation. In this paper, the correlation functions on $m-$sequences of different lengths are investigated. Two classes of $m-$sequences of different lengths are considered, and some properties of the correlation functions between these $m-$sequences are presented.

Keywords: Autocorrelation Function; Binary Sequence; Cross-correlation Function; m−sequence; Perfect Sequence

1 Introduction

Correlation is a measure of the similarity or relatedness. If properly normalized, the correlation measure is a real number between -1 and $+1$. A correlation value of -1 indicates that the two phenomena are diametrically opposite while a correlation value 0 means that they are uncorrelated, and a correlation value $+1$ means that they are identical. In other sources, the correlation between two sets of data is called their covariance. In linear algebra, the correlation between two vectors is their (normalized) dot product [3].

Let $\mathbf{a} = (a_0, a_1, \cdots, a_{N-1})$ and $\mathbf{b} = (b_0, b_1, \cdots, b_{N-1})$ be two binary sequences of period N. The *(periodic) cross-correlation function* between these two sequences at shift τ, where $0 \le \tau \le N-1$, is defined by

$$\mathcal{C}_{\mathbf{a},\mathbf{b}}(\tau) = \sum_{i=0}^{N-1} (-1)^{a_{i+\tau}+b_i}, \tag{1}$$

where the subscripts are reduced modulo N, that is, $\mathcal{C}_{\mathbf{a},\mathbf{b}}(\tau)$ is the dot product of two vectors

$$((-1)^{a_\tau}, (-1)^{a_{\tau+1}}, \cdots, (-1)^{a_{\tau+N-1}}).$$

and

$$((-1)^{b_0}, (-1)^{b_1}, \cdots, (-1)^{b_{N-1}}).$$

If the sequences $\mathbf{a}$ and $\mathbf{b}$ are the same, we call $\mathcal{C}_{\mathbf{a},\mathbf{a}}(\tau)$ the *(periodic) autocorrelation function* of $\mathbf{a}$, denoted by $\mathcal{A}_{\mathbf{a}}(\tau)$. $\mathcal{A}_{\mathbf{a}}(\tau)$ measures the amount of similarity between the sequence and its phase shift. This is always the highest for $\tau = 0$, because

$$\mathcal{A}_{\mathbf{a}}(0) = \sum_{i=0}^{N-1} (-1)^{a_i+a_i} = N.$$

During the last decades, many applications of sequences with low correlation have been found in coding, communication and cryptography [1,4,5,9,13,15,16]. Using sequences with low(auto and cross)correlation values, the interference of different users during the transmission can be reduced. Therefore, sequences with low correlation have been an important research problem enjoying considerable interests. If the autocorrelation properties are optimum, the sequences would been called perfect. Conventional autocorrelation functions have two different definitions: periodic and aperiodic autocorrelations. Traditionally, the studies of the periodic autocorrelation of a binary sequence, especially about the sequences families of code-division multiple access communication (CDMA) system, have attracted more and more attention in the research of this field. However, the aperiodic autocorrelation is considered to better characterize a binary sequence for more realistic communication systems [15].

A well-studied problem is to find the cross-correlation function between two binary $m-$sequences $\{s_t\}$ and $\{s_{dt}\}$ of the same period $2^m - 1$ that differs by a decimation d such that $\mathtt{gcd}(d, 2^m-1) = 1$. A survey of some of the basic researches on the cross-correlation between $m-$sequencs

of the same length can be found [5]. Several sequence families with practical applications use $m-$sequencs of different periods, a prime example is the optimal small family of Kasami sequences. The correlation properties of this family depends on the correlation properties of an $m-$sequence of period 2^m-1 and an $m-$sequence of period $2^{m/2-1}$ where m is even.

A cross-correlation function between two periodic sequences $\mathbf{a}=\{a_i\}$, of period s, and $\mathbf{b}=\{b_i\}$, of period t, over F_2 can be defined as [3].

$$\mathcal{CC}_{\mathbf{a},\mathbf{b}}(\tau)=\sum_{i=0}^{N-1}(-1)^{a_{i+\tau}+b_i}, \tau=0,1,\cdots, \tag{2}$$

where $N=\mathtt{lcm}[s,t]$.

In [10], Ness and Helleseth studied the cross correlation between an $m-$sequencs $\{s_t\}$ of length $n=2^m-1$ and an $m-$sequence $\{u_{dt}\}$ of length 2^k-1, where $m=2k$ and $\gcd(d,2^k-1)=1$. Here $\{u_t\}$ denotes the $m-$sequence which used in constructing the small family of Kasami sequence. The cross-correlation functions between $m-$sequences of different lengths were investigated in [2, 6–8, 10–12, 14]. The first infinite family of pairs of $m-$sequences with four-valued cross-correlation was constructed and the complete correlation distribution of this family was also determined [11]. From the above studies, we are motivated by [2,6–8,10–12,14] to study correlation functions of $m-$sequences of different lengths.

2 Preliminaries

Definition 1. *If the autocorrelation function $\mathcal{A}_{\boldsymbol{a}}(\tau)$ is two-valued, given by*

$$\mathcal{A}_{\boldsymbol{a}}(\tau)=\begin{cases} N, & \text{if } \tau\equiv 0 \pmod N,\\ K, & \text{if } \tau\not\equiv 0 \pmod N,\end{cases} \tag{3}$$

where K is a constant. If $K=-1$, N is an odd and $K=0$, N is an even, then we say that the sequence $\boldsymbol{a}$ has the (ideal) 2-level autocorrelation function, and $\boldsymbol{a}$ is called the perfect sequence.

If $N=2^n-1$, the autocorrelation function $\mathcal{A}_{\mathbf{a}}(\tau)$ is two-valued and is given by

$$\mathcal{A}_{\mathbf{a}}(\tau)=\begin{cases} 2^n-1, & \text{if } \tau\equiv 0 \pmod{2^n-1},\\ -1, & \text{if } \tau\not\equiv 0 \pmod{2^n-1}.\end{cases} \tag{4}$$

Binary sequences with 2-level autocorrelation have many applications in communication such as radar distance ranging, hardware testing, coding theory, and cryptography. A binary sequence of period 2^n-1 with 2-level autocorrelation corresponds to a cyclic Hadamard difference set with parameters $(2^n-1,2^{n-1}-1,2^{n-2}-1)$. A sequence $\mathbf{a}$ is called *balanced* if the number of ones and zeros in each period is $n/2$ if n is even or $(n\pm 1)/2$ if n is odd. Balanced sequences with autocorrelation -1 are widely used in communications and cryptography. The $m-$sequences is the first class of binary sequences of period 2^n-1 with 2-level autocorrelation for any positive integer n, and it corresponds to the Singer Hamamard difference sets which were discovered by Singer in 1938. Golomb found $m-$sequences from the approach linear feedback shift register sequences in 1954. These sequences have several other common names, e.g., *pseudo-noise(PN) sequences* and *maximal length shift register sequences*. The importance of $m-$sequences is largely due to the part which they called pseudo randomness properties, *i.e.*, properties that make $m-$sequences behave like sequences whose elements are chosen at random.

Let p be any prime, r be a positive integer, $q=p^r$. Let $\omega=e^{2\pi i/p}$ be a primitive pth root of unity. The *(canonical) additive character* of F_{p^r} is defined by

$$\chi(x)=e^{2\pi i Tr(x)/p}, x\in \mathrm{F}_{p^r}, \tag{5}$$

where $Tr(x)$ is the trace function from F_{p^r} to F_p, given by

$$Tr(x)=x+x^p+\cdots+x^{p^{r-1}}, x\in \mathrm{F}_{p^r}.$$

Definition 2. *A cross-correlation function between two periodic sequences $\boldsymbol{a}$, of period s, and $\boldsymbol{b}$, of period t, over F_q can be defined as*

$$\mathcal{CC}_{\boldsymbol{a},\boldsymbol{b}}(\tau)=\sum_{i=0}^{N-1}\chi(a_{i+\tau})\chi^*(b_i), \tau=0,1,\cdots, \tag{6}$$

where $\chi^*(x)=(\chi(x))^*$, the complex conjugation of $\chi(x)$, and $N=\mathtt{lcm}[s,t]$. In particular, the cross-correlation function of $\boldsymbol{a}$ and $\boldsymbol{b}$ defined by Equation (6) becomes Equation (2) and the following formulae:

$$\mathcal{CC}_{\boldsymbol{a},\boldsymbol{b}}(\tau)=\sum_{i=0}^{N-1}\omega^{a_{i+\tau}-b_i}, \tau=0,1,\cdots, \text{for } \ \mathrm{q=p>2}. \tag{7}$$

In [3], the basic properties of cross-correlation function which defined as Equation (7) were studied. We list in the following:

Property 1[1]: Let $\mathbf{a}$ and $\mathbf{b}$ be two periodic sequences over F_p with periods $s=p^n-1$ and t, where $t|s$ respectively. Let L be the left shift operator.

1) $\mathcal{CC}_{\mathbf{a},\mathbf{b}}(\tau)=\mathcal{CC}_{\mathbf{a},\mathbf{b}}(\tau+kt), k=0,1,\cdots$

2) $\mathcal{CC}_{L^k\mathbf{a},L^k\mathbf{b}}(\tau)=\mathcal{CC}_{\mathbf{a},\mathbf{b}}(\tau), 0\le k<s.$

3) $\mathcal{CC}_{L^i\mathbf{a},L^j\mathbf{b}}(\tau)=\mathcal{CC}_{\mathbf{a},\mathbf{b}}(\tau+i-j), 0\le i,j<s$, where $\tau+i-j$ is reduced modulo s.

4) $\mathcal{CC}_{\mathbf{a},\mathbf{b}}(\tau)=\overline{\mathcal{CC}_{\mathbf{b},\mathbf{a}}(-\tau)}$. In particular, if $p=2$, then $\mathcal{CC}_{\mathbf{a},\mathbf{b}}(\tau)=\mathcal{CC}_{\mathbf{b},\mathbf{a}}(-\tau)$, where $-\tau$ is reduced modulo s.

5) If $p=2$, according to the assumption, $t|2^n-1$. Let $d>1$ satisfying $\gcd(d,t)=1$, then

$$\mathcal{CC}_{\mathbf{a},\mathbf{b}^{(d^{-1})}}(\tau)=\mathcal{CC}_{\mathbf{b},\mathbf{a}^{(d)}}(-d^{-1}\tau).$$

3 Main Results

In this section, we mainly discuss the properties of the cross-correlation function $\mathcal{CC}_{\mathbf{a},\mathbf{b}}(\tau)$ defined as Equation (2).

3.1 The $\gcd(s,t)=1$ Property

3.1.1 Product Sequences

In [15], Yu and Gong discussed the applications of the perfect binary sequence for binary sequences with optimal (periodic) autocorrelation. Let $\mathbf{a}$ and $\mathbf{b}$ be binary sequences of periods N_1 and N_2 respectively, where $\gcd(N_1,N_2)=1$. Then the product sequence $\mathbf{p}=\mathbf{a}+\mathbf{b}=(p_0,p_1,\cdots,p_{N-1})$ of period $N=N_1N_2$ is defined by the component-wise addition of $p_i=a_i+b_i \pmod 2$, $0\le i\le N-1$. The (periodic) autocorrelation $\mathcal{A}_{\mathbf{p}}(\tau)$ of the product sequence is given by

$$\begin{aligned}\mathcal{A}_{\mathbf{p}}(\tau) &= \sum_{i=0}^{N-1}(-1)^{p_{i+\tau}+p_i}\\ &= \sum_{i=0}^{N_1N_2-1}(-1)^{a_{i+\tau}+b_{i+\tau}+a_i+b_i}\\ &= \left[\sum_{j=0}^{N_1-1}(-1)^{a_{j+\tau}+a_j}\right]\cdot\left[\sum_{k=0}^{N_2-1}(-1)^{b_{k+\tau}+b_k}\right]\\ &= \mathcal{A}_{\mathbf{a}}(\tau)\cdot\mathcal{A}_{\mathbf{b}}(\tau),\end{aligned}$$

where $0\le\tau\le N-1$, and the indices of a sequence are computed modulo its own period.

That is, the (periodic) autocorrelation functions of the products of periodic sequences of relatively prime lengths are themselves the products of the individual autocorrelation functions.

Example 1. *Let* $\boldsymbol{a}=110,\boldsymbol{b}=11010$, *then*

$$\boldsymbol{a}: 110110110110110,$$

$$\boldsymbol{b}: 110101101011010,$$

$$\boldsymbol{p}: 000011011101100.$$

We compute their autocorrelation functions as following:

$$\begin{aligned}&\mathcal{A}_{\mathbf{a}}(0),\mathcal{A}_{\mathbf{a}}(1),\cdots,\mathcal{A}_{\mathbf{a}}(14)\\ =\ &3,-1,-1,3,-1,-1,3,-1,-1,3,-1,-1,3,-1,-1,\\ &\mathcal{A}_{\mathbf{b}}(0),\mathcal{A}_{\mathbf{b}}(1),\cdots,\mathcal{A}_{\mathbf{b}}(14)\\ =\ &5,-3,1,1,-3,5,-3,1,1,-3,5,-3,1,1,-3,\\ &\mathcal{A}_{\mathbf{p}}(0),\mathcal{A}_{\mathbf{p}}(1),\cdots,\mathcal{A}_{\mathbf{p}}(14)\\ =\ &15,3,-1,3,3,-5,-9,-1,-1,-9,-5,3,3,-1,3.\end{aligned}$$

Obviously, from the above example, we have $\mathcal{A}_{\mathbf{p}}(\tau)=\mathcal{A}_{\mathbf{a}}(\tau)\cdot\mathcal{A}_{\mathbf{b}}(\tau), 0\le\tau\le 14$.

3.1.2 Cross-Correlation Functions Between $m-$Sequences of Relatively Prime Lengths

In this section, we discuss the cross-correlation function between an arbitrary pair of $m-$sequences whose periods are relatively prime. First, we give the following fact.

Proposition 1. *Let* $\boldsymbol{a}$ *and* $\boldsymbol{b}$ *be binary sequences of periods* s *and* t *respectively, where* $\boldsymbol{gcd}(s,t)=1$. *Let* $\mathcal{CC}_{\boldsymbol{a},\boldsymbol{b}}(\tau)$ *be the cross-correlation function between* $\boldsymbol{a}$ *and* $\boldsymbol{b}$ *defined by Equation (2). Then the* $\mathcal{CC}_{\boldsymbol{a},\boldsymbol{b}}(\tau)$ *is a periodic function, and the period of* $\mathcal{CC}_{\boldsymbol{a},\boldsymbol{b}}(\tau)$ *is equal to* $\boldsymbol{min}(s,t)$.

In fact, without loss of generality, we assume $s<t$. So, $\mathbf{a}$ is a short sequence. If the sequence $\mathbf{a}$ shifts s times later, it will return to the original location again. Hence, the value of $\mathcal{CC}_{\mathbf{a},\mathbf{b}}(\tau)$ won't change any more.

Example 2. *Let* $\boldsymbol{a}=011,\boldsymbol{b}=0010111$, *they are* $m-$*sequences of length 3 and 7 respectively. By Equation (2), we compute*

$$\mathcal{CC}_{\boldsymbol{a},\boldsymbol{b}}(\tau)=\sum_{i=0}^{20}(-1)^{a_{i+\tau}+b_i},\tau=0,1,\cdots$$

Since,

$$\mathbf{a}: 01101101101101101101,$$

$$\mathbf{b}: 00101110010111010111.$$

Then, we have

$$\mathcal{CC}_{\mathbf{a},\mathbf{b}}(0)=1,\mathcal{CC}_{\mathbf{a},\mathbf{b}}(1)=1,\mathcal{CC}_{\mathbf{a},\mathbf{b}}(2)=1,\mathcal{CC}_{\mathbf{a},\mathbf{b}}(3)=1,\cdots$$

Before proceeding, we will give a somewhat technical result about gcd's(*greatest common divisor*) that will be used in the following.

Lemma 1. *Let* a,m,n *be positive integers, and* $a>1$. *Then*

$$\boldsymbol{gcd}(a^m-1,a^n-1)=a^{\boldsymbol{gcd}(m,n)}-1.$$

Proof. If $m=n$, the result is trivial. If $m\ne n$, and $m,n>1$, without loss of generality, we assume $m>n$. Using Division algorithm, then

$$m=qn+r, 0\le r<n.$$

□

We have

$$\begin{aligned}a^m-1 &= a^{qn+r}-1\\ &= a^{qn+r}-a^r+a^r-1\\ &= a^r(a^{qn}-1)+(a^r-1).\end{aligned}$$

Since $(a^n-1)|(a^{qn}-1)$, *i.e.*, $\exists A\in\mathbf{N}, a^{qn}-1=A(a^n-1)$, then $a^m-1=Aa^r(a^n-1)+(a^r-1)$. Therefore, $\gcd(a^m-1,a^n-1)=\gcd(a^n-1,a^r-1)$.

Note that $\gcd(m,n)=\gcd(n,r)$. If $r=0$, then $\gcd(m,n)=n$, the result is true. If $r>0$, then we discuss $\gcd(a^n-1,a^r-1)$ by using the same method. According to Euclid's algorithm, the result is true.

The above result is very powerful. For example, it says that $\texttt{gcd}(3^9-1, 3^8-1) = 3^{\texttt{gcd}(9,8)}-1 = 2$, a fact which is unobvious if we had written $3^9-1 = 19682, 3^8-1 = 6560$. Similarly, using Lemma 1, we can compute many polynomial gcd's effortlessly: $\texttt{gcd}(x^9-1, x^{12}-1) = x^3-1$.

Theorem 1. *Let $\boldsymbol{a}$ and $\boldsymbol{b}$ be binary $m-$sequences of periods 2^m-1 and 2^n-1 respectively, where $\texttt{gcd}(m,n)=1$. Let $\mathcal{CC}_{\boldsymbol{a},\boldsymbol{b}}(\tau)$ be the cross-correlation function between $\boldsymbol{a}$ and $\boldsymbol{b}$ defined by Equation (2). Then*

$$\mathcal{CC}_{\boldsymbol{a},\boldsymbol{b}}(\tau) = 1.$$

Since $\texttt{gcd}(m,n)=1$, according to Lemma 1, we have $\texttt{gcd}(2^m-1, 2^n-1)=1$. Using the knowledge of probability, we calculate $\mathcal{CC}_{\mathbf{a},\mathbf{b}}(\tau)$. Let $\mathbf{a}$ be an $m-$sequence of period P. Hence, the sequence $\mathbf{a}$ satisfies the balance property, *i.e.*, in every period, 0's occur $(P-1)/2$ times and 1's occur $(P+1)/2$ times. So, in every period,

$$\texttt{Pr}(0) = (P-1)/2P, \quad \texttt{Pr}(1) = (P+1)/2P.$$

Note that for two periodic sequences $\mathbf{a} = \{a_i\}, \mathbf{b} = \{b_i\}$ of relatively prime lengths s, t, the cross-correlation function Equation (2) can be defined as

$$\mathcal{CC}_{\mathbf{a},\mathbf{b}}(\tau) = A(\tau) - D(\tau), 0 \le \tau \le st-1,$$

where $A(\tau)$ and $D(\tau)$ denote the number of agreements and disagreements between $\mathbf{a}$'s phase shift $\{a_{i+\tau}\}, 0 \le i \le st-1$ and $\{b_i\}, 0 \le i \le st-1$ respectively. In statistical terms, the cross-correlation function Equation (2) becomes the standard correlation coefficient defined as

$$\mathcal{CC}_{\mathbf{a},\mathbf{b}}(\tau) = (2^m-1)(2^n-1)(\texttt{Pr}\{a_{i+\tau}=b_i\} - \texttt{Pr}\{a_{i+\tau}\neq b_i\}).$$

Proof of Theorem 1: For the sequences $\mathbf{a}$ and $\mathbf{b}$, we have

$$\texttt{Pr}_{\mathbf{a}}(0) = \frac{2^{m-1}-1}{2^m-1}, \ \texttt{Pr}_{\mathbf{a}}(1) = \frac{2^{m-1}}{2^m-1},$$

$$\texttt{Pr}_{\mathbf{b}}(0) = \frac{2^{n-1}-1}{2^n-1}, \ \texttt{Pr}_{\mathbf{b}}(1) = \frac{2^{n-1}}{2^n-1}.$$

Since the two sequences $\mathbf{a}$ and $\mathbf{b}$ are statistically independent, we get four joint probabilities

$$\texttt{Pr}(00) = \texttt{Pr}_{\mathbf{a}}(0)\cdot\texttt{Pr}_{\mathbf{b}}(0) = \frac{(2^{m-1}-1)(2^{n-1}-1)}{(2^m-1)(2^n-1)},$$

$$\texttt{Pr}(01) = \texttt{Pr}_{\mathbf{a}}(0)\cdot\texttt{Pr}_{\mathbf{b}}(1) = \frac{2^{n-1}(2^{m-1}-1)}{(2^m-1)(2^n-1)},$$

$$\texttt{Pr}(10) = \texttt{Pr}_{\mathbf{a}}(1)\cdot\texttt{Pr}_{\mathbf{b}}(0) = \frac{2^{m-1}(2^{n-1}-1)}{(2^m-1)(2^n-1)},$$

$$\texttt{Pr}(11) = \texttt{Pr}_{\mathbf{a}}(1)\cdot\texttt{Pr}_{\mathbf{b}}(1) = \frac{2^{m-1}\cdot 2^{n-1}}{(2^m-1)(2^n-1)}.$$

Therefore,

$$\begin{aligned}\texttt{Pr}\{a_{i+\tau}=b_i\} &= \texttt{Pr}(00)+\texttt{Pr}(11)\\ &= \frac{(2^{m-1}-1)(2^{n-1}-1)+2^{m-1}\cdot 2^{n-1}}{(2^m-1)(2^n-1)},\end{aligned}$$

$$\begin{aligned}\texttt{Pr}\{a_{i+\tau}\neq b_i\} &= \texttt{Pr}(01)+\texttt{Pr}(10)\\ &= \frac{2^{n-1}(2^{m-1}-1)+2^{m-1}(2^{n-1}-1)}{(2^m-1)(2^n-1)}.\end{aligned}$$

Thus,

$$\begin{aligned}\mathcal{CC}_{\mathbf{a},\mathbf{b}}(\tau) &= (2^m-1)(2^n-1)\\ &\cdot (\texttt{Pr}\{a_{i+\tau}=b_i\} - \texttt{Pr}\{a_{i+\tau}\neq b_i\})\\ &= 1.\end{aligned}$$

Example 3: With the notation in **Example 2**. We compute $\mathcal{CC}_{\mathbf{a},\mathbf{b}}(\tau)$ by using Theorem 1. First, we have

$$\texttt{Pr}_{\mathbf{a}}(0) = 1/3, \ \texttt{Pr}_{\mathbf{a}}(1) = 2/3, \ \texttt{Pr}_{\mathbf{b}}(0) = 3/7, \ \texttt{Pr}_{\mathbf{b}}(1) = 4/7,$$

and

$$\begin{aligned}\texttt{Pr}(00) &= \texttt{Pr}_{\mathbf{a}}(0)\cdot\texttt{Pr}_{\mathbf{b}}(0) = 3/21,\\ \texttt{Pr}(01) &= \texttt{Pr}_{\mathbf{a}}(0)\cdot\texttt{Pr}_{\mathbf{b}}(1) = 4/21,\\ \texttt{Pr}(10) &= \texttt{Pr}_{\mathbf{a}}(1)\cdot\texttt{Pr}_{\mathbf{b}}(0) = 6/21,\\ \texttt{Pr}(11) &= \texttt{Pr}_{\mathbf{a}}(1)\cdot\texttt{Pr}_{\mathbf{b}}(1) = 8/21,\end{aligned}$$

then,

$$\begin{aligned}\texttt{Pr}\{a_{i+\tau}=b_i\} &= \texttt{Pr}(00)+\texttt{Pr}(11) = 11/21,\\ \texttt{Pr}\{a_{i+\tau}\neq b_i\} &= \texttt{Pr}(01)+\texttt{Pr}(10) = 10/21.\end{aligned}$$

Therefore,

$$\begin{aligned}\mathcal{CC}_{\mathbf{a},\mathbf{b}}(\tau) &= 21(\texttt{Pr}\{a_{i+\tau}=b_i\} - \texttt{Pr}\{a_{i+\tau}\neq b_i\})\\ &= 1, \ \tau = 0, 1, \cdots\end{aligned}$$

3.2 The $t|s$ Property

In Equation (2), if $t|s$, then we have

$$\mathcal{CC}_{\mathbf{a},\mathbf{b}}(\tau) = \sum_{i=0}^{s-1}(-1)^{a_{i+\tau}+b_i}, \tau = 0, 1, \cdots, \tag{8}$$

In this section, we mainly study the properties of $\mathcal{CC}_{\mathbf{a},\mathbf{b}}(\tau)$ defined as Equation (8).

Lemma 2. *Let $\boldsymbol{a}$ and $\boldsymbol{b}$ be two binary sequences of periods t and s, where $t|s$. Then*

$$\sum_{\tau=0}^{t-1}\mathcal{CC}_{\boldsymbol{a},\boldsymbol{b}}(\tau)\mathcal{CC}_{\boldsymbol{a},\boldsymbol{b}}(\tau+k) = \sum_{l=0}^{s-1}\mathcal{A}_{\boldsymbol{b}}(l)\mathcal{A}_{\boldsymbol{a}}(l+k). \tag{9}$$

Proof. By Equation (8), we obtain

$$\begin{aligned}&\textstyle\sum_{\tau=0}^{t-1} \ \mathcal{CC}_{\mathbf{a},\mathbf{b}}(\tau)\mathcal{CC}_{\mathbf{a},\mathbf{b}}(\tau+k)\\ &= \sum_{\tau=0}^{t-1}\sum_{i=0}^{s-1}(-1)^{a_{i+\tau}+b_i}\sum_{j=0}^{s-1}(-1)^{a_{j+\tau+k}+b_j}\\ &= \sum_{i=0}^{s-1}\sum_{j=0}^{s-1}(-1)^{b_i+b_j}\sum_{\tau=0}^{t-1}(-1)^{a_{i+\tau}+a_{j+\tau+k}}\\ &= \sum_{l=0}^{s-1}\sum_{i=0}^{s-1}(-1)^{b_{i+l}+b_i}\sum_{\theta=0}^{t-1}(-1)^{a_{\theta+l+k}+a_\theta}\\ &= \sum_{l=0}^{s-1}\mathcal{A}_{\mathbf{b}}(l)\mathcal{A}_{\mathbf{a}}(l+k).\end{aligned}$$

□

Theorem 2. *Let $\boldsymbol{a}$ and $\boldsymbol{b}$ be two $m-$sequences of periods t and s, where $t|s$,* i.e., *$\exists d \in \mathrm{Z}$ such that $s = dt$. Then the cross-correlation function value $\mathcal{CC}_{\boldsymbol{a},\boldsymbol{b}}(\tau)$ defined as Equation (8) satisfies the following relations.*

(2-1) $\sum_{\tau=0}^{t-1} \mathcal{CC}_{\mathbf{a},\mathbf{b}}(\tau) = 1.$

(2-2) $\sum_{\tau=0}^{t-1} (\mathcal{CC}_{\mathbf{a},\mathbf{b}}(\tau) + 1) = t + 1.$

(2-3) $\sum_{\tau=0}^{t-1} \mathcal{CC}^2_{\mathbf{a},\mathbf{b}}(\tau) = st + t - d.$

(2-4) $\sum_{\tau=0}^{t-1} \mathcal{CC}_{\mathbf{a},\,\mathbf{b}}(\tau)\mathcal{CC}_{\mathbf{a},\,\mathbf{b}}(\tau+k) = -s-d-1,$ for $k \neq 0.$

(2-5) $\sum_{\tau=0}^{t-1} (\mathcal{CC}_{\mathbf{a},\,\mathbf{b}}(\tau) + 1)^2 = st + 2t - d + 2.$

Proof. (2-1) According to Equation (8), then

$$\begin{aligned}\sum_{\tau=0}^{t-1} \mathcal{CC}_{\mathbf{a},\mathbf{b}}(\tau) &= \sum_{\tau=0}^{t-1}\sum_{i=0}^{s-1}(-1)^{a_{i+\tau}+b_i} \\ &= \sum_{i=0}^{s-1}(-1)^{b_i}\sum_{\tau=0}^{t-1}(-1)^{a_{i+\tau}} \\ &= (-1)(-1) = 1.\end{aligned}$$

Since an $m-$sequence is almost balanced in the sense that it contains one more one than a zero during its period. (2-2) Due to (2-1), the assertion (2-2) is clear. (2-3) By using Equation (9) and the 2-level autocorrelation properties of the two $m-$sequences, we have

$$\begin{aligned}\sum_{\tau=0}^{t-1} \mathcal{CC}^2_{\mathbf{a},\mathbf{b}}(\tau) &= \sum_{l=0}^{s-1} \mathcal{A}_{\mathbf{b}}(l)\mathcal{A}_{\mathbf{a}}(l) \\ &= \mathcal{A}_{\mathbf{b}}(0)\mathcal{A}_{\mathbf{a}}(0) + \mathcal{A}_{\mathbf{b}}(1)\mathcal{A}_{\mathbf{a}}(1) + \cdots \\ &= st - t(d-1) + (s - 1 - (d-1)) \\ &= st + t - d.\end{aligned}$$

(2-4) According to Equation (8) and the 2-level autocorrelation properties of the two $m-$sequences, then

$$\begin{aligned}\sum_{\tau=0}^{t-1} \mathcal{CC}_{\mathbf{a},\,\mathbf{b}}(\tau)\mathcal{CC}_{\mathbf{a},\,\mathbf{b}}(\tau+k) &= -s - td + (s - 1 - d) \\ &= -s - d - 1.\end{aligned}$$

(2-5) By using (2-1)(2-3), the assertion (2-5) is clear. □

Example 3. *Let $\boldsymbol{a} = 011$ and $\boldsymbol{b} = 000100110101111$ be two $m-$sequences with periods 3 and 15, respectively, then by using Equation (8), we compute their cross-correlation functions as following:*

$\mathcal{CC}_{\boldsymbol{a},\boldsymbol{b}}(0), \mathcal{CC}_{\boldsymbol{a},\boldsymbol{b}}(1), \cdots, \mathcal{CC}_{\boldsymbol{a},\boldsymbol{b}}(14)$
$= -5, 3, 3, -5, 3, 3, -5, 3, 3, -5, 3, 3, -5, 3, 3.$

Thus, we obtain

$$\mathcal{CC}_{\mathbf{a},\mathbf{b}}(\tau) = \begin{cases} -5, & \text{if } \tau \equiv 0 \pmod 3, \\ 3, & \text{if } \tau \not\equiv 0 \pmod 3. \end{cases} \quad (10)$$

Case 1: Using Equation (10), we have

(1-a) $\sum_{\tau=0}^{2} \mathcal{CC}_{\mathbf{a},\mathbf{b}}(\tau) = -5 + 3 + 3 = 1.$

(1-b) $\sum_{\tau=0}^{2} (\mathcal{CC}_{\mathbf{a},\mathbf{b}}(\tau)+1) = (-5+1)+(3+1)+(3+1) = 4.$

(1-c) $\sum_{\tau=0}^{2} \mathcal{CC}^2_{\mathbf{a},\mathbf{b}}(\tau) = \mathcal{CC}^2_{\mathbf{a},\mathbf{b}}(0) + \mathcal{CC}^2_{\mathbf{a},\mathbf{b}}(1) + \mathcal{CC}^2_{\mathbf{a},\mathbf{b}}(2) = 25 + 9 + 9 = 43.$

(1-d) $\sum_{\tau=0}^{2} \mathcal{CC}_{\mathbf{a},\,\mathbf{b}}(\tau)\mathcal{CC}_{\mathbf{a},\,\mathbf{b}}(\tau+k)$
$= \mathcal{CC}_{\mathbf{a},\,\mathbf{b}}(0)\mathcal{CC}_{\mathbf{a},\,\mathbf{b}}(k) + \mathcal{CC}_{\mathbf{a},\,\mathbf{b}}(1)\mathcal{CC}_{\mathbf{a},\,\mathbf{b}}(1+k) + \mathcal{CC}_{\mathbf{a},\,\mathbf{b}}(2)\mathcal{CC}_{\mathbf{a},\,\mathbf{b}}(2+k)$
$= -5 \times 3 + 3 \times 3/ - 5 \times 3 = -21.$

(1-e) $\sum_{\tau=0}^{2} (\mathcal{CC}_{\mathbf{a},\,\mathbf{b}}(\tau) + 1)^2$
$= (\mathcal{CC}_{\mathbf{a},\,\mathbf{b}}(0)+1)^2 + (\mathcal{CC}_{\mathbf{a},\,\mathbf{b}}(1)+1)^2 + (\mathcal{CC}_{\mathbf{a},\,\mathbf{b}}(2)+1)^2$
$= (-5+1)^2 + (3+1)^2 + (3+1)^2 = 48.$

Case 2: Using Theorem 2, where $s = 15, t = 3$ and $d = 5$, we have

(2-a) $\sum_{\tau=0}^{2} \mathcal{CC}_{\mathbf{a},\mathbf{b}}(\tau) = 1.$

(2-b) $\sum_{\tau=0}^{2} (\mathcal{CC}_{\mathbf{a},\mathbf{b}}(\tau) + 1) = t + 1 = 3 + 1 = 4.$

(2-c) $\sum_{\tau=0}^{2} \mathcal{CC}^2_{\mathbf{a},\mathbf{b}}(\tau) = st + t - d = 15 \times 3 + 3 - 5 = 43.$

(2-d) $\sum_{\tau=0}^{2} \mathcal{CC}_{\mathbf{a},\,\mathbf{b}}(\tau)\mathcal{CC}_{\mathbf{a},\,\mathbf{b}}(\tau+k) = -s - d - 1 = -15 - 5 - 1 = -21.$

(2-e) $\sum_{\tau=0}^{2} (\mathcal{CC}_{\mathbf{a},\,\mathbf{b}}(\tau)+1)^2 = st + 2t - d + 2 = 15 \times 3 + 2 \times 3 - 5 + 2 = 48.$

By comparison, we get that Case 2 is simpler than Case 1.

4 Conclusions

We study the correlation functions on $m-$sequences of different lengths in this paper. Also, we consider two classes of $m-$sequences of different lengths and give some properties of the correlation functions between these $m-$sequences.

Acknowledgments

This study was supported by the Anhui Provincial Natural Science Foundation (Grant No.1608085MF143) and the Natural Science Foundation of Anhui Higher Education Institutions of China(No.KJ2018A0678). The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] A. D. Bugrov, "The cross correlation of linear recurrent sequences," *Discrete Mathematics and Applications*, vol. 28, no. 2, pp. 65-73, 2018.

[2] X. L. Fang, "New binary sequences with different periods," Master's Thesis, Central China Normal University, Wuhan, China, 2017.(in Chinese)

[3] S. W. Golomb and G. Gong, "Signal design for good correlation for wireless communication, cryptography, and radar," *Engineering & Transportation*, 2005. (`https://www.amazon.com/Signal-Design-Good-Correlation-Communication/dp/0521821045`)

[4] G. Gong and S. W. Golomb, "Binary sequences with two-level autocorrelation," *IEEE Transactions on Information Theory*, vol. 45, no. 2, pp. 692-693, 1999.

[5] T. Helleseth, "Some results about the cross-correlation function between two maximal linear sequences," *Discrete Mathematics*, vol. 16, no. 3, pp. 209-232, 1976.

[6] T. Helleseth, A. Kholosha and G. J. Ness, "Characterization of $m-$sequences of lengths $2^{2k}-1$ and 2^k-1 with three-valued cross correlation," *IEEE Transactions on Information Theory*, vol. 53, no. 6, pp. 2236-2245, 2007.

[7] T. Helleseth, L. Hu, A. Kholosha *et al.*, "Period-different $m-$sequences with at most four-valued cross correlation," *IEEE Transactions on Information Theory*, vol. 55, no. 7, pp. 3305-3311, 2009.

[8] T. Helleseth and A. Kholosha, "$m-$sequences of lengths $2^{2k}-1$ and 2^k-1 with at most four-valued cross correlation," in *Proceedings of the 5th International Conference on Sequences and Their Applications (SETA'08)*, pp.106-120, Sep. 2008.

[9] R. F. Meng, T. J. Yan, "New constructions of binary interleaved sequences with low autocorrelation," *International Journal of Network Security*, vol. 19, no. 4, pp. 546-550, 2017.

[10] G. J. Ness and T. Helleseth, "Cross correlation of $m-$sequences of different lengths," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1637-1648, 2006.

[11] G. J. Ness and T. Helleseth, "A new family of four-valued cross correlation between $m-$sequences of different lengths," *IEEE Transactions on Information Theory*, vol. 53, no. 11, pp. 4308-4313, 2007.

[12] G. J. Ness and T. Helleseth, "A new three-valued cross correlation of between $m-$sequences of different lengths," *IEEE Transactions on Information Theory*, vol. 52, no. 10, pp. 4695-4701, 2006.

[13] W. Su, Y. Yang, Z. C. Zhou, *et al.*, "New quaternary sequences of even length with optimal autocorrelation," *Science China Information Sciences*, vol. 61, no. 2, pp. 022308, 2018.

[14] Y. H. Sun, H. Li, T. J. Yan, "Properties of cross-correlation between two P-ary $m-$sequencs of different periods," *Journal of Xidian University*, vol. 39, no. 5, pp. 30-34, 2012.

[15] N. Y. Yu and G. Gong, "The perfect binary sequence of period 4 for low periodic and aperiodic autocorrelations," in *Sequences, Subsequences, Consequences, International Workshop (SSC'07)*, pp. 37-49, 2007.

[16] T. Zhang, S. X. Li, T. Feng, *et al.*, "Some new results on the cross correlation of $m-$sequences," *IEEE Transactions on Information Theory*, vol. 60, no. 5, pp. 3062-3068, 2014.

Biography

Zepeng Zhuo was born in 1978. He received the M.S. degree from Huaibei Normal University in 2007, and the Ph.D. degree from Xidian University in 2012. Since 2002, he has been with the School of Mathematical Science, Huaibei Normal Universtiy, where he is currently a professor. His research interests include cryptography and information theory.

Jinfeng Chong was born in 1979. She received the M.S. degree from Huaibei Normal University in 2007. Since 2002, she has been with the School of Mathematical Science, Huaibei Normal Universtiy, where she is currently an associate professor. Her research interests include cryptography and information theory.

Lei Yu was born in 1978. He received the M.S. degree from Huaibei Normal University in 2009. Since 2002, he has been with the School of Computer Science and Technology, Huaibei Normal Universtiy, where he is currently an associate professor. His research interests include cryptography and information theory.

Network Security Situation Prediction Based on Grey Relational Analysis and Support Vector Machine Algorithm

Xiaoyi Hong

(Corresponding author: Xiaoyi Hong)

Xinxiang Vocational and Technical College

No. 6, Jingsan Road, Economic and technological development zone, Xinxiang, Henan 453006, China

(Email: hongxiaoy@yeah.net)

(Received Feb. 2, 2019; revised and accepted Dec. 2, 2019)

Abstract

At present, the Internet tends to be omni-directional and multi-angle, and the era of big data dominance has come. However, due to the complex network users and the huge amount of network data, the current situation of network security is worrying. Therefore, the prediction of network security situation is a key link. In this study, the network evaluation index was weighed using grey relational analysis (GRA) theory, the prediction process was simulated based on support vector machine (SVM) algorithm, GRA-SVM based network security situation prediction model was constructed, actual data were substituted into the model, and the results of GRA-SVM and SVM algorithms were compared. The results showed that, compared with single SVM algorithm, the model built by GRA-SVM algorithm had higher prediction precision, which was a reliable algorithm for predicting network security situation. The application of GRA-SVM algorithm can predict the various risks of network development, which can provide a reference for the early preparation of the protection system and minimize the damage to network security.

Keywords: Grey Relational Analysis Theory; Network Security; Support Vector Machine Algorithm

1 Introduction

With the gradual construction and improvement of the Internet, the number of users has a blowout growth. However, the openness of network information and the randomness of the use of network data make the network security problems behind the prosperity frequent. People gradually realize the urgency of improving network security, and network security situation prediction is the most important.

Wei *et al.* [8] proposed a weighted hidden Markov model (HMM), applied multi-scale entropy information to solve the problem of training data, and optimized the transfer matrix of HMM. In addition, they proved that the autocorrelation coefficient could reasonably use the correlation between the characteristics of historical data to predict future security conditions.

Jiang *et al.* [7] trained radial basis function (RBF) neural network to find the mapping relationship between the first N data and the subsequent Mdata and then adjusted its value. The results showed that the method had fast convergence and good prediction effect.

Huang *et al.* [6] proposed a new approach based on artificial immune system and phase space reconstruction, analyzed the time series of network security, reconstructed the appropriate time series phase space, and constructed the prediction model using immune evolution mechanism.

Zhang *et al.* [16] constructed a prediction model based on wavelet neural network (WNN) using the improved niche genetic algorithm (INGA), optimized the parameters of WNN by adaptive genetic algorithm (GA) to make it search more effectively, and solved the premature convergence problem of genetic algorithm using dynamic fuzzy clustering and elimination mechanism.

Hu *et al.* [5] proposed a new prediction model called cloud belief rule base (CBRB) model and represented belief parameter rules using cloud model, which made it more accurate to express expert knowledge. The experimental results proved the practicability of the proposed CBRB model. In this study, correlation analysis based on grey relational analysis (GRA) principle and support vector machine (SVM) algorithm were used for parameter integration. On the basis of them, a GRA-SVM network security situation prediction model was constructed, and it was applied to the specific network situation prediction and compared with the SVM model without correlation analysis.

2 Network Security and Situation Prediction

Network security [10] generally refers to various security problems occurring on the network. However, unlike traditional security issues, it is a special concept based on the development of network and the new challenges of information security in the process of network development. Issues such as personal information fraud, frequent network vulnerabilities and network system shocks are the consequences of network security damage.

Network security situation prediction [14] is the recognition of network security status, including the fusion processing of the old data obtained, extracting the background state and activity semantics of the network system, identifying the possible abnormal activities in the network, and finally outputting the network security situation prediction results based on the above characterization. It is an early warning of network security problems and an important defensive measure to protect the stability of network system, so it is widely used in the maintenance of network system.

3 SVM Algorithm

SVM algorithm as a data mining technology is used to solve classification problems [9]. Based on the basic construction of statistical principle, a kernel function is added to the calculation process to map the low-dimensional problem to the high-dimensional space, and finally the optimal solution in the high-dimensional solution space is obtained [11]. It means that using SVM algorithm can unlock hidden patterns in a large amount of data, so as to discover the information behind the data. After uploading information to the system, the system can identify the time series or development trend of the data and make accurate judgments. The basic structure of the SVM algorithm is shown in Figure 1.

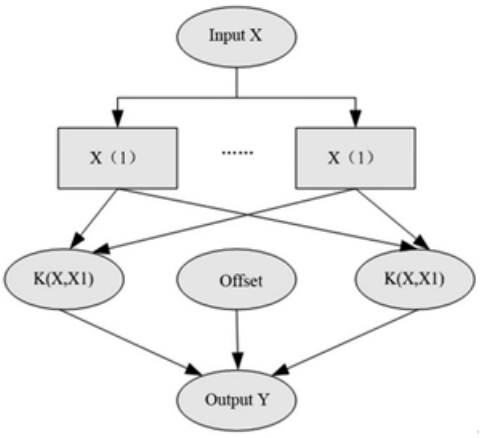


Figure 1: Basic structure of SVM algorithm

4 GRA-SVM Based Model Construction

4.1 Construction Method

4.1.1 SVM Algorithm

The regression equation [1] is an expression that reflects the regression relationship between one variable and another. The regression equation of SVM algorithm is [15]:

$$\begin{aligned} f(x) &= w \cdot \varphi(x) + b \\ \varphi &: R^n \to G, w \to G, \end{aligned}$$

where n represents n situation value training samples $\{x, y\}$, $i = 1, 2, \cdots .n$, x represents the training input value, y_i stands for the output results, w stands for weight vector, and b stands for offset vector.

The optimization function is used for optimization:

$$\min J = \frac{1}{2}||w||^2 + C\sum_{i=1}^{n}(\xi_i^* + \xi_i),$$

where ξ_i^*, ξ_i are the relaxation factors and C is the punishment factor. The constraints in the formula are as follows.

In order to get the final regression result of SVM, it is necessary to substitute Lagrange multiplier. The Lagrange multiplier method [4] is an algorithm for network parameter error identification, which is often used to process support vectors and solve conditional extremum problems. Therefore a_i and a_i^* are substituted into the formula, and then the SVM regression expression is obtained:

$$f(x) = \sum_{i=1}^{R}(a_i - a_i^*)(\varphi(x_i), \varphi(x)) + b$$

Finally, $(\varphi(x_i), \varphi(x))$ is replaced by kernel function $k(x_i, x)$ to solve the problem of curse of dimensionality in the process of non-linear regression prediction. Finally, the following formula is obtained:

$$f(x) = \sum_{i=1}^{n}(a_i - a_i^*)k(x_i, x) + b.$$

4.1.2 GRA Principle

Because the parameters of GRA and SVM algorithm are repetitive for the same case, the same part will not be repeated. The situation value is set as (χ_0), and the evaluation index is set as χ_i; then the corresponding observation value can be obtained. In order to facilitate the calculation, the data need to be dimensionless, so as to simplify the calculation process [3]. The expressions are:

$$\begin{aligned} \chi_0 &= \{x_0(1), x_0(2), \cdots, x_0(n)\}, \\ \chi_i &= \{x_i(1), x_i(2), \cdots, x_i(n)\}, \end{aligned}$$

where

$$x_0(k) = \frac{y_0(k)}{\sum_{t=1}^{n} y_0(t)}$$
$$x_i(k) = \frac{y_i(k)}{\sum_{t=1}^{n} y_i(t)}$$

The correlation coefficient is the bridge and link to obtain the correlation degree. It can be calculated according to the following formula:

$$\xi(k) = \frac{\min_i \min_k |x_0(k) - x_i(k)| + \rho \cdot \max_i \max_k |x_0(k) - x_i(k)|}{|x_0(k) - x_i(k)| + \rho \cdot \max_i \max_k |x_0(k) - x_i(k)|}$$

where ρ represents the resolution coefficient.

When the data is large, there will be many correlation coefficients, so in order to facilitate comparison, the average correlation coefficient is taken as the correlation degree in GRA principle [13]. Correlation degree r_i can be expressed by:

$$r_i = \frac{1}{n} \sum_{k=1}^{n} \varphi_i(k).$$

Finally, by weighting the sequence, the final weighted correlation degree is obtained:

$$r_i = \frac{1}{n} \sum_{k=1}^{n} W(k)\varphi_i(k).$$

4.2 Construction Process

The preliminary construction process of GRA-SVM based network security situation prediction model can be represented by Figure 2.

After initializing the historical data of samples, the weight of correlation degree can be calculated by GRA correlation formula. According to these data, training set and prediction set are generated. Then SVM algorithm is used to model the training samples. The evaluation values of each class are stored in training set List1 in descending order for training [2]. The data in List1 is transmitted into the training module of the prediction model. 24 h is taken as a round, and the attack situation values of different indicators per hour are predicted and output. The actual values of historical data are input and compared with the predicted value. GRA-SVM based prediction model can be established when certain accuracy is satisfied.

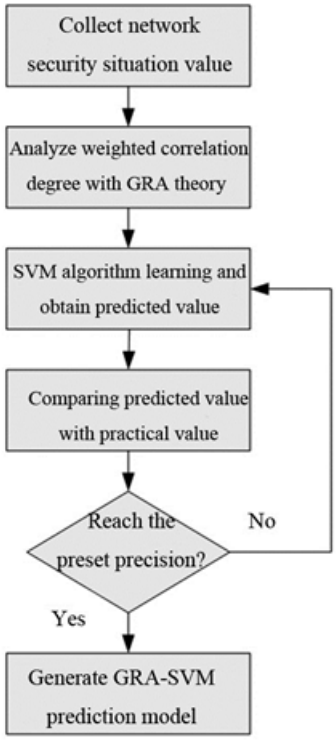


Figure 2: Overall flow of GRA-SVM based model construction

5 Experimental Verification

5.1 Prediction Results of GRA-SVM Based Model

A GRA-SVM prediction model can be obtained by inputting the weighted correlation degree of the index after GRA analysis into the training program of SVM prediction. After the model is established, the original stored historical data are cleared, and only the running program of the algorithm itself is left. Then the new prediction results can be obtained through the analysis of the GRA-SVM based prediction model.

1) Index Collection:
In order to ensure the accuracy of the prediction system and the scientificity of the experiment, a lot of information needs to be collected. The historical data of 200 alarm messages from July 1 to July 10, 2019 was collected and classified according to different network security issues. After analyzing the corresponding historical data, the GRA-SVM based prediction model can get the final result. Similarly, the alarm information from July 20 to July 30 was selected as the actual situation value in the future.

   As the selected data set is very large, the data are normalized [12] to simplify the process. The following formula is used:

$$\hat{x} = \frac{x - x_{\min}}{x_{\max} - x_{\min}}$$

   where x stands for the original situation value, $\hat{x}$ stands for the normalization number, $x_{\max}$ stands for the maximum situation value, and $x_{\min}$ stands for the minimum situation value.

2) Model Analysis Data
The average correlation coefficient in the model is cal-

culated after substituting the formula, and the historical data are selected. The category item with few alarms which are selected from the 200 alarm information was abandoned. The remaining categories are sorted, and the top six evaluation indicators of the correlation degree are selected and ranked in descending order. The results of GRA analysis for each index are shown in Table 1.

According to the correlation degree, the weighted correlation degrees, and the weights of different evaluation indexes, can be obtained. The weights are input into the SVM algorithm program, and the final prediction results of different indexes are numbered in descending order. The results are shown in Table 2.

5.2 Result Comparison

After calculation, the predicted value of the GRA-SVM based network security situation prediction model was compared with the actual value of the future security situation, as shown in Figure 3. The indexes are ranked again and numbered according to the weight of correlation degree, corresponding to the x axis.

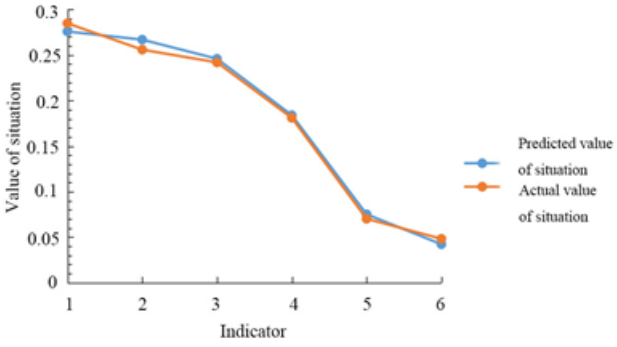


Figure 3: Comparison of predicted and actual situation values

It can be seen from Figure 3 that the two curves are approximately equal and the endpoints coincide approximately. The numerical difference of network bandwidth situation value numbered 4 is the smallest, which is only 0.003, while the network vulnerability number index numbered 2 is the largest one among all situation values, which is 0.011. However, on the whole, the predicted situation value obtained by the analysis of GRA-SVM based network security situation prediction model is close to the actual situation value obtained from statistical investigation. So it can be said that GRA-SVM based network security situation prediction model is an effective model to predict network security situation.

Since Figure 3 is only a simple comparison of absolute values, it is not enough to get the final result. In order to ensure the rigor of the experiment and further verify the accuracy of GRA-SVM based prediction model, a single SVM prediction model is selected as the control group, which directly inputs the data into the SVM algorithm system without GRA processing. The error between GRA-SVM based prediction model and SVM based prediction model is shown in Figure 4. Two decimals of the error rate are kept.

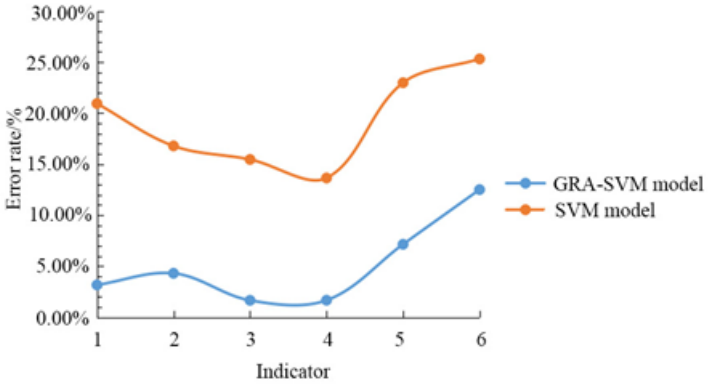


Figure 4: Comparison of error rates between GRA-SVM based prediction model and SVM based prediction model

Figure 4 shows that the error rate curve of the SVM based prediction model is always under the SVM based model. Both of them have small errors in network bandwidth, but in the aspect of the average fault-free time of subnetworks numbered 6, both the results of GRA-SVM and SVM based models have large errors. The greatest error appears when calculating the frequency of critical devices accessing secure websites. The error rate of GRA-SVM based model is 3.16%, while that of SVM based model is 20.93%; the difference between the two models is 17.77%. Thus the accuracy of GRA-SVM algorithm is 17.77% higher than that of the traditional SVM method. Therefore, it is concluded that the application of GRA-SVM based model can significantly improve the accuracy of network security situation prediction and ensure the reliability of the results.

In the practical application of network security situation prediction model, the prediction time and detection range are also important parts. On the one hand, it is because of the complexity of the network security situation value samples; on the other hand, fast prediction of the situation value results is also an indispensable means to solve the network security problems in the first time. Therefore, taking the GRA-SVM based prediction model and SVM based prediction model as the comparison objects, the detection coverage and average detection time of the two models are shown in Figures 5 and 6.

Figure 5 shows that the detection coverage rate of GRA-SVM based prediction model is 26.42% higher than that of SVM based prediction model, i.e. 1.4 times. Figure 6 shows that the average time used by GRA-SVM based prediction model in detecting each index data is 0.22 s less than that of SVM based prediction model, and 38.6% of time can be saved. According to Figures 5 and 6, the GRA-SVM based prediction model is superior to

Table 1: GRA analysis results of correlation degree

Evaluation indicator	Original situation value	Correlation degree
Number of vulnerabilities in network	0.285	0.945
Average failure-free time of subnet	0.256	0.938
Network bandwidth	0.242	0.826
Alarm probability	0.181	0.746
Network bandwidth usage frequency	0.070	0.713
Frequency of critical devices accessing secure websites	0.048	0.695

Table 2: Predicted situation value of network attack

Number	Evaluation indicator	Predicted situation value
1	Frequency of critical devices accessing secure websites	0.276
2	Number of network vulnerabilities	0.267
3	Alarm probability	0.246
4	Network bandwidth	0.184
5	Network bandwidth usage frequency	0.075
6	Average failure-free time of subnet	0.042

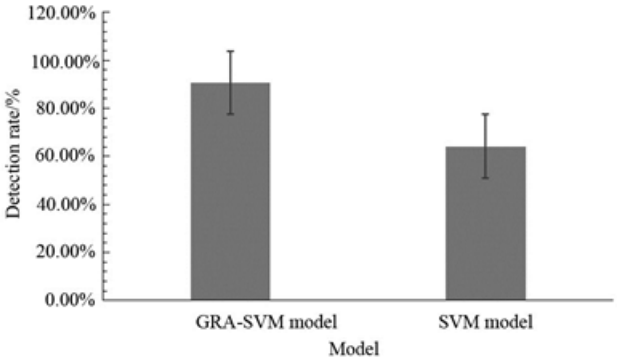


Figure 5: Comparison of detection rate between GRA-SVM based prediction model and SVM based prediction model

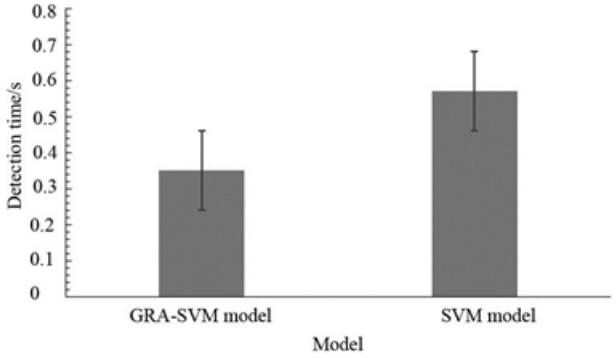


Figure 6: Comparison of detection time between GRA-SVM based prediction model and SVM based prediction model

the SVM based prediction model in terms of overall detection coverage and average detection time. So it can be said that GRA-SVM based prediction model is relatively efficient and more suitable for the practical application of network security issues.

6 Conclusion

Network security situation prediction is an indispensable measure and condition to maintain network security. Based on GRA principle and SVM algorithm, a GRA-SVM network security situation prediction model is constructed. Through the analysis of historical data, we can get accurate prediction of future situation value. In order to verify the relative reliability of the results, it is compared with the model constructed by the traditional SVM algorithm. The research process of this paper shows that:

1) The construction of GRA-SVM based network security situation prediction model can effectively improve the efficiency of situation value prediction, thus gaining more time for repairing network security problems. Moreover it is conducive to making full preparations for network security threats in the future and reducing the risk of accidents.

2) This experiment contributes to maintaining the stability of the network environment and is conducive to providing a more secure and unblocked network environment for Internet users, thus promoting the development of the Internet industry.

References

[1] A. Bhatnagar, A. Sinha, S. Chaudhary, N. Manuja, H. Kaur, T. R. Chaitra, "Accuracy and evaluation of a new regression equation in predicting the width of unerupted permanent canines and premolar teeth," *European Archives of Paediatric Dentistry Official Journal of the European Academy of Paediatric Denstry*, vol. 18, no. 1, pp. 31–37, 2017.

[2] B. Biggio, B. Nelson, P. Laskov, "Poisoning attacks against support vector machines," in *Proceedings of the 29th International Coference on International Conference on Machine Learning*, pp. 1467–1474, 2012.

[3] M. Conesa, J. F. Sánchez, I. Alhama, F. Alhama, "On the nondimensional of coupled, nonlinear ordinary differential equations," *Nonlinear Dynamics*, vol. 84, no. 1, pp. 91–105, 2016.

[4] M. D. R. De Pinho, I. Shvartsman, "Lipschitz continuity of optimal control and Lagrange multipliers in a problem with mixed and pure state constraints," *Discrete & Continuous Dynamical Systems-Series A*, vol. 29, no. 2, pp. 505–522, 2017.

[5] G. Y. Hu, P. L. Qiao, "Cloud belief rule base model for network security situation prediction," *IEEE Communications Letters*, vol. 20, no. 5, pp. 914–917, 2016.

[6] T. Q. Huang, Y. Zhuang, "An approach to real-time network security situation prediction," *Journal of Chinese Computer Systems*, vol. 35, no. 2, pp. 303–306, 2014.

[7] Y. Jiang, C. H. Li, L. S. Yu, B. Bao, "On network security situation prediction based on RBF neural network," in *36th Chinese 0Control Conference (CCC'17)*, 2017.

[8] W. Liang, Z. Chen, X. L. Yan, X. D. Zheng, P. Zhuo, "Multiscale entropy-based weighted hidden markov network security situation prediction model," in *IEEE International Congress on Internet of Things*, 2017.

[9] M. D. C. Moura, E. Zio, I. D. Lins, E. L. Droguett, "Failure and reliability prediction by support vector machines regression of time series data," *Reliability Engineering & System Safety*, vol. 96, no. 11, pp. 1527–1534, 2017.

[10] Z. E. Mrabet, N. Kaabouch, H. E. Ghazi, "Cybersecurity in smart grid: Survey and challenges," *Computers & Electrical Engineering*, vol. 67, pp. 469–482, 2018.

[11] L. R. Quitadamo, F. Cavrini, L. Sbernini, F. Riillo, L. Bianchi, S. Seri, G. Saggio, "Support vector machines to detect physiological patterns for EEG and EMG-based human-computer interaction: A review," *Journal of Neural Engineering*, vol. 14, no. 1, ID: 011001, 2017.

[12] A. Suzuki, K. Yamanishi, "Exact calculation of normalized maximum likelihood code length using fourier analysis," Jan. 11, 2018. (`https://arxiv.org/pdf/1801.03705v1.pdf`)

[13] D. Wen, J. Li, P. F. He, "Grey correlation analysis of agro-meteorological disasters and soybean yield in Heilongjiang province," *Journal of Natural Disasters*, vol. 26, no. 4, pp. 56–62, 2017.

[14] R. F. Wu, G. L. Chen, "Research of network security situation prediction based on multidimensional cloud model," in *International Conference on Innovative Mobile & Internet Services in Ubiquitous Computing*, 2012.

[15] Z. Xue, R. Zhang, C. Qin, X. Zeng, "An adaptive twin support vector regression machine based on rough and fuzzy set theories," *Neural Computing and Applications*, 2018. (`https://link.springer.com/article/10.1007\%2Fs00521-018-3823-4`)

[16] H. Zhang, Q. Huang, F. Li, J. Zhu, "A network security situation prediction model based on wavelet neural network with optimized parameters," *Digital Communications and Networks*, vol. 2, no. 3, pp. 139–144, 2016.

Biography

Xiaoyi Hong, born in December 1982, female, master of engineering, is currently teaching in Xinxiang vocational and technical college, and her research direction is computer science and technology.

Guide for Authors
International Journal of Network Security

IJNS will be committed to the timely publication of very high-quality, peer-reviewed, original papers that advance the state-of-the art and applications of network security. Topics will include, but not be limited to, the following: Biometric Security, Communications and Networks Security, Cryptography, Database Security, Electronic Commerce Security, Multimedia Security, System Security, etc.

1. Submission Procedure

Authors are strongly encouraged to submit their papers electronically by using online manuscript submission at http://ijns.jalaxy.com.tw/.

2. General

Articles must be written in good English. Submission of an article implies that the work described has not been published previously, that it is not under consideration for publication elsewhere. It will not be published elsewhere in the same form, in English or in any other language, without the written consent of the Publisher.

2.1 Length Limitation:

All papers should be concisely written and be no longer than 30 double-spaced pages (12-point font, approximately 26 lines/page) including figures.

2.2 Title page

The title page should contain the article title, author(s) names and affiliations, address, an abstract not exceeding 100 words, and a list of three to five keywords.

2.3 Corresponding author

Clearly indicate who is willing to handle correspondence at all stages of refereeing and publication. Ensure that telephone and fax numbers (with country and area code) are provided in addition to the e-mail address and the complete postal address.

2.4 References

References should be listed alphabetically, in the same way as follows:
For a paper in a journal: M. S. Hwang, C. C. Chang, and K. F. Hwang, ``An ElGamal-like cryptosystem for enciphering large messages," *IEEE Transactions on Knowledge and Data Engineering*, vol. 14, no. 2, pp. 445--446, 2002.

For a book: Dorothy E. R. Denning, *Cryptography and Data Security*. Massachusetts: Addison-Wesley, 1982.

For a paper in a proceeding: M. S. Hwang, C. C. Lee, and Y. L. Tang, ``Two simple batch verifying multiple digital signatures," in *The Third International Conference on Information and Communication Security (ICICS2001)*, pp. 13--16, Xian, China, 2001.

In text, references should be indicated by [number].

Subscription Information

Individual subscriptions to IJNS are available at the annual rate of US$ 200.00 or NT 7,000 (Taiwan). The rate is US$1000.00 or NT 30,000 (Taiwan) for institutional subscriptions. Price includes surface postage, packing and handling charges worldwide. Please make your payment payable to "Jalaxy Technique Co., LTD." For detailed information, please refer to http://ijns.jalaxy.com.tw or Email to ijns.publishing@gmail.com.